Section 37 Video and Graphics Functions

37.1 Architecture of the Graphics Subsystem

The Graphics Subsystems comprise following modules:

- Video channels with the Video Input and Output interfaces, the Image Synthesizers and the Video Output Warping Engine.
- Sprite Engine with RLE Units for de-compressing RLE encoded colour data and Sprite Units for sprite layers.

The Graphics Subsystem is connected to the XC0 cross-connect for transfer of video and graphics data to and from the various memories.

The Video Input Interfaces and the Distortion Correction with the Video Output Warping Engine have directly access to the XC0 cross-connect - and thus to the memories.

All memory read accesses of the Image Synthesizers are routed via the XC2 cross-connect to the Sprite Engine, which acquires the colour data from the memories via the cross-connect XC0.

Likewise the other cross-connect systems XC0 and XC1 the XC2 is also a multilayer transaction based bus system.
Refer to Section 14, Bus Architecture for details about the bus architecture.

Refer to the following sections for detailed descriptions about the various functional components:

- Video Input Interface: **Input Controller** in Section 38, Video Data Controller E (VDCE)
- Scaler: **Scaler** in Section 38, Video Data Controller E (VDCE)
- Format Converter: **Image Quality Improver** in Section 38, Video Data Controller E (VDCE)
- Image Synthesizer: **Image Synthesizer** in Section 38, Video Data Controller E (VDCE)
- Distortion Correction: **Output Image Generator** in Section 38, Video Data Controller E (VDCE) and Section 39, Video Output Warping Engine (VOWE)
- Video Output Interface: **Output Controller** in Section 38, Video Data Controller E (VDCE)
- General control functions: **System Controller** in Section 38, Video Data Controller E (VDCE)

CAUTION

Correct operation of the video and graphics functions is not guaranteed if CPU runs operated by the EMCLK.

However, the registers of the video and graphics modules can be accessed.

37.2 Graphics Subsystem Block Diagrams

The Graphics Subsystems of the various devices are different and shown in the following sections.

NOTES

1. The XC0 cross-connect in the following diagrams are not part of the Graphics Subsystem, but are included better understanding of the entire colour data transfers.
2. The arrows in the following diagrams indicate the direction of the colour data flow. For information about the master and slave ports at the cross-connects refer to Section 14, Bus Architecture.

37.2.1 D1L2(H) Graphics Subsystem

The D1L2(H) device features one video output channel and supports four graphic layers.

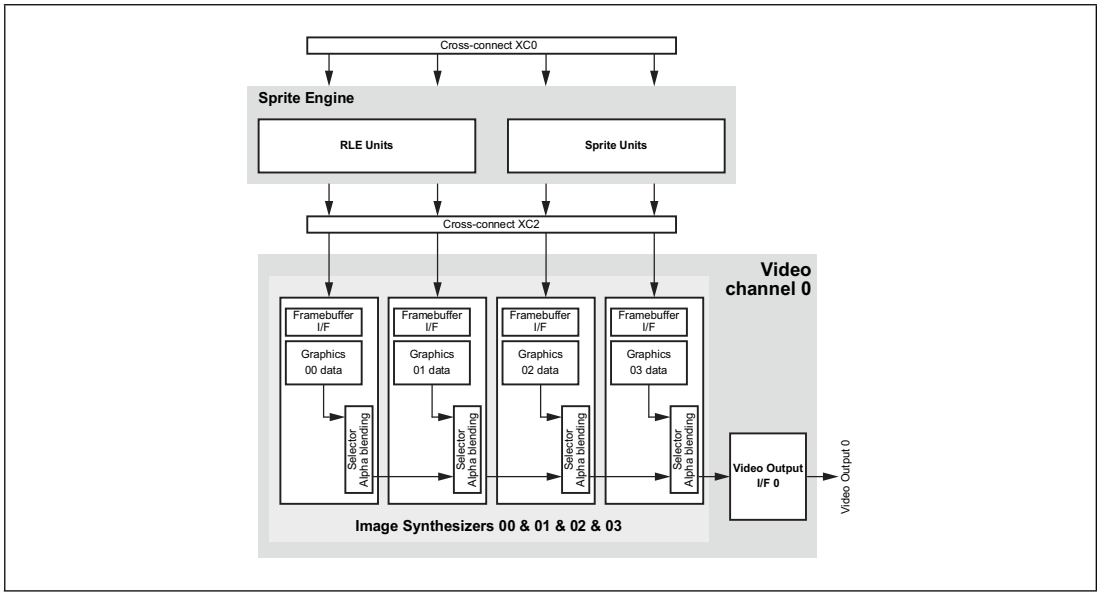


Figure 37.1 D1L2(H) Graphics Subsystem

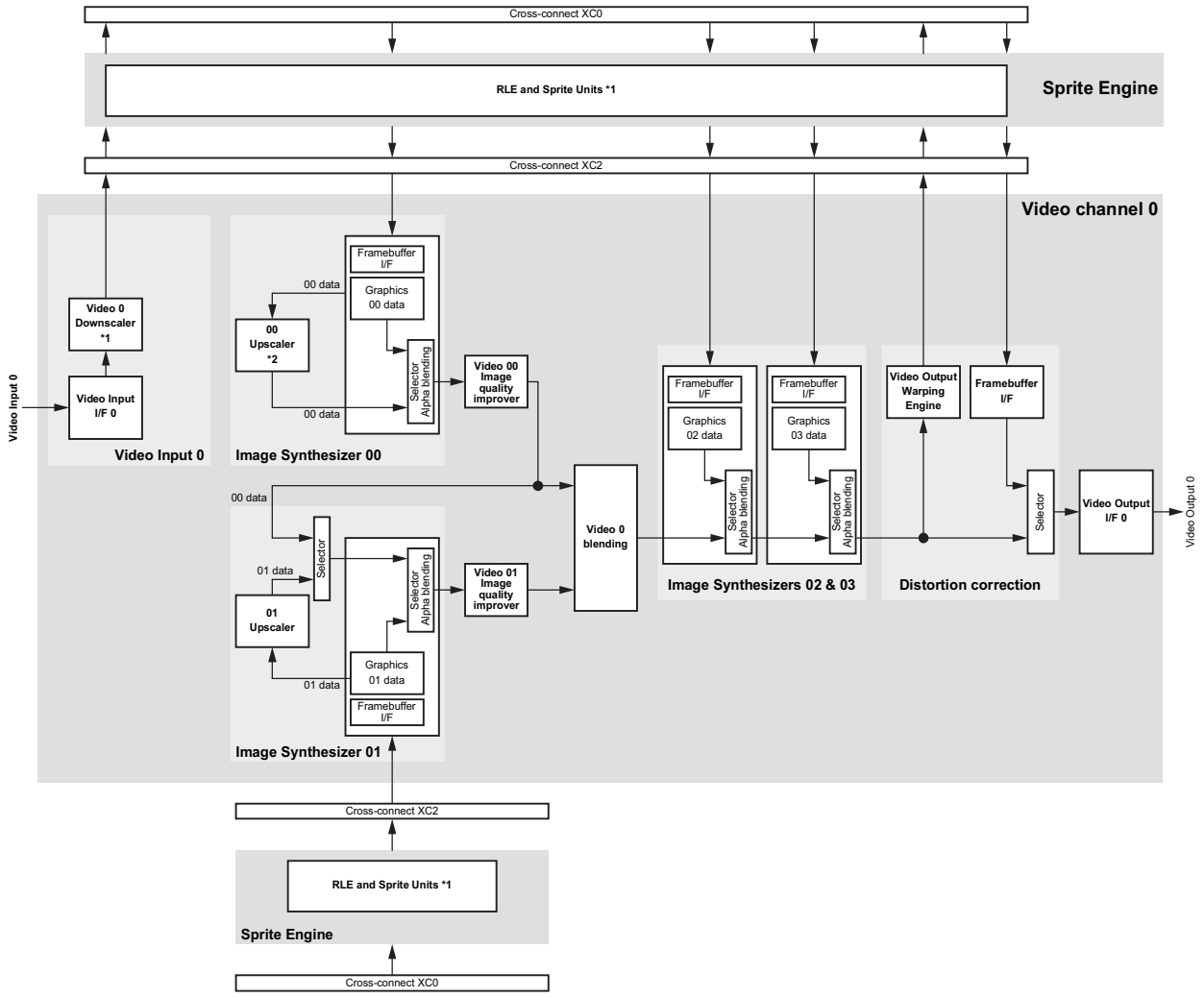
37.2.2 D1M1(H), D1M1-V2, D1M2(H) and D1M1A Graphics Subsystem

The D1M1(H), D1M1-V2devices features one video input and output channel, while the D1M2(H) device provides two video input and two video output channels.

Following layer combinations are possible:

- D1M1(H), D1M1-V2:
  - 4 graphics layers
  - 1 video layer + 3 graphics layers
- D1M2(H) for each of channel 0 and channel 1:
  - 4 graphics layers
  - 1 video layer + 3 graphics layers
  - 2 video layers + 2 graphics layers
- D1M1A:
  - 4 graphics layers (channel 0 and channel 1)
  - 1 video layer + 3 graphics layers (only channel 0)

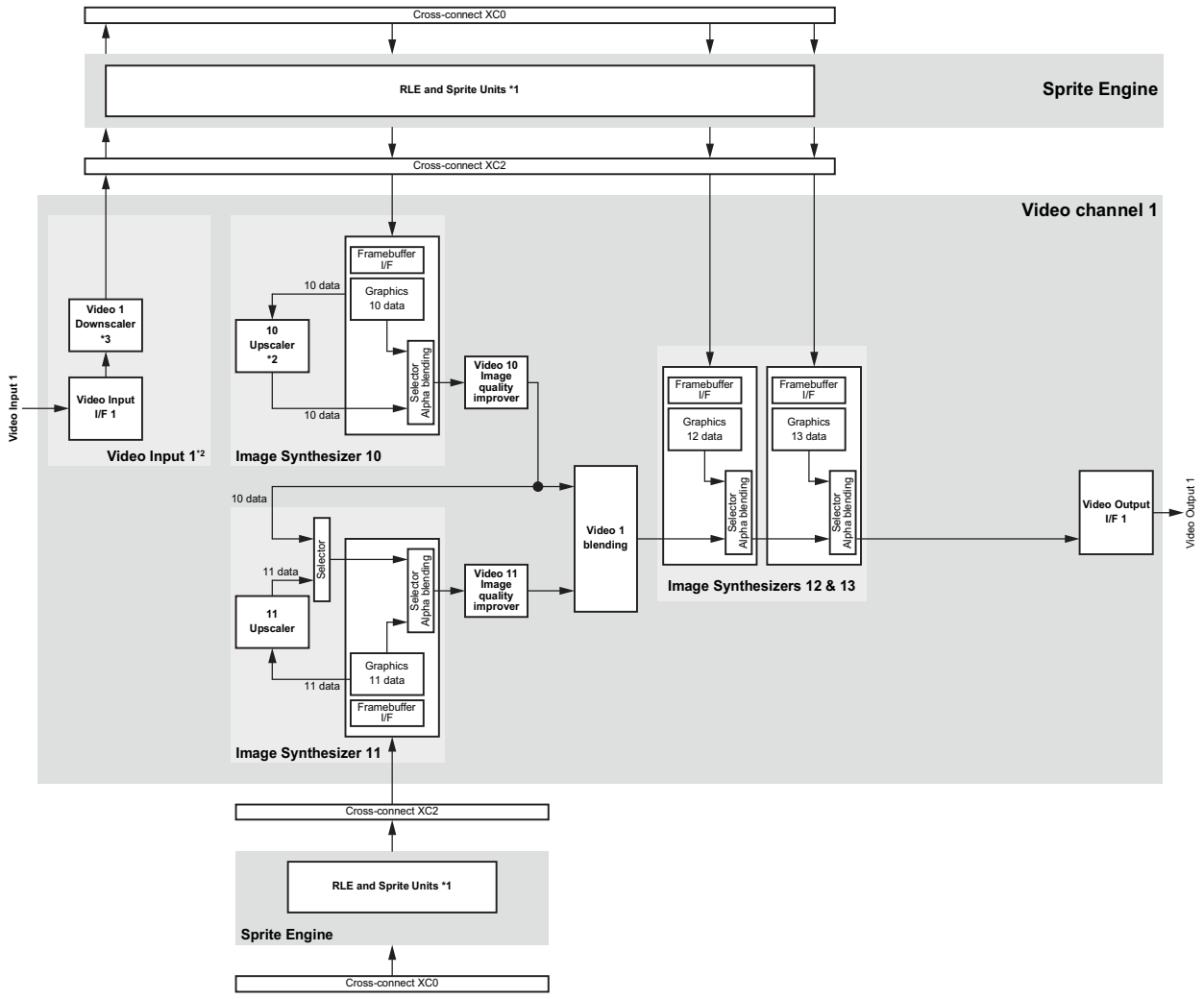
The Video Output channel 0 of the D1M1(H), D1M1-V2, D1M2(H) and D1M1A devices have a Video Output Warping Engine for predistortion of video images.



Note 1. – D1M1(H), D1M2(H): RLE unit can be used for background layer.
– D1M1-V2, D1M1A: RLE units can be used for all layers.

Note 2. The Video 0 Downscaler and 00 Upscaler can not be used concurrently. Either the Downscaler or the Upscaler can be used at the same time.

Figure 37.2 D1M1(H), D1M1-V2, D1M2(H) and D1M1A Graphics Subsystem video channel 0 data flow



Note 1. – D1M2(H): RLE unit can be used for background layer.
– D1M1A: RLE units can be used for all layers.

Note 2. The Video 1 Downscaler and 10 Upscaler can not be used concurrently. Either the Downscaler or the Upscaler can be used at the same time.

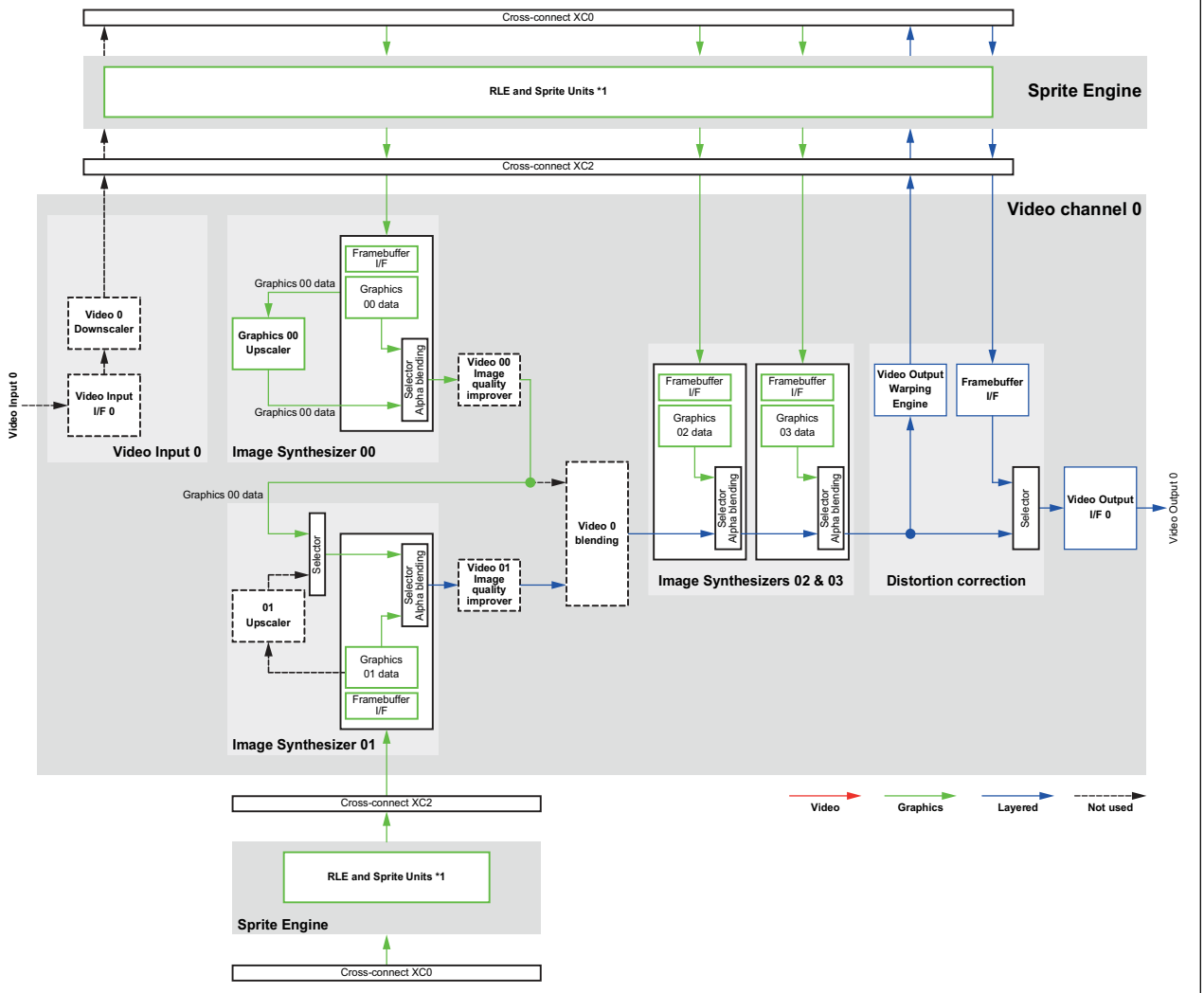
Note 3. The Video Input I/F 1 is not available for D1M1A devices.

Figure 37.3 D1M2(H) and D1M1A Graphics Subsystem video channel 1 data flow

37.3 Video input and output data flow

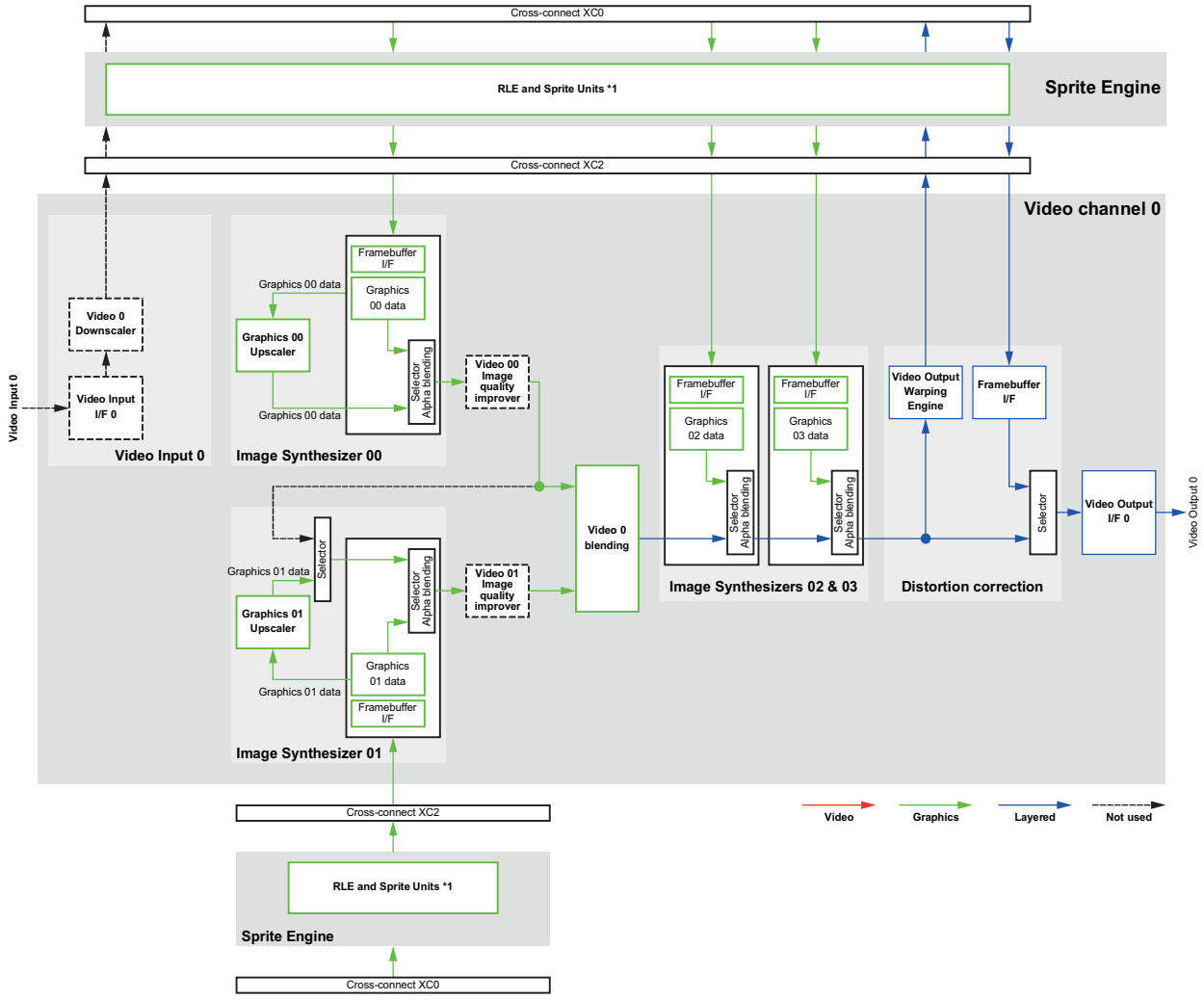
37.3.1 No video layers

If no video layers are used, each of the four Image Synthesizers can generate one graphic layer. The first Image Synthesizer (00 for channel 0, 10 for channel 1 of D1M2(H)) can utilize the upscaler for enlarging the graphics data, which constitutes the lowest layer 0, i.e. the background layer.



Note 1. – D1M1(H), D1M2(H): RLE unit can be used for background layer.
– D1M1-V2, D1M1A: RLE units can be used for all layers.

Figure 37.4 D1M1(H), D1M1-V2, D1M2(H) and D1M1A video channel 0 data flow: four graphics layers - configuration 1

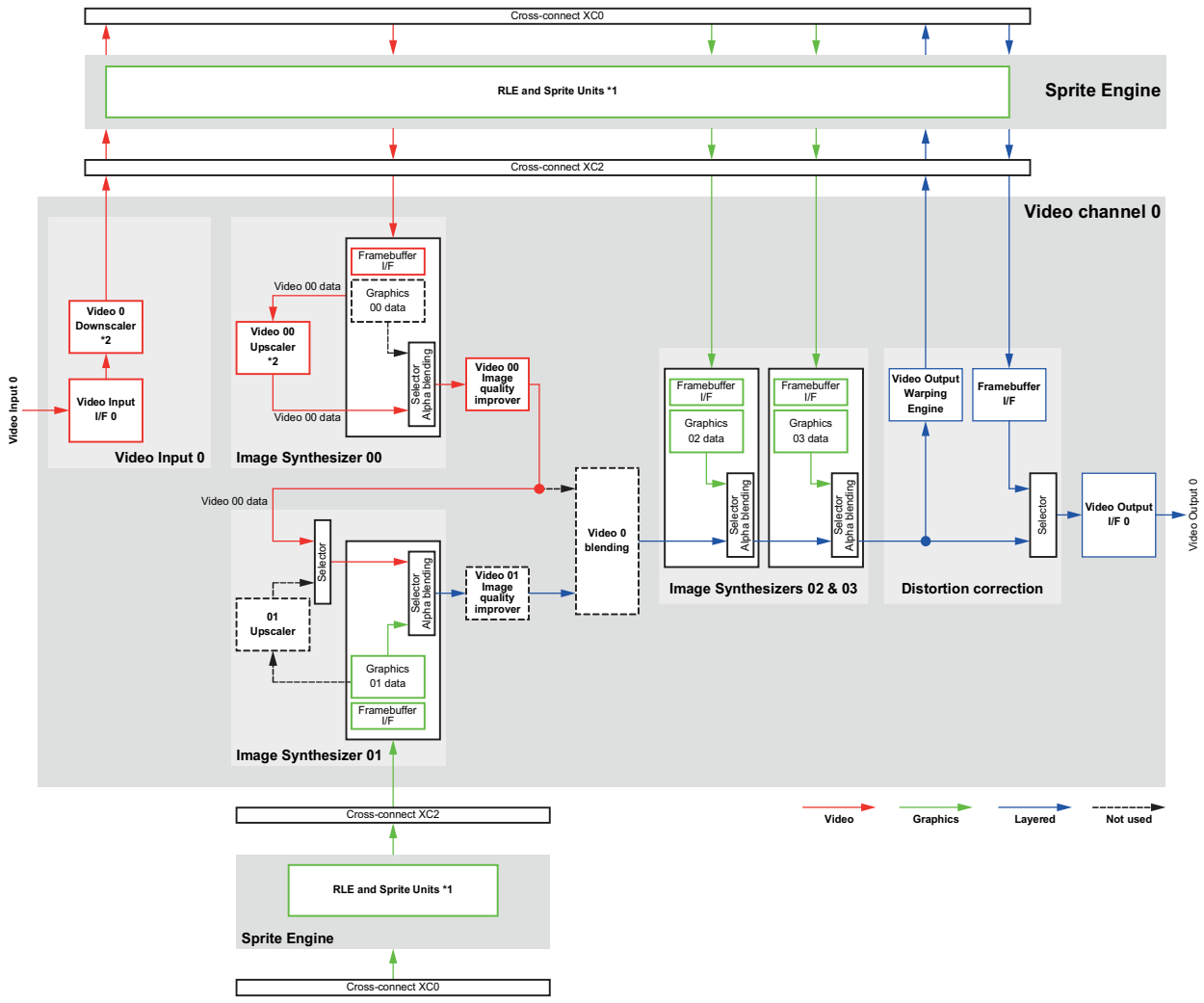


Note 1. – D1M1(H), D1M2(H): RLE unit can be used for background layer.
– D1M1-V2, D1M1A: RLE units can be used for all layers.

Figure 37.5 D1M1(H), D1M1-V2, D1M2(H) and D1M1A video channel 0 data flow: four graphics layers - configuration 2

37.3.2 One video layer

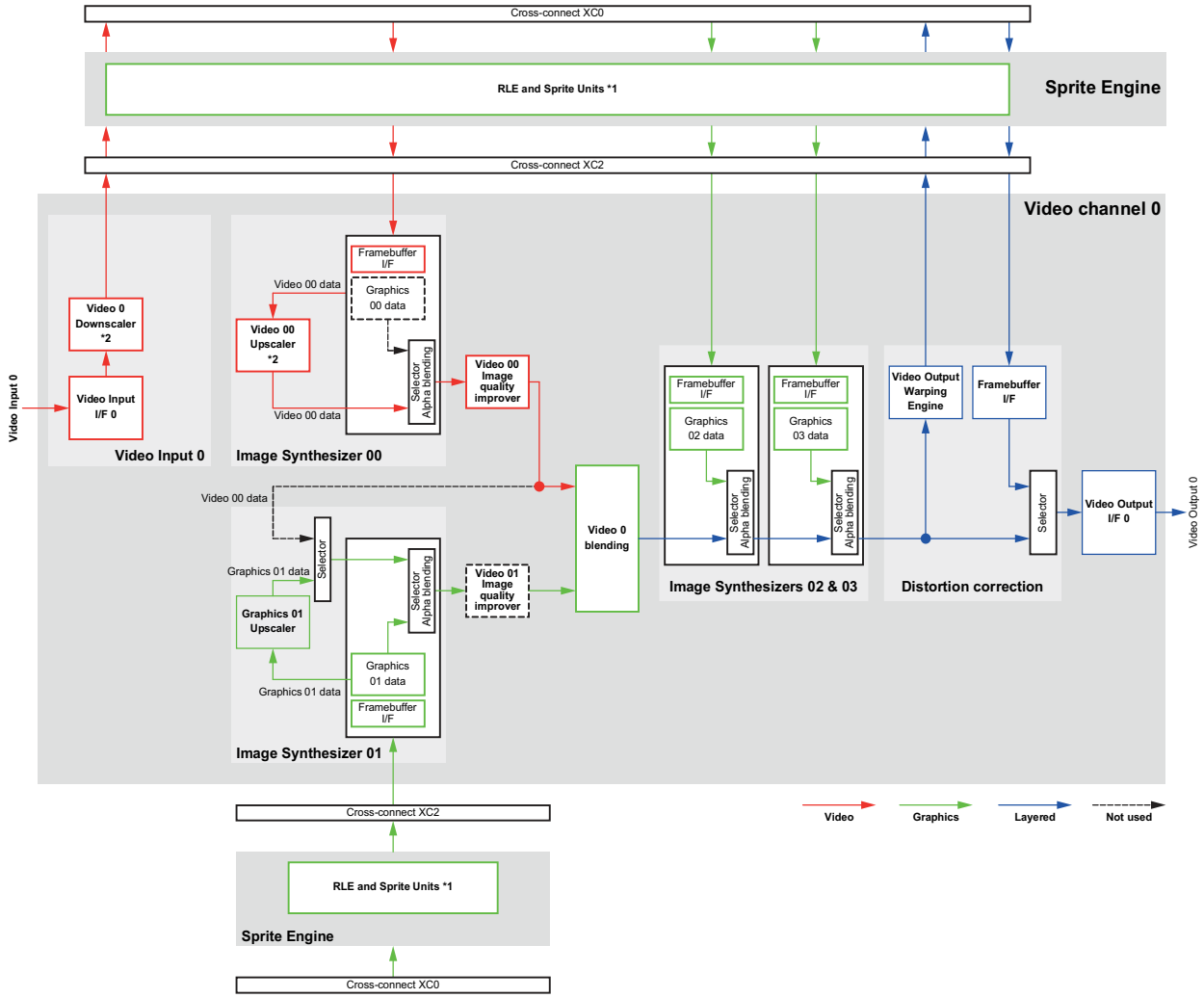
If one video layer is used, the first Image Synthesizer (00 for channel 0, 10 for channel 1 of D1M2(H) and D1M1A) can utilize the upscaler for enlarging the video data, which constitutes the lowest layer 0, i.e. the background layer.
All other Image Synthesizers can add up to three graphics layers.



Note 1. – D1M1(H), D1M2(H): RLE unit can be used for background layer.
– D1M1-V2, D1M1A: RLE units can be used for all layers.
However since captured video data from the Video Input interface is to be used on the background layer in this example, the RLE engines in the RLE units are bypassed anyway.

Note 2. The Video 0 Downscaler and Video 00 Upscaler can not be used concurrently. Either the Downscaler or the Upscaler can be used at the same time.

Figure 37.6 D1M1(H), D1M1-V2, D1M2(H) and D1M1A video channel 0 data flow: one video + up to three graphics layers - configuration 1



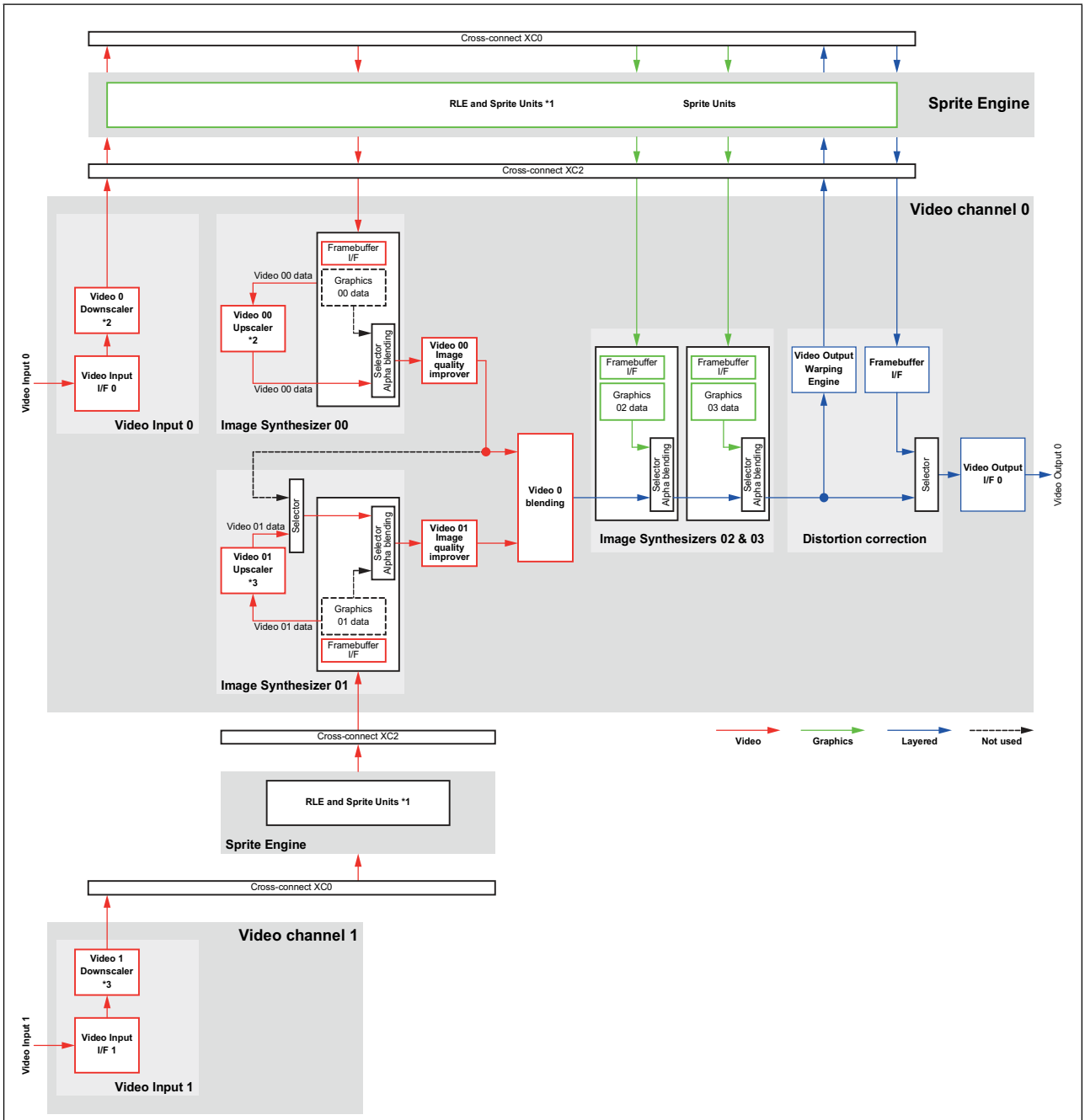
Note 1. – D1M1(H), D1M2(H): RLE unit can be used for background layer.
– D1M1-V2, D1M1A: RLE units can be used for all layers.
However since captured video data from the Video Input interface is to be used on the background layer in this example, the RLE engines in the RLE units are bypassed anyway.

Note 1. The Video 0 Downscaler and Video 00 Upscaler can not be used concurrently. Either the Downscaler or the Upscaler can be used at the same time.

Figure 37.7 D1M1(H), D1M1-V2, D1M2(H) and D1M1A video channel 0 data flow: one video + up to three graphics layers - configuration 2

37.3.3 Two video layers with D1M2H

If the captured data of both D1M2H Video Inputs are used, the Video Input data are scaled up by Image Synthesizer 00 and 01 (10 and 11 for channel 1) and combined by a separate video data blending unit. Blending is performed with a definable constant α value.
The remaining Image Synthesizers can add up to two graphics layers.



Note 1. RLE unit can be used for background layers.
However since captured video data from the Video Input interfaces are to be used on the background layer in this example, the RLE engines in the RLE units are bypassed anyway.

Note 2. The Video 0 Downscaler and Video 00 Upscaler can not be used concurrently. Either the Downscaler or the Upscaler can be used at the same time.

Note 3. The Video 1 Downscaler and Video 01 Upscaler can not be used concurrently. Either the Downscaler or the Upscaler can be used at the same time.

Figure 37.8 D1M2H video channel 0 data flow: two video + up to two graphics layers

37.4 D1M1(H), D1M1-V2, D1M2(H) and D1M1A video input/output synchronization

The D1M1(H), D1M1-V2, D1M2(H) and D1M1A devices provide basically two modes for selecting screen synchronization signals:

- Video output - video input synchronized mode
  In this mode the Video Input and Output is operating frame-synchronous, i.e. the external Video Input vertical synchronization signal is used throughout the entire input - output path.
- Separate mode
  In this mode a separate vertical synchronization signal is generated for the output path, independently from the Video Input.

In either case the horizontal synchronization signals are generated independently from the Video Input. The VSYNC signals are checked and erroneous VSYNC pulses are suppressed and missing VSYNC pulses are substituted in order to ensure stable continuous frame synchronization.

Interrupts

Three interrupts are generated related to the synchronization:

- Video input VSYNC interrupt, marks occurrence of the external VSYNC signal
- Video output VSYNC interrupt, marks occurrence of the VSYNC signals, used in the video output path
- VSYNC substitution interrupts, marks occurrence of a VSYNC substitution

NOTE

For more details refer to “Synchronization Control” in the “Scaler” section of “Video Data Controller (VDCE)”.

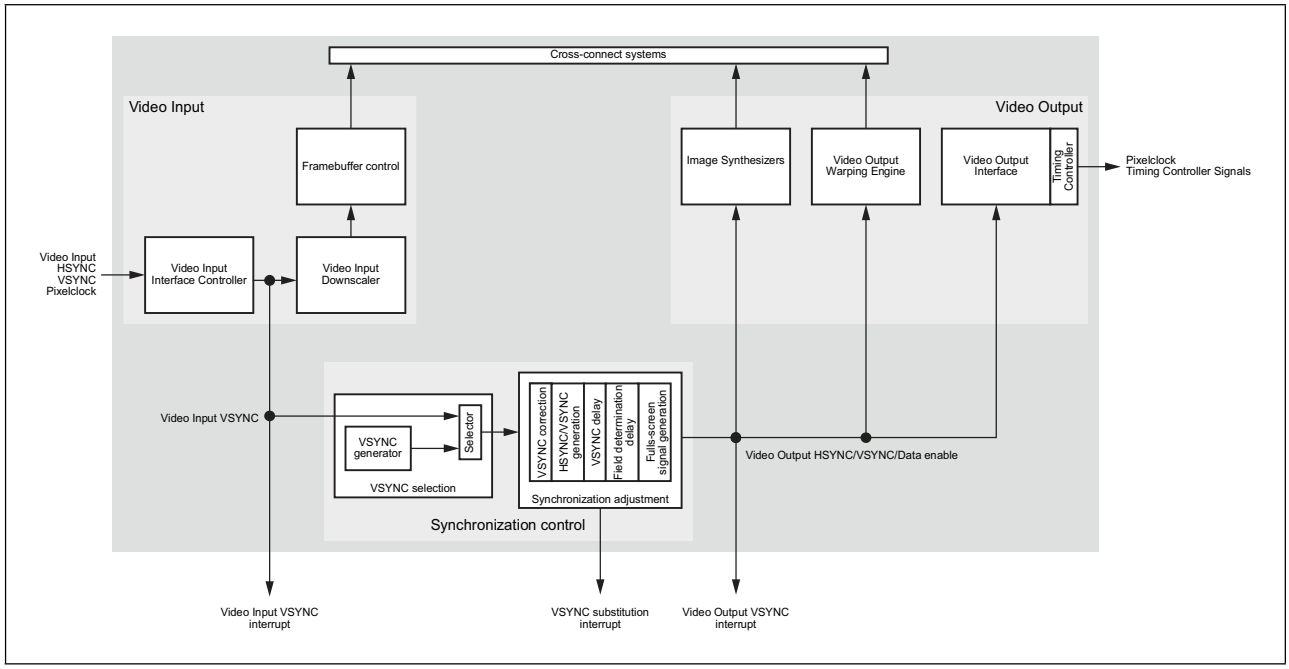


Figure 37.9 D1M1(H), D1M1-V2, D1M2(H) and D1M1A video input/output synchronization

37.5 Video channels clock generation

The clocks for the video channels can be derived from the Spread-Spectrum clock generator PLL0 or from a non-jittered clock of another PLL circuit, in case the connected display does not allow a jittered pixel clock.

The maximum video output pixel clock frequencies C_ISO_VDCEnCLK are:

- D1L2(H): 10 MHz
- D1M1(H), D1M1-V2: 30 MHz
- D1M2(H), D1M1A: 48 MHz

37.5.1 D1L2(H) pixel clock generator

The pixel clock C_ISO_VDCE0CLK of the video output interface is generated by a divider 2..255 from one of the clocks from the clock generators (PLLs).

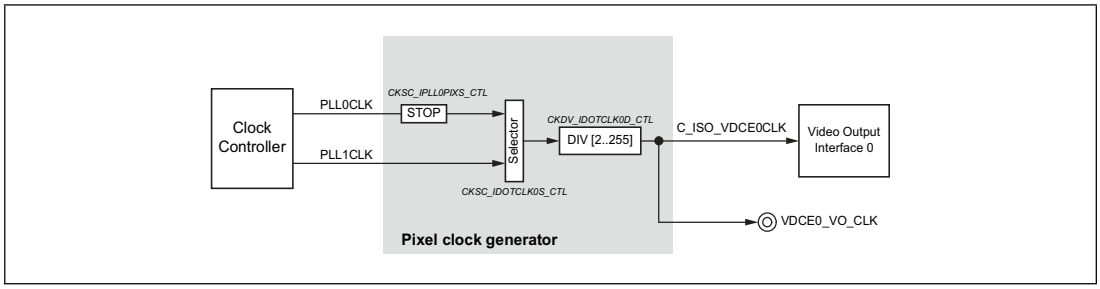


Figure 37.10 D1L2(H) video output pixel clock generation

37.5.2 D1M1(H)pixel clock generator

The pixel clock DOTCLK is generated by a divider 2..255 from one of the clocks from the clock generators (PLLs).

Several options are provided for C_ISO_VI0PIXCLK and C_ISO_VDCE0CLK, depending on the application scenario.

Video input interface used

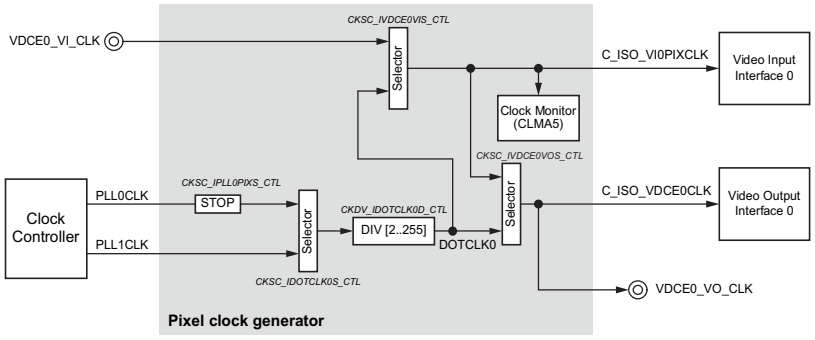
I.e. external video input clock VDCE0_VI_CLK is available.

- C_ISO_VI0PIXCLK = VDCE0_VI_CLK
- C_ISO_VDCE0CLK =
  - DOTCLK: in case video input and output operating with different pixel clocks
  - C_ISO_VI0PIXCLK: in case video input and output operating with same pixel clock

Video input interface not used

I.e. external video input clock VDCE0_VI_CLK is not available.

- C_ISO_VI0PIXCLK = DOTCLK
- C_ISO_VDCE0CLK = DOTCLK



Note 1. The name of the clock control and selection registers are shown in *italics*.

Figure 37.11 D1M1(H) video output pixel clock generation

37.5.3 D1M2(H) video channels clock generator

Video output pixel clock

To generate a video output pixel clock with sufficient granularity the D1M2(H) devices feature an additional dedicated PLL2 circuit, which can generate a jittered (via PLL0) or non-jittered (directly via MainOsc) pixel clock.
However the two Video Output Interfaces share the dedicated PLL2 for pixel clock generation.

Generally the pixel clock C_ISO_VDCEnCLK of the video output interface n is generated by a divider [2..255] from one of the clocks from the clock generators (PLLs).

Additionally each video channel can also select the external video input clock as output pixel clock.

The video output pixel clock generation differs in case a LVTTL/RGB output interface or a RSDS output interface is used.

For operating the RSDS output interface of the D1M2(H) devices a clock with 4 x pixel clock frequency is required by the RSDS logic. The pixel clock of the Video Output Interfaces needs to select a DIV4 of this clock as pixel clock.

If only a LVTTL output interface is used the pixel clock can be derived directly from the DIB [2:255] divider output.

Video input pixel clock

In normal use case the video input pixel clock is taken from the corresponding external signal.

In case of MIPI interface usage the video input clock in generated from the MIPI controller clock (by dividing by 3, 6, 12).

1. MIPI interface used (VDCE0)

   I.e.

   – C_ISO_VI0PIXCLK = C_ISO_MIPIPIXCLK

2. Video input interface used (VDCE0)

   I.e. external video input clock VDCE0_VI_CLK is available:

– C_ISO_VI0PIXCLK = VDCE0_VI_CLK

3. Video input interface not used (VDCE0)

I.e. external video input clock VDCE0_VI_CLK is not available:

– C_ISO_VI0PIXCLK = C_ISO_MIPIPIXCLK

4. Video input interface used (VDCE1)

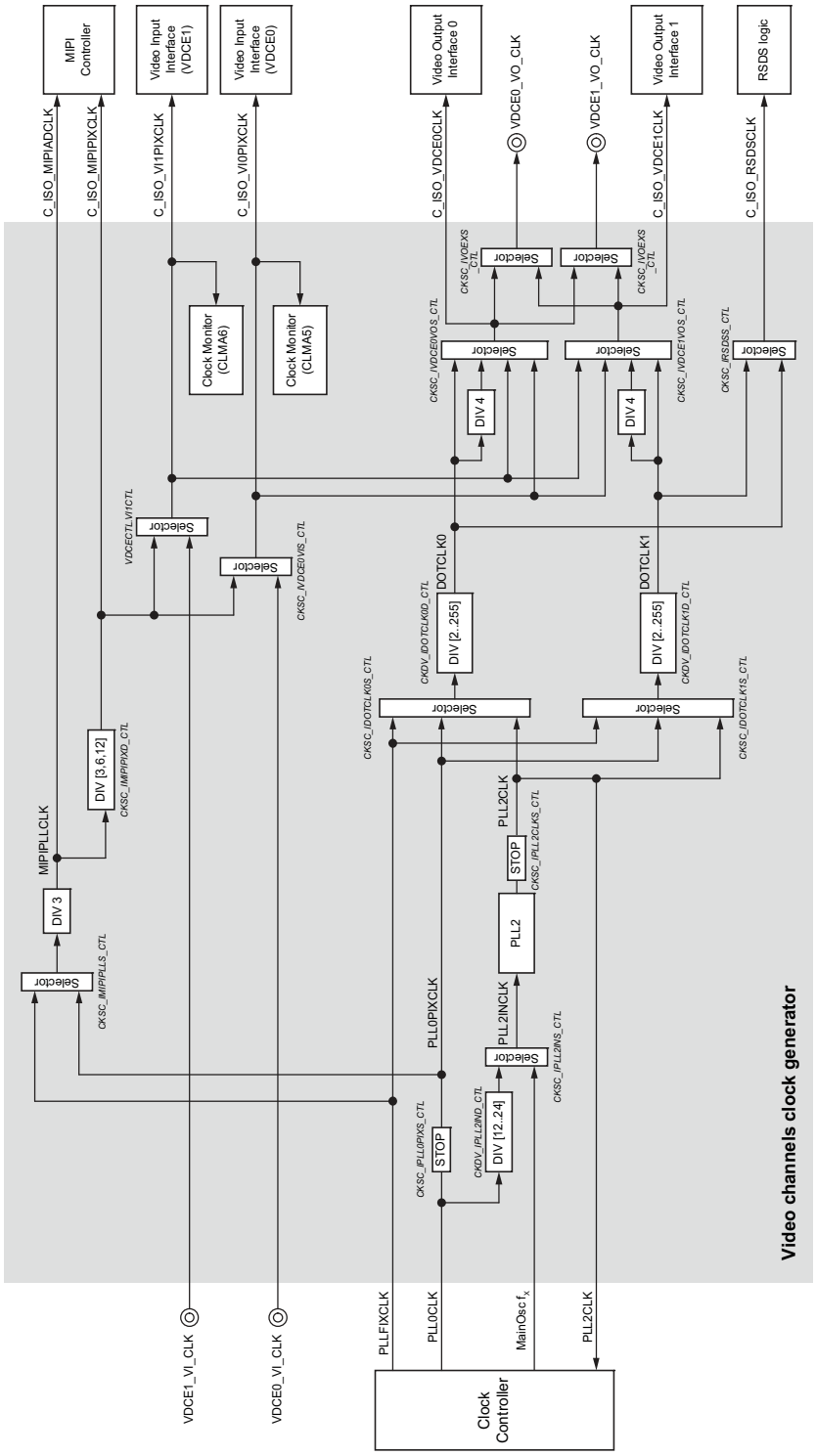
I.e. external video input clock VDCE1_VI_CLK is available:

– C_ISO_VI1PIXCLK = VDCE1_VI_CLK

5. Video input interface not used (VDCE1)

I.e. external video input clock VDCE1_VI_CLK is not available:

– C_ISO_VI1PIXCLK = C_ISO_MIPIPIXCLK



Note 1. The name of the clock control and selection registers are shown in *italics*.

Figure 37.12 D1M2(H) video channels clock generator

37.5.4 D1M1A video channels clock generator

The pixel clock DOTCLKn (n = 0, 1) is generated by a divider [1..255] from one of the clocks from the clock generators (PLLs).

In addition, D1M1A has options for VODDR, Serial RGB and Open LDI output. The video clocks have to be set by following options.

Video input interface used

I.e. external video input clock VDCE0_VI_CLK is available:

- C_ISO_VI0PIXCLK = VDCE0_VI_CLK

Video input interface not used

I.e. external video input clock VDCE0_VI_CLK is not available:

- C_ISO_VI0PIXCLK = DOTCLK

Video output interface clocks selections

The video output clock must be properly selected for the respective output mode.

Regarding the video ouput mode selection, refer to Section 37.9.3.2, D1M1A Video output selection.

(1) Video Output Interface n (VDCEn, n = 0, 1) input clocks selections

Table 37.1 Video Output Interface n input clocks selections

Output mode	C_ISO_VDCEnCLK	C_ISO_VDCEnCLK_OUT
Parallel RGB	CKSC_IVDCEnVOS_CTL*1	
Serial RGB*2	CKSC_IVDCE1VOS_CTL /[3 or 4]*1	CKSC_IVDCE1VOS_CTL*1
VODDR	VOn_PCLK	
Open LDI	DOTCLKn /7	

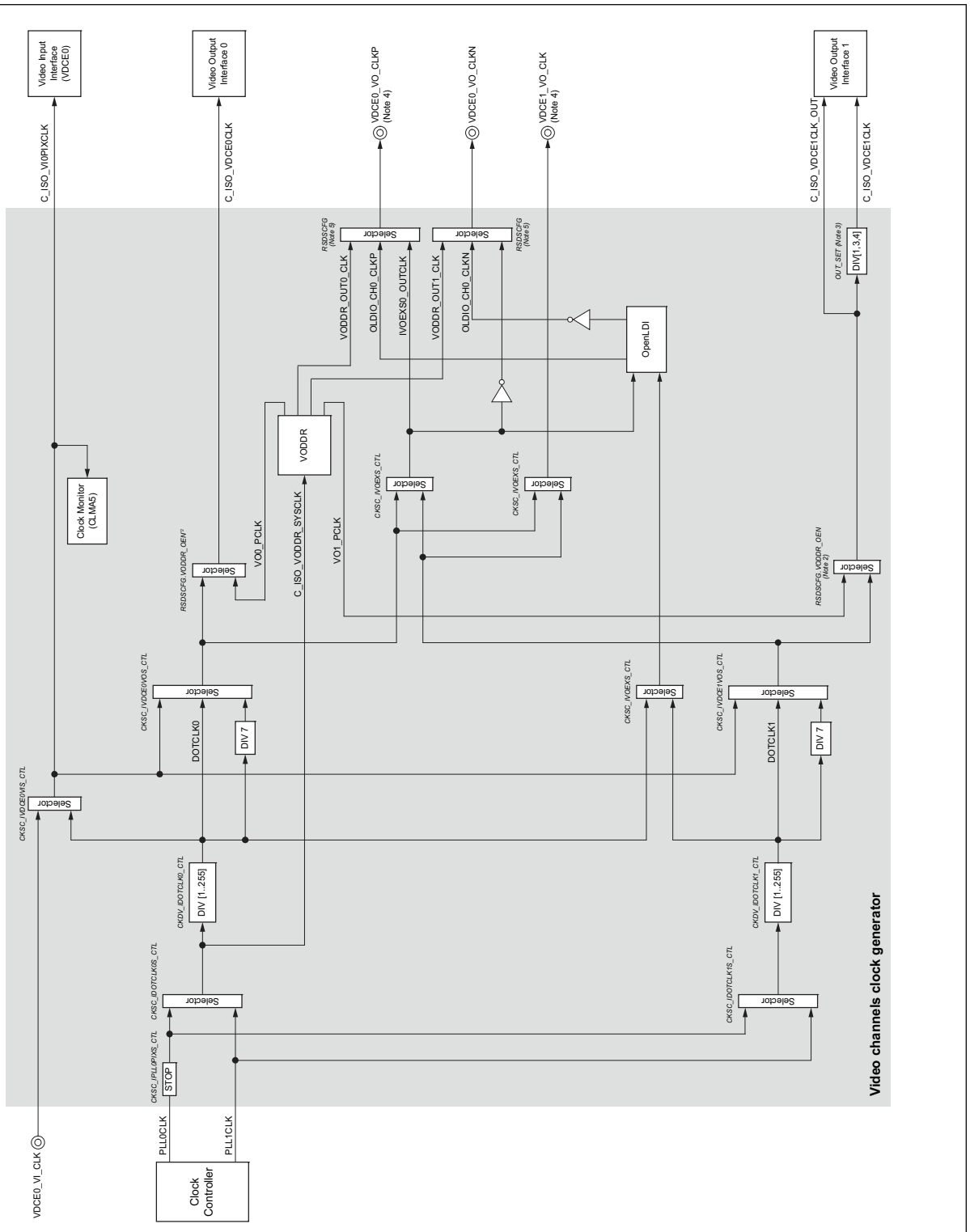
Note 1. The name of a selector register denotes the selected clock.
Note 2. Serial RGB is only available from VDCE1.

(2) External video output clocks selections

Table 37.2 External video output clocks selections

Output mode	VDCE0_VO_CLKP	VDCE1_VO_CLK
Parallel RGB	CKSC_IVDCE0VOS_CTL, (CKSC_IVDCE1VOS_CTL*2)*1	
Serial RGB	CKSC_IVDCE0VOS_CTL, (CKSC_IVDCE1VOS_CTL*2)*1	
VODDR	VODDR_OUT0_CLK	VODDR_OUT1_CLK
Open LDI	OpenLDIO_OUTCLK	–

Note 1. The name of a selector register denotes the selected clock.
Note 2. In case of CKSC_IVDCE0VOS_CTL = 1.



Note 1. The name of the clock control and selection registers are shown in *italics*.

Note 2. Refer to Section 37.10.1.2, RSDSCFG — RSDS and other video output control register.

Note 3. This selector is controlled by OUT_SET.OUT_FRQ_SEL bits of the respective Video Data Controller (VDCE). For details, refer to the OUT_SET register description in Section 38, Video Data Controller E (VDCE).

Note 4. Refer to Table 37.2, External video output clocks selections.

Note 5. Refer to Section 37.9.3, D1M1A Video output configuration.

Figure 37.13 D1M1A video channels clock generator

37.5.5 D1M1-V2 video channels clock generator

The pixel clock DOTCLK0 is generated by a divider [2..255] from one of the clocks from the clock generators (PLLs).

Several options are provided for C_ISO_VI0PIXCLK and C_ISO_VDCE0CLK, depending on the application scenario.

In addition, D1M1-V2 has options for Serial RGB output. The video clocks have to be set by following options.

Video input interface used

I.e. external video input clock VDCE0_VI_CLK is available:

- C_ISO_VI0PIXCLK = VDCE0_VI_CLK
- C_ISO_VDCE0CLK_OUT =
  - DOTCLK0: in case video input and output operating with different pixel clocks
  - C_ISO_VI0PIXCLK: in case video input and output operating with same pixel clock

Video input interface not used

I.e. external video input clock VDCE0_VI_CLK is not available:

- C_ISO_VI0PIXCLK = DOTCLK0
- C_ISO_VDCE0CLK_OUT = DOTCLK0

Video output interface clocks selections

The video output clock must be properly selected for the respective output mode.

Regarding the video output mode selection, refer to Section 37.9.4.1, D1M1-V2 video output mode selection.

(1) Video Output Interface 0 (VDCE0) input clock selections

Table 37.3 Video Output Interface n input clock selections

Output mode	C_ISO_VDCE0CLK	C_ISO_VDCE0CLK_OUT
Parallel RGB	CKSC_IVDCE0VOS_CTL*1	
Serial RGB	CKSC_IVDCE0VOS_CTL /[3 or 4]*1	CKSC_IVDCE0VOS_CTL*1

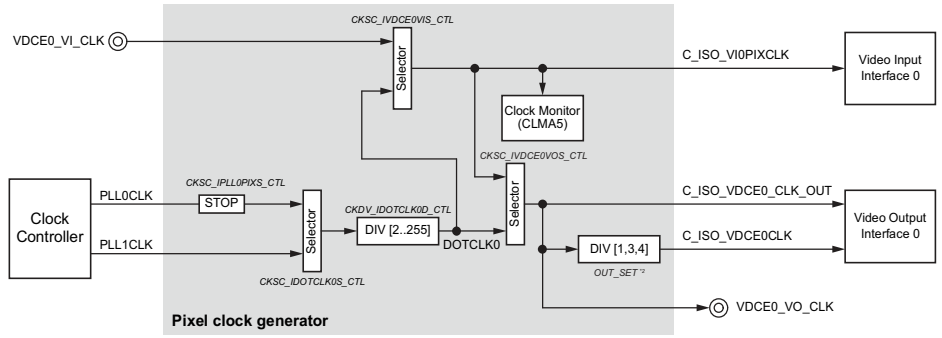
Note 1. The name of a selector register denotes the selected clock.

(2) External video output clock selections

Table 37.4 External video output clock selections

Output mode	VDCE0_VO_CLKP	VDCE0_VO_CLKN
Parallel RGB	CKSC_IVDCE0VOS_CTL*1	
Serial RGB	CKSC_IVDCE0VOS_CTL*1	

Note 1. The name of a selector register denotes the selected clock.



Note 1. The name of the clock control and selection registers are shown in *italics*.

Note 2. This selector is controlled by OUT_SET.OUT_FRQ_SEL bits of the respective Video Data Controller (VDCE).
For details, refer to the OUT_SET register description in Section 38, Video Data Controller E (VDCE).

Figure 37.14 D1M1-V2 video output pixel clock generation

37.5.6 D1M2(H) clock selections for different layer configurations

Depending on the types of layers, the data flow configuration and the usage of scalers certain settings need to be applied in order to supply all engaged modules with appropriate clocks.

The Section 37.3, Video input and output data flow shows diagrams for the various data flow configurations, which are referred to in the following table.

Table 37.5 Clock selections for different layer configurations

Layers		Clock supply for used scalers			
Types	**Configuration*4**	**Video input 0 downscaler**	**Image synthesizer 00 upscaler**	**Video input 1 downscaler**	**Image synthesizer 01 upscaler**
4 graphics	1	–	C_ISO_MIPIPIXCLK	–	–
	2				C_ISO_MIPIPIXCLK*2
1 video input + 3 graphics	1	C_ISO_MIPIPIXCLK or VDCE0_VI_CLK*1*3		–	–
	2				C_ISO_MIPIPIXCLK*2
2 video inputs + 2 graphics	–	C_ISO_MIPIPIXCLK or VDCE0_VI_CLK*1*3		VDCE1_VI_CLK*2*3	

Note 1. Selection by
CKSC_IVDCE0_VIS_CTL = 01_B: port VDCE0_VI_CLK
CKSC_IVDCE0_VIS_CTL = 10_B: C_ISO_MIPIPIXCLK

Note 2. Selection by
VDCECTL.VI1CTL = 0: port VDCE1_VI_CLK
VDCECTL.VI1CTL = 1: C_ISO_MIPIPIXCLK
The VDCECTL register is described in Section 37.10.1.1, VDCECTL — VDCE control register.

Note 3. The video input 0 or 1 downscaler and the image synthesizer 00 or 01 upscaler can not be used concurrently. Either the Downscaler or the Upscaler can be used at the same time.

Note 4. Refer to Section 37.3, Video input and output data flow for diagrams of the different configurations.

37.5.7 Video channels clock generators registers

The video channels clock generators are controlled and operated by the following registers:

Table 37.6 List of video channels clock generators registers

Register name	Shortcut	Address
PLL0PIXCLK clock control register	CKSC_IPLL0PIXS_CTL	FFF8 5640$_H$
PLL0PIXCLK clock active register	CKSC_IPLL0PIXS_ACT	FFF8 5648$_H$
PLL2INCLK clock divider register	CKDV_IPLL2IND_CTL	FFF8 5040$_H$
PLL2INCLK clock divider status register	CKDV_IPLL2IND_STAT	FFF8 5044$_H$
PLL2INCLK source clock selection register	CKSC_IPLL2INS_CTL	FFF8 5080$_H$
PLL2INCLK source clock active register	CKSC_IPLL2INS_ACT	FFF8 5088$_H$
PLL2CLK clock control register	CKSC_IPLL2CLKS_CTL	FFF8 5100$_H$
PLL2CLK clock active register	CKSC_IPLL2CLKS_ACT	FFF8 5108$_H$
DOTCLK0 source clock selection register	CKSC_IDOTCLK0S_CTL	FFF8 5940$_H$
DOTCLK0 source clock active register	CKSC_IDOTCLK0S_ACT	FFF8 5948$_H$
DOTCLK1 source clock selection register	CKSC_IDOTCLK1S_CTL	FFF8 5980$_H$
DOTCLK1 source clock active register	CKSC_IDOTCLK1S_ACT	FFF8 5988$_H$
DOTCLK0 clock divider register	CKDV_IDOTCLK0D_CTL	FFF8 59C0$_H$
DOTCLK0 clock divider status register	CKDV_IDOTCLK0D_STAT	FFF8 59C4$_H$
DOTCLK1 clock divider register	CKDV_IDOTCLK1D_CTL	FFF8 5A00$_H$
DOTCLK1 clock divider status register	CKDV_IDOTCLK1D_STAT	FFF8 5A04$_H$
VDCE0CLK source clock selection register	CKSC_IVDCE0VOS_CTL	FFF8 5880$_H$
VDCE0CLK source clock active register	CKSC_IVDCE0VOS_ACT	FFF8 5888$_H$
VDCE1CLK source clock selection register	CKSC_IVDCE1VOS_CTL	FFF8 58C0$_H$
VDCE1CLK source clock active register	CKSC_IVDCE1VOS_ACT	FFF8 58C8$_H$
Video output pixel clocks exchange register	CKSC_IVOEXS_CTL	FFF8 5900$_H$
Video output pixel clocks exchange active register	CKSC_IVOEXS_ACT	FFF8 5908$_H$
C_ISO_RSDSCLK source clock selection register	CKSC_IRSDSS_CTL	FFF8 5A40$_H$
C_ISO_RSDSCLK source clock active register	CKSC_IRSDSS_ACT	FFF8 5A48$_H$
C_ISO_VI0PIXCLK source clock selection register	CKSC_IVDCE0VIS_CTL	FFF8 5840$_H$
C_ISO_VI0PIXCLK source clock active register	CKSC_IVDCE0VIS_ACT	FFF8 5848$_H$
MIPIPLLCLK source clock selection register	CKSC_IMIPIPLLS_CTL	FFF8 5680$_H$
MIPIPLLCLK source clock active register	CKSC_IMIPIPLLS_ACT	FFF8 5688$_H$
C_ISO_MIPIPIXCLK clock divider register	CKSC_IMIPIPIXD_CTL	FFF8 56C0$_H$
C_ISO_MIPIPIXCLK clock divider active register	CKSC_IMIPIPIXD_ACT	FFF8 56C8$_H$

NOTE

The function of the PLL2 and its control registers are described in Section 12, Clock Controller.

NOTE

In the header files the module name of the above register is defined as:

SYS.

37.5.7.1 CKSC_IPLL0PIXS_CTL — PLL0PIXCLK clock control register (D1L2(H), D1M1(H), D1M1-V2, D1M2(H) and D1M1A only)

Writing to this register is protected by a special sequence of instructions by using the protection command register PROTCMDD1.

For details, refer to **Section 4, Write-Protected Registers**.

Access: This register can be read/written in 32-bit units.

Address: FFF8 5640$_H$

Initial value: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PLL0PIXSSTP
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Table 37.7 CKSC_IPLL0PIXS_CTL register contents

Bit Position	Bit Name	Function
31 to 1	Reserved	When written, write the initial value.
0	PLL0PIXSSTP	PLL0PIXCLK stop control 0: Stop PLL0PIXCLK. 1: Activate PLL0PIXCLK.

37.5.7.2 CKSC_IPLL0PIXS_ACT — PLL0PIXCLK clock active register (D1L2(H), D1M1(H), D1M1-V2, D1M2(H) and D1M1A only)

Access: This register can be read in 32-bit units.

Address: FFF8 5648$_H$

Initial value: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PLL0 PIXS ACT
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 37.8 CKSC_IPLL0PIXS_ACT register contents

Bit Position	Bit Name	Function
31 to 1	Reserved	When written, write the initial value.
0	PLL0PIXS ACT	PLL0PIXCLK status 0: PLL0PIXCLK is stopped (default). 1: PLL0PIXCLK i s active.

37.5.7.3 CKDV_IPLL2IND_CTL — PLL2INCLK clock divider register (D1M2(H) only)

Writing to this register is protected by a special sequence of instructions by using the protection command register PROTCMDD1.

For details, refer to **Section 4, Write-Protected Registers**.

Access: This register can be read/written in 32-bit units.

Address: FFF8 5040$_H$

Initial value: 0000 0018$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	PLL2INDCSID[4:0]				
Initial value	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W

Table 37.9 CKDV_IPLL2IND_CTL register contents

Bit Position	Bit Name	Function
31 to 5	Reserved	When written, write the initial value.
4 to 0	PLL2IND CSID[4:0]	Clock divider setting for PLL2INCLK 0: Disabled 12: PLL0CLK /12 ... 24: PLL0CLK /24 (default) All others: Setting prohibited

37.5.7.4 CKDV_IPLL2IND_STAT — PLL2INCLK clock divider status register (D1M2(H) only)

Access: This register can be read in 32-bit units.

Address: FFF8 5044$_H$

Initial value: 0000 0001$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PLL2IN ACT	PLL2IN SYNC
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 37.10 CKDV_IPLL2IND_STAT register contents

Bit Position	Bit Name	Function
31 to 2	Reserved	When written, write the initial value.
1	PLL2IN ACT	PLL2IN clock output active state 0: clock is inactive 1: clock is active
0	PLL2IN SYNC	PLL2INCLK clock divider state 0: output clock does not correspond to the current PLL2IND clock divider 1: output clock corresponds to the current PLL2IND clock divider

37.5.7.5 CKSC_IPLL2INS_CTL — PLL2INCLK source clock selection register (D1M2(H) only)

Writing to this register is protected by a special sequence of instructions by using the protection command register PROTCMDD1.

For details, refer to **Section 4, Write-Protected Registers**.

Access: This register can be read/written in 32-bit units.

Address: FFF8 5080$_H$

Initial value: 0000 0002$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PLL2INS CSID[1:0]	
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Table 37.11 CKSC_IPLL2INS_CTL register contents

Bit Position	Bit Name	Function
31 to 2	Reserved	When written, write the initial value.
1 to 0	PLL2INS CSID[1:0]	Source clock selection for PLL2INCLK 01_B: CKDV_IPLL2IND_CTL output clock 10_B: MainOsc f_X (default) All others: Setting prohibited

37.5.7.6 CKSC_IPLL2INS_ACT — PLL2INCLK source clock active register (D1M2(H) only)

Access: This register can be read in 32-bit units.

Address: FFF8 5088$_H$

Initial value: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PLL2INSACT[1:0]	
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 37.12 CKSC_IPLL2INS_ACT register contents

Bit Position	Bit Name	Function
31 to 2	Reserved	When written, write the initial value.
1 to 0	PLL2INS ACT[1:0]	Current active IPLL2INS source clock selection 0: selected clock inactive All others: selected and active clock

NOTE

As long as the clock, selected by the source clock selection register, is not active, this source clock active register remains 0.
Its shows the correct value, when the selected clock becomes active.

37.5.7.7 CKSC_IPLL2CLKS_CTL — PLL2CLK clock control register (D1M2(H) only)

Writing to this register is protected by a special sequence of instructions by using the protection command register PROTCMDD1.

For details, refer to **Section 4, Write-Protected Registers**.

Access: This register can be read/written in 32-bit units.

Address: FFF8 5100$_H$

Initial value: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PLL2CLKSSTP
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Table 37.13 CKSC_IPLL2CLKS_CTL register contents

Bit Position	Bit Name	Function
31 to 1	Reserved	When written, write the initial value.
0	PLL2CLKSSTP	PLL2CLK clock stop control 0: Stop PLL2CLK clock (default). 1: Activate PLL2CLK clock.

37.5.7.8 CKSC_IPLL2CLKS_ACT — PLL2CLK clock active register (D1M2(H) only)

Access: This register can be read in 32-bit units.

Address: FFF8 5108$_H$

Initial value: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PLL2CLKSACT
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 37.14 CKSC_IPLL2CLKS_ACT register contents

Bit Position	Bit Name	Function
31 to 1	Reserved	When written, write the initial value.
0	PLL2CLKSACT	PLL2CLK status 0: PLL2CLK clock is stopped (default). 1: PLL2CLK clock i s active.

37.5.7.9 CKSC_IDOTCLK0S_CTL — DOTCLK0 source clock selection register

Writing to this register is protected by a special sequence of instructions by using the protection command register PROTCMDD1.

For details, refer to **Section 4, Write-Protected Registers**.

Access: This register can be read/written in 32-bit units.

Address: FFF8 5940$_H$

Initial value: 0000 0001$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DOTCLK0S CSID[1:0]	
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Table 37.15 CKSC_IDOTCLK0S_CTL register contents

Bit Position	Bit Name	Function
31 to 2	Reserved	When written, write the initial value.
1 to 0	DOTCLK0S CSID[1:0]	Source clock selection for DOTCLK0 • For D1M2(H) devices: 00_B: Disabled 01_B: PLL0PIXCLK (default) 10_B: PLLFIXCLK 11_B: PLL2CLK • For all other devices: 00_B: Disabled 01_B: PLL0PIXCLK (default) 10_B: PLL1CLK 11_B: Setting prohibited

37.5.7.10 CKSC_IDOTCLK0S_ACT — DOTCLK0 source clock active register

Access: This register can be read in 32-bit units.

Address: FFF8 5948$_H$

Initial value: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DOTCLK0S ACT[1:0]	
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 37.16 CKSC_IDOTCLK0S_ACT register contents

Bit Position	Bit Name	Function
31 to 2	Reserved	When written, write the initial value.
1 to 0	DOTCLK0S ACT[1:0]	Current active IDOTCLK0S source clock selection 0: selected clock inactive All others: selected and active clock

NOTE

As long as the clock, selected by the source clock selection register, is not active, this source clock active register remains 0.
Its shows the correct value, when the selected clock becomes active.

37.5.7.11 CKSC_IDOTCLK1S_CTL — DOTCLK1 source clock selection register (D1M2(H), D1M1A only)

Writing to this register is protected by a special sequence of instructions by using the protection command register PROTCMDD1.

For details, refer to **Section 4, Write-Protected Registers**.

Access: This register can be read/written in 32-bit units.

Address: FFF8 5980$_H$

Initial value: 0000 0001$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DOTCLK1S CSID[1:0]	
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Table 37.17 CKSC_IDOTCLK1S_CTL register contents

Bit Position	Bit Name	Function
31 to 2	Reserved	When written, write the initial value.
1 to 0	DOTCLK1S CSID[1:0]	Source clock selection for DOTCLK1 00_B: Disabled 01_B: PLL0PIXCLK (default) 10_B: PLLFIXCLK 11_B: PLL2CLK (D1M2(H) only)

37.5.7.12 CKSC_IDOTCLK1S_ACT — DOTCLK1 source clock active register (D1M2(H), D1M1A only)

Access: This register can be read in 32-bit units.

Address: FFF8 5988$_H$

Initial value: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DOTCLK1S ACT[1:0]	
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 37.18 CKSC_IDOTCLK1S_ACT register contents

Bit Position	Bit Name	Function
31 to 2	Reserved	When written, write the initial value.
1 to 0	DOTCLK1S ACT[1:0]	Current active IDOTCLK1S source clock selection 0: selected clock inactive All others: selected and active clock

NOTE

As long as the clock, selected by the source clock selection register, is not active, this source clock active register remains 0.
Its shows the correct value, when the selected clock becomes active.

37.5.7.13 CKDV_IDOTCLK0D_CTL — DOTCLK0 clock divider register

Writing to this register is protected by a special sequence of instructions by using the protection command register PROTCMDD1.

For details, refer to **Section 4, Write-Protected Registers**.

Access: This register can be read/written in 32-bit units.

Address: FFF8 59C0$_H$

Initial value: 0000 0010$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	DOTCLK0DCSID[7:0]							
Initial value	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 37.19 CKSC_IDOTCLK0D_CTL register contents

Bit Position	Bit Name	Function
31 to 8	Reserved	When written, write the initial value.
7 to 0	DOTCLK0D CSID[7:0]	Clock divider setting for DOTCLK0 0: Disabled 1: CKSC_IDOTCLK0S_CTL selection /1 (D1M1A only) 2: CKSC_IDOTCLK0S_CTL selection /2 ... 16: CKSC_IDOTCLK0S_CTL selection /16 (default) ... 255: CKSC_IDOTCLK0S_CTL selection /255

37.5.7.14 CKDV_IDOTCLK0D_STAT — DOTCLK0 clock divider status register

Access: This register can be read in 32-bit units.

Address: FFF8 59C4$_H$

Initial value: 0000 0001$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DOTCLK0ACT	DOTCLK0SYNC
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 37.20 CKDV_IDOTCLK0D_STAT register contents

Bit Position	Bit Name	Function
31 to 2	Reserved	When written, write the initial value.
1	DOTCLK0 ACT	DOTCLK0 clock output active state 0: clock is inactive 1: clock is active
0	DOTCLK0 SYNC	DOTCLK0 clock divider state 0: output clock does not correspond to the current DOTCLK0D clock divider 1: output clock corresponds to the current DOTCLK0D clock divider

37.5.7.15 CKDV_IDOTCLK1D_CTL — DOTCLK1 clock divider register (D1M2(H), D1M1A only)

Writing to this register is protected by a special sequence of instructions by using the protection command register PROTCMDD1.

For details, refer to **Section 4, Write-Protected Registers**.

Access: This register can be read/written in 32-bit units.

Address: FFF8 5A00$_H$

Initial value: 0000 0010$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	DOTCLK1DCSID[7:0]							
Initial value	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 37.21 CKSC_IDOTCLK1D_CTL register contents

Bit Position	Bit Name	Function
31 to 8	Reserved	When written, write the initial value.
7 to 0	DOTCLK1D CSID[7:0]	Clock divider setting for DOTCLK1 0: Disabled 1: CKSC_IDOTCLK1S_CTL selection /1 (D1M1A only) 2: CKSC_IDOTCLK1S_CTL selection /2 ... 16: CKSC_IDOTCLK1S_CTL selection /16 (default) ... 255: CKSC_IDOTCLK1S_CTL selection /255

37.5.7.16 CKDV_IDOTCLK1D_STAT — DOTCLK1 clock divider status register (D1M2(H), D1M1A only)

Access: This register can be read in 32-bit units.

Address: FFF8 5A04$_H$

Initial value: 0000 0001$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DOT CLK1 ACT	DOT CLK1 SYNC
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 37.22 CKDV_IDOTCLK1D_STAT register contents

Bit Position	Bit Name	Function
31 to 2	Reserved	When written, write the initial value.
1	DOTCLK1 ACT	DOTCLK1 clock output active state 0: clock is inactive 1: clock is active
0	DOTCLK1 SYNC	DOTCLK1 clock divider state 0: output clock does not correspond to the current DOTCLK1D clock divider 1: output clock corresponds to the current DOTCLK1D clock divider

37.5.7.17 CKSC_IVDCE0VOS_CTL — C_ISO_VDCE0CLK source clock selection register (D1M1(H), D1M1-V2, D1M2(H) and D1M1A only)

Writing to this register is protected by a special sequence of instructions by using the protection command register PROTCMDD1.

For details, refer to **Section 4, Write-Protected Registers**.

Access: This register can be read/written in 32-bit units.

Address: FFF8 5880$_H$

Initial value: 0000 0001$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	VDCE0VOSCSID[2:0]		
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Table 37.23 CKSC_IVDCE0VOS_CTL register contents

Bit Position	Bit Name	Function
31 to 3	Reserved	When written, write the initial value.
2 to 0	VDCE0VOS CSID[2:0]	Source clock selection for C_ISO_VDCE0CLK 000_B: Disabled 001_B: DOTCLK0 (default) 010_B: DOTCLK0 /4 (D1M2(H) only) 011_B: Video Input interface 0 pixel clock C_ISO_V0PIXCLK 100_B: Video Input interface 1 pixel clock C_ISO_V1PIXCLK (D1M2(H) only) 101_B: DOTCLK0 /7 (D1M1A only) All others: Setting prohibited

CAUTION

For D1M2(H) only:

If the RSDS video output is used, set "VDCE0VOSCSID[2:0] = 010_B: DOTCLK0 /4".

For D1M1A only:

If the LVDS video output is used, set "VDCE0VOSCSID[2:0] = 101_B: DOTCLK0 /7".

37.5.7.18 CKSC_IVDCE0VOS_ACT — C_ISO_VDCE0CLK source clock active register (D1M1(H), D1M1-V2, D1M2(H) and D1M1A only)

Access: This register can be read in 32-bit units.

Address: FFF8 5888$_H$

Initial value: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	VDCE0VOSACT[2:0]		
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 37.24 CKSC_IVDCE0VOS_ACT register contents

Bit Position	Bit Name	Function
31 to 3	Reserved	When written, write the initial value.
2 to 0	VDCE0VOS ACT[2:0]	Current active IVDCE0VOS source clock selection 0: selected clock inactive All others: selected and active clock

NOTE

As long as the clock, selected by the source clock selection register, is not active, this source clock active register remains 0.
Its shows the correct value, when the selected clock becomes active.

37.5.7.19 CKSC_IVDCE1VOS_CTL — C_ISO_VDCE1CLK source clock selection register (D1M2(H), D1M1A only)

Writing to this register is protected by a special sequence of instructions by using the protection command register PROTCMDD1.

For details, refer to **Section 4, Write-Protected Registers**.

Access: This register can be read/written in 32-bit units.

Address: FFF8 58C0$_H$

Initial value: 0000 0001$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	VDCE1VOSCSID[2:0]		
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Table 37.25 CKSC_IVDCE1VOS_CTL register contents

Bit Position	Bit Name	Function
31 to 3	Reserved	When written, write the initial value.
2 to 0	VDCE1VOS CSID[2:0]	Source clock selection for C_ISO_VDCE1CLK 000_B: Disabled 001_B: DOTCLK1 (default) 010_B: DOTCLK1 /4 (D1M2(H) only) 011_B: Video Input interface 1 pixel clock C_ISO_V1PIXCLK (D1M2(H) only) 100_B: Video Input interface 0 pixel clock C_ISO_V0PIXCLK 101_B: DOTCLK1 /7 (D1M1A only) All others: Setting prohibited

CAUTION

For D1M2(H) only:
If the RSDS video output is used, set "VDCE1VOSCSID[2:0] = 010_B: DOTCLK1 /4".

For D1M1A only:
If the LVDS video output is used, set "VDCE1VOSCSID[2:0] = 101_B: DOTCLK1 /7".

37.5.7.20 CKSC_IVDCE1VOS_ACT — C_ISO_VDCE1CLK source clock active register (D1M2(H), D1M1A only)

Access: This register can be read in 32-bit units.

Address: FFF8 58C8$_H$

Initial value: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	VDCE1VOSACT[2:0]		
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 37.26 CKSC_IVDCE1VOS_ACT register contents

Bit Position	Bit Name	Function
31 to 3	Reserved	When written, write the initial value.
2 to 0	VDCE1VOS ACT[2:0]	Current active IVDCE1VOS source clock selection 0: selected clock inactive All others: selected and active clock

NOTE

As long as the clock, selected by the source clock selection register, is not active, this source clock active register remains 0.
Its shows the correct value, when the selected clock becomes active.

37.5.7.21 CKSC_IVOEXS_CTL — Video output pixel clocks exchange register (D1M2(H), D1M1A only)

Writing to this register is protected by a special sequence of instructions by using the protection command register PROTCMDD1.
For details, refer to **Section 4, Write-Protected Registers**.

Access: This register can be read/written in 32-bit units.
Address: FFF8 5900$_H$
Initial value: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOEXSCSID0
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Table 37.27 CKSC_IVOEXS_CTL register contents

Bit Position	Bit Name	Function
31 to 1	Reserved	When written, write the initial value.
0	VOEXS CSID0	For D1M2(H) Clock selection for VDCE0_VO_CLK and VDCE1_VO_CLK 0: Disable exchange (default) – VDCE0_VO_CLK = C_ISO_VDCE0CLK – VDCE1_VO_CLK = C_ISO_VDCE1CLK 1: Enable exchange – VDCE1_VO_CLK = C_ISO_VDCE0CLK – VDCE0_VO_CLK = C_ISO_VDCE1CLK For D1M1A: • OpenLDI use case 0: OLDICLK = DOTCLK0 1: OLDICLK = DOTCLK1 **Note:** OLDICLK is the operation clock for OpenLDI (maximum is 240 MHz). In the OpenLDI use case, the pixel clock for OpenLDI is also selected by CKSC_IVOEXS_CTL. For example, when CKSC_IVOEXS_CTL is 0, OLDICLK is DOTCLK0. And, pixel clock for OpenLDI is from CKSC_IVDCE0VOS_CTL. • Other use cases 0: Disable exchange – IVOEXS0_OUTCLK = DOTCLK0 – VDCE1_VO_CLK = DOTCLK1 1: Enable exchange – IVOEXS0_OUTCLK = DOTCLK1 – VDCE1_VO_CLK = DOTCLK0

37.5.7.22 CKSC_IVOEXS_ACT — Video output pixel clocks exchange active register (D1M2(H), D1M1A only)

Access: This register can be read in 32-bit units.

Address: FFF8 5908$_H$

Initial value: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VOEXSACT0
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 37.28 CKSC_IVOEXS_ACT register contents

Bit Position	Bit Name	Function
31 to 1	Reserved	When written, write the initial value.
0	VOEXS ACT0	Current active IVOEXS clock selection

37.5.7.23 CKSC_IRSDSS_CTL — C_ISO_RSDSCLK source clock selection register (D1M2(H) only)

Writing to this register is protected by a special sequence of instructions by using the protection command register PROTCMDD1.

For details, refer to **Section 4, Write-Protected Registers**.

Access: This register can be read/written in 32-bit units.

Address: FFF8 5A40$_H$

Initial value: 0000 0002$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RSDSSCSID[1:0]	
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Table 37.29 CKSC_IRSDSS_CTL register contents

Bit Position	Bit Name	Function
31 to 2	Reserved	When written, write the initial value.
1 to 0	RSDSS CSID[1:0]	Source clock selection for C_ISO_RSDSCLK 01_B: DOTCLK0 10_B: DOTCLK1 (default) All others: Setting prohibited

37.5.7.24 CKSC_IRSDSS_ACT — C_ISO_RSDSCLK source clock active register (D1M2(H) only)

Access: This register can be read in 32-bit units.

Address: FFF8 5A48$_H$

Initial value: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RSDSSACT[1:0]	
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 37.30 CKSC_IRSDSS_ACT register contents

Bit Position	Bit Name	Function
31 to 2	Reserved	When written, write the initial value.
1 to 0	RSDSS ACT[1:0]	Current active IRSDSS clock selection 0: selected clock inactive All others: selected and active clock

NOTE

As long as the clock, selected by the source clock selection register, is not active, this source clock active register remains 0.
Its shows the correct value, when the selected clock becomes active.

37.5.7.25 CKSC_IVDCE0VIS_CTL — C_ISO_VI0PIXCLK source clock selection register (D1M1(H), D1M1-V2, D1M2(H) and D1M1A only)

Writing to this register is protected by a special sequence of instructions by using the protection command register PROTCMDD1.

For details, refer to **Section 4, Write-Protected Registers**.

Access: This register can be read/written in 32-bit units.

Address: FFF8 5840$_H$

Initial value: 0000 0001$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VDCE0VIS CSID[1:0]	
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Table 37.31 CKSC_IVDCE0VIS_CTL register contents

Bit Position	Bit Name	Function
31 to 2	Reserved	When written, write the initial value.
1 to 0	VDCE0VIS CSID[1:0]	Source clock selection for C_ISO_VI0PIXCLK 01_B: Port VDCE0_VI_CLK (default) 10_B: C_ISO_MIPIPIXCLK (D1M2(H)) DOTCLK0 (D1M1(H), D1M1-V2 and D1M1A) All others: Setting prohibited

37.5.7.26 CKSC_IVDCE0VIS_ACT — C_ISO_VI0PIXCLK source clock active register (D1M1(H), D1M1-V2, D1M2(H) and D1M1A only)

Access: This register can be read in 32-bit units.

Address: FFF8 5848$_H$

Initial value: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VDCE0VIS ACT[1:0]	
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 37.32 CKSC_IVDCE0VIS_ACT register contents

Bit Position	Bit Name	Function
31 to 2	Reserved	When written, write the initial value.
1 to 0	VDCE0VIS ACT[1:0]	Current active IVDCE0VIS source clock selection 0: selected clock inactive All others: selected and active clock

NOTE

As long as the clock, selected by the source clock selection register, is not active, this source clock active register remains 0.
Its shows the correct value, when the selected clock becomes active.

37.5.7.27 CKSC_IMIPIPLLS_CTL — MIPIPLLCLK source clock selection register (D1M2(H) only)

Writing to this register is protected by a special sequence of instructions by using the protection command register PROTCMDD1.

For details, refer to **Section 4, Write-Protected Registers**.

Access: This register can be read/written in 32-bit units.

Address: FFF8 5680$_H$

Initial value: 0000 0001$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	MIPIPLLS CSID[1:0]	
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Table 37.33 CKSC_IMIPIPLLS_CTL register contents

Bit Position	Bit Name	Function
31 to 2	Reserved	When written, write the initial value.
1 to 0	MIPIPLLS CSID[1:0]	Source clock selection for MIPIPLLCLK 01_B: PLL0PIXCLK (default) 10_B: PLLFIXCLK All others: Setting prohibited

37.5.7.28 CKSC_IMIPIPLLS_ACT — MIPIPLLCLK source clock active register (D1M2(H) only)

Access: This register can be read in 32-bit units.

Address: FFF8 5688$_H$

Initial value: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	MIPIPLLS ACT[1:0]	
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 37.34 CKSC_IMIPIPLLS_ACT register contents

Bit Position	Bit Name	Function
31 to 2	Reserved	When written, write the initial value.
1 to 0	MIPIPLLS ACT[1:0]	Current active IMIPIPLLS source clock selection 0: selected clock inactive All others: selected and active clock

NOTE

As long as the clock, selected by the source clock selection register, is not active, this source clock active register remains 0.
Its shows the correct value, when the selected clock becomes active.

37.5.7.29 CKSC_IMIPIPIXD_CTL — C_ISO_MIPIPIXCLK clock divider register (D1M2(H) only)

Writing to this register is protected by a special sequence of instructions by using the protection command register PROTCMDD1.

For details, refer to **Section 4, Write-Protected Registers**.

Access: This register can be read/written in 32-bit units.

Address: FFF8 56C0$_H$

Initial value: 0000 0001$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	MIPIPIXD CSID[1:0]	
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Table 37.35 CKSC_IMIPIPIXD_CTL register contents

Bit Position	Bit Name	Function
31 to 2	Reserved	When written, write the initial value.
1 to 0	MIPIPIXD CSID[1:0]	Clock divider setting for C_ISO_MIPIPIXCLK 00_B: Disabled 01_B: MIPIPLLCLK /3 (default) 10_B: MIPIPLLCLK /6 11_B: MIPIPLLCLK /12

37.5.7.30 CKSC_IMIPIPIXD_ACT — C_ISO_MIPIPIXCLK clock divider active register (D1M2(H) only)

Access: This register can be read in 32-bit units.

Address: FFF8 56C8$_H$

Initial value: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	MIPIPIXD ACT[1:0]	
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 37.36 CKSC_IMIPIPIXD_ACT register contents

Bit Position	Bit Name	Function
31 to 2	Reserved	When written, write the initial value.
1 to 0	MIPIPIXD ACT[1:0]	Current active IMIPIPIXD clock divider 0: selected clock inactive All others: selected and active clock

NOTE

As long as the clock, selected by the source clock selection register, is not active, this source clock active register remains 0.
Its shows the correct value, when the selected clock becomes active.

37.5.8 Video Input Pixel Clock Monitoring

The video input pixel clocks are supervised by the Clock Monitors CLMA5 and CLMA6.

Table 37.37 Video Input pixel Clock Monitors

Module	Clock	Connected to
CLMA5:		
CLMATSMP	CLMA5 sampling clock	Low Speed IntOsc f_{RL} (240 kHz)
CLMATMON	CLMA5 monitored clock	C_ISO_VI0PIXCLK (Video Input Interface 0 pixel clock)
CLMA6 (D1M2(H) only):		
CLMATSMP	CLMA6 sampling clock	Low Speed IntOsc f_{RL} (240 kHz)
CLMATMON	CLMA6 monitored clock	C_ISO_VI1PIXCLK (Video Input Interface 1 pixel clock)

In case of a pixel clock fail the clock monitor generates an error signal, that is input to the Error Control Module.

For the functional description of the Clock Monitors CLMA5 and CLMA6 refer to **Section 12.7, Clock Monitor A (CLMA)**.

37.6 Sprite Engine registers update control

The sprite definition registers of the Sprite Engine are updated, i.e. new sprite definition values become effective, with internal vertical enable from VDCE and software control, depending which signal the update for a certain sprite is assigned to.
Refer to “Sprite definition registers modification” in Section 43, Sprite Engine (SPEA) for details.

The following diagram shows how the Sprite Engine VUPDATEn (n=0,1) inputs are generated.

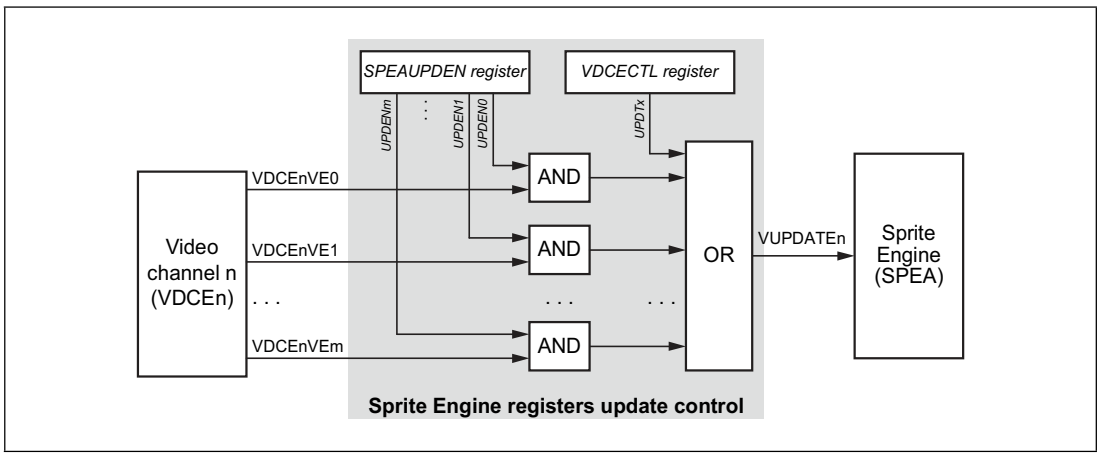


Figure 37.15 Sprite Engine registers update control for video channel n

Each stage of a video channel n issues vertical enable signals VDCEnVEm, which indicate active data processing of an image synthesizer stage.

These signals turn active with the first visible line of image data displayed by a layer and turns inactive after the last visible line has been displayed by that layer.

The Sprite Engine’s VUPDATEn input changes to low level when all VDCEnVEm signals turned to inactive (low level). This triggers the update of the sprite registers.

Each VDCEnVEm signal can be masked, i.e. disregarded, via mask bits in the Sprite Engine update timing control register SPEAUPDEN:

- UPDENm = 0: VDCEnVEm does not impact VUPDATEn signal generation
- UPDENm = 1: VDCEnVEm impacts VUPDATEn signal generation

The following table lists all available VDCEnVEm signals.

Table 37.38 VDCEnVEm signals (1/2)

VDCEn	VDCEnVEm signal		Mask bit
VDCE0	VDCE0VE0	VE of Image Synthesizer 00	SPEAUPDEN.UPDEN0
	VDCE0VE1	VE of Image Synthesizer 01	SPEAUPDEN.UPDEN1
	VDCE0VE2	VE of Image Synthesizer 02	SPEAUPDEN.UPDEN2
	VDCE0VE3	VE of Image Synthesizer 03	SPEAUPDEN.UPDEN3
	VDCE0VE4	VE of Output Image Generator	SPEAUPDEN.UPDEN4
	VDCE0VE5	VE after alpha blending of Image Synthesizer 02 and Image Synthesizer 03	SPEAUPDEN.UPDEN5

Table 37.38 VDCEnVEm signals (2/2)

VDCEn	VDCEnVEm signal		Mask bit
VDCE1 (D1M1A, D1M2(H) only)	VDCE1VE0	VE of Image Synthesizer 10	SPEAUPDEN.UPDEN8
	VDCE1VE1	VE of Image Synthesizer 11	SPEAUPDEN.UPDEN9
	VDCE1VE2	VE of Image Synthesizer 12	SPEAUPDEN.UPDEN10
	VDCE1VE3	VE of Image Synthesizer 13	SPEAUPDEN.UPDEN11
	VDCE1VE4	VE after alpha blending of Image Synthesizer 12 and Image Synthesizer 13	SPEAUPDEN.UPDEN12

Hardware and software update

By use of the VDCECTL.UPDTn bit and the mask bits SPEAUPDEN.UPDENm it can be selected whether the Sprite Engine registers are update by hardware, i.e. by the VDCEnVEm signals, or by the application program.

- Hardware update: VDCECTL.UPDTn = 0

  SPEAUPDEN.UPDENm = 1 selects the VDCEnVEm signals to generate VUPDATEn

- Software update: all SPEAUPDEN.UPDENm = 0

  The VSYNCn signal is directly controlled by the VDCECTL.UPDTn bit.
  Following procedure triggers the Sprite Engine registers update:

  – Set UPDTn = 1: VUPDATEn → high
  – Set UPDTn = 0: VUPDATEn → low: triggers Sprite Engine registers update.

37.7 Video Input selection

Depending on the data format of the video input signals and the ports, which are used to connect these, several control registers have to be set:

- correct alternative port function by use of the port control registers PFCn and PFCEn
- VDCE0: video input selection via VDCECTL.PXSL and VDCECTL.VISL0
- VDCE1: video input selection via VDCECTL.VISL1

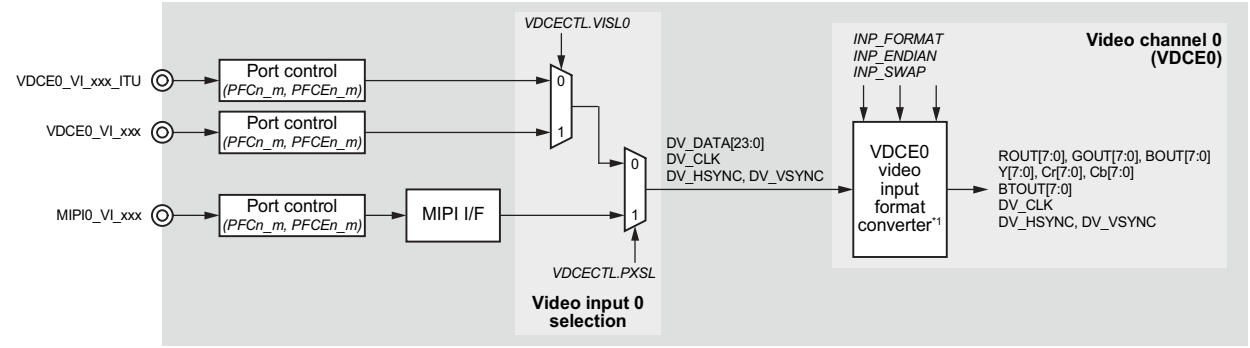
Refer to Section 2, Pins for detailed information about alternative port functions.

The tables in the following sections summarize the correct selection of the video input data format.

37.7.1 Video input channel 0 selection

Table 37.39 Video input channel 0 selection

Alternative port function	2nd alternative input (PFCn_m = 1, PFCEn_m = 0)	3rd alternative input (PFCn_m = 0, PFCEn_m = 1)		MIPI
Input ports	VDCE0_VI_xxx_ITU	VDCE0_VI_xxx_ITU	VDCE0_VI_xxx	MIPI0_VI_xxx
Data format	BT656, BT601	BT656, BT601	RGB666, RGB565, YCbCr422, BT656, BT601	RGB888, RGB565, YCbCr422
VISL0	0		1	–
PXSL	0			1



Note 1. The Video Data Controllers VDCEn include facilities to change the mapping of the color input data DV_DATA[23:0] to the VDCEn input data ROUT[7:0], GOUT[7:0], BOUT[7:0], BTOUT[7:0], Y[7:0], Cr[7:0] and Cb[7:0]. Refer to Section 38, Video Data Controller E (VDCE) for details about the INP_FORMAT, INP_ENDIAN and INP_SWAP settings.

Figure 37.16 Video channel 0 video input selections

The following table shows the assignment between the various video input signals for following settings:

- VDCEn video input format converter settings:
  - INP_EXT_SYNC_CNT.INP_ENDIAN_ON = 0
  - INP_EXT_SYNC_CNT.INP_SWAP_ON = 0

  For other bit allocations of DV_DATA[23:0] to ROUT[7:0], GOUT[7:0], BOUT[7:0], Y[7:0], Cr[7:0], Cb[7:0] and BTOUT[7:0] refer to Section 38, Video Data Controller E (VDCE).
- Video input selection:

– VDCECTL.PXSL = 0: input via VDCE0_VI signals (no MIPI)

Table 37.40 Video channel 0 input format selection examples

	VDCE0 video input format converter output			
	INP_FORMAT[2:0] = 001_B	INP_FORMAT[2:0] = 010_B	INP_FORMAT[2:0] = 011_B, 100_B	INP_FORMAT[2:0] = 101_B
Port signal name	**RGB666**	**RGB565**	**ITU formats**	**YCrCb422**
Color data signals				
VDCE0_VI_DATA17	ROUT[7]	–	–	–
VDCE0_VI_DATA16	ROUT[6]	–	–	–
VDCE0_VI_DATA15	ROUT[5]	ROUT[7]	–	Y[7]
VDCE0_VI_DATA14	ROUT[4]	ROUT[6]	–	Y[6]
VDCE0_VI_DATA13	ROUT[3]	ROUT[5]	–	Y[5]
VDCE0_VI_DATA12	ROUT[2]	ROUT[4]	–	Y[4]
VDCE0_VI_DATA11	GOUT[7]	ROUT[3]	–	Y[3]
VDCE0_VI_DATA10	GOUT[6]	GOUT[7]	–	Y[2]
VDCE0_VI_DATA9	GOUT[5]	GOUT[6]	–	Y[1]
VDCE0_VI_DATA8	GOUT[4]	GOUT[5]	–	Y[0]
VDCE0_VI_DATA7	GOUT[3]	GOUT[4]	BTOUT[7]	Cb[7] / Cr[7]
VDCE0_VI_DATA6	GOUT[2]	GOUT[3]	BTOUT[6]	Cb[6] / Cr[6]
VDCE0_VI_DATA5	BOUT[7]	GOUT[2]	BTOUT[5]	Cb[5] / Cr[5]
VDCE0_VI_DATA4	BOUT[6]	BOUT[7]	BTOUT[4]	Cb[4] / Cr[4]
VDCE0_VI_DATA3	BOUT[5]	BOUT[6]	BTOUT[3]	Cb[3] / Cr[3]
VDCE0_VI_DATA2	BOUT[4]	BOUT[5]	BTOUT[2]	Cb[2] / Cr[2]
VDCE0_VI_DATA1	BOUT[3]	BOUT[4]	BTOUT[1]	Cb[1] / Cr[1]
VDCE0_VI_DATA0	BOUT[2]	BOUT[3]	BTOUT[0]	Cb[0] / Cr[0]
Pixel clock and synchronization signals				
VDCE0_VI_CLK	DV_CLK			
VDCE0_VI_HSYNC	DV_HSYNC			
VDCE0_VI_VSYNC	DV_VSYNC			

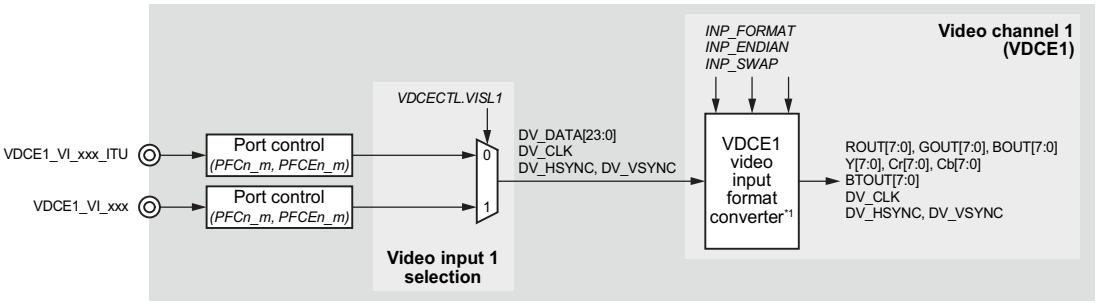
NOTE

If the MIPI interface is used for video data input to video channel 0 (VDCECTL.PXSL = 1) the separate MIPI ports MIPI0_VI_xxx are used. The mapping of the MIPI interface output to the VDCE0 video input format converter is fixed and occupies all 24 bit DV_DATA[23:0] for the RGB888 format.

37.7.2 Video input channel 1 selection (D1M2(H) only)

Table 37.41 Video input channel 1 selection

Alternative port function	2nd alternative input (PFCn_m = 1, PFCEn_m = 0)	1st alternative input (PFCn_m = 0, PFCEn_m = 0)
Input ports	VDCE1_VI_xxx_ITU	VDCE1_VI_xxx
Data format	BT656, BT601	RGB888, RGB666, RGB565, YCbCr444, YCbCr422, BT656, BT601
VISL1	0	1



Note 1. The Video Data Controllers VDCEn include facilities to change the mapping of the color input data DV_DATA[23:0] to the VDCEn input data ROUT[7:0], GOUT[7:0], BOUT[7:0], BTOUT[7:0], Y[7:0], Cr[7:0] and Cb[7:0]. Refer to Section 38, Video Data Controller E (VDCE) for details about the INP_FORMAT, INP_ENDIAN and INP_SWAP settings.

Figure 37.17 Video channel 1 video input selections

The following table shows the assignment between the various video input signals for following settings:

- VDCEn video input format converter settings:
  - INP_EXT_SYNC_CNT.INP_ENDIAN_ON = 0
  - INP_EXT_SYNC_CNT.INP_SWAP_ON = 0

  For other bit allocations of DV_DATA[23:0] to ROUT[7:0], GOUT[7:0], BOUT[7:0], Y[7:0], Cr[7:0], Cb[7:0] and BTOUT[7:0] refer to Section 38, Video Data Controller E (VDCE).

Table 37.42 Video channel 1 input format selection examples (1/2)

	VDCE1 video input format converter output				
	INP_FORMAT[2:0] = 000_B	INP_FORMAT[2:0] = 001_B	INP_FORMAT[2:0] = 010_B	INP_FORMAT[2:0] = 011_B, 100_B	INP_FORMAT[2:0] = 101_B
Port signal name	RGB888, YCrCb444	RGB666	RGB565	ITU formats	YCrCb422
Color data signals					
VDCE1_VI_DATA23	ROUT[7] / Cr[7]	–	–	–	–
VDCE1_VI_DATA22	ROUT[6] / Cr[6]	–	–	–	–
VDCE1_VI_DATA21	ROUT[5] / Cr[5]	–	–	–	–
VDCE1_VI_DATA20	ROUT[4] / Cr[4]	–	–	–	–
VDCE1_VI_DATA19	ROUT[3] / Cr[3]	–	–	–	–
VDCE1_VI_DATA18	ROUT[2] / Cr[2]	–	–	–	–
VDCE1_VI_DATA17	ROUT[1] / Cr[1]	ROUT[7]	–	–	–

Table 37.42 Video channel 1 input format selection examples (2/2)

	VDCE1 video input format converter output				
	INP_FORMAT[2:0] = 000_B	INP_FORMAT[2:0] = 001_B	INP_FORMAT[2:0] = 010_B	INP_FORMAT[2:0] = 011_B, 100_B	INP_FORMAT[2:0] = 101_B
Port signal name	RGB888, YCrCb444	RGB666	RGB565	ITU formats	YCrCb422
VDCE1_VI_DATA16	ROUT[0] / Cr[0]	ROUT[6]	–	–	–
VDCE1_VI_DATA15	GOUT[7] / Y[7]	ROUT[5]	ROUT[7]	–	Y[7]
VDCE1_VI_DATA14	GOUT[6] / Y[6]	ROUT[4]	ROUT[6]	–	Y[6]
VDCE1_VI_DATA13	GOUT[5] / Y[5]	ROUT[3]	ROUT[5]	–	Y[5]
VDCE1_VI_DATA12	GOUT[4] / Y[4]	ROUT[2]	ROUT[4]	–	Y[4]
VDCE1_VI_DATA11	GOUT[3] / Y[3]	GOUT[7]	ROUT[3]	–	Y[3]
VDCE1_VI_DATA10	GOUT[2] / Y[2]	GOUT[6]	GOUT[7]	–	Y[2]
VDCE1_VI_DATA9	GOUT[1] / Y[1]	GOUT[5]	GOUT[6]	–	Y[1]
VDCE1_VI_DATA8	GOUT[0] / Y[0]	GOUT[4]	GOUT[5]	–	Y[0]
VDCE1_VI_DATA7	BOUT[7] / Cb[7]	GOUT[3]	GOUT[4]	BTOUT[7]	Cb[7] / Cr[7]
VDCE1_VI_DATA6	BOUT[6] / Cb[6]	GOUT[2]	GOUT[3]	BTOUT[6]	Cb[6] / Cr[6]
VDCE1_VI_DATA5	BOUT[5] / Cb[5]	BOUT[7]	GOUT[2]	BTOUT[5]	Cb[5] / Cr[5]
VDCE1_VI_DATA4	BOUT[4] / Cb[4]	BOUT[6]	BOUT[7]	BTOUT[4]	Cb[4] / Cr[4]
VDCE1_VI_DATA3	BOUT[3] / Cb[3]	BOUT[5]	BOUT[6]	BTOUT[3]	Cb[3] / Cr[3]
VDCE1_VI_DATA2	BOUT[2] / Cb[2]	BOUT[4]	BOUT[5]	BTOUT[2]	Cb[2] / Cr[2]
VDCE1_VI_DATA1	BOUT[1] / Cb[1]	BOUT[3]	BOUT[4]	BTOUT[1]	Cb[1] / Cr[1]
VDCE1_VI_DATA0	BOUT[0] / Cb[0]	BOUT[2]	BOUT[3]	BTOUT[0]	Cb[0] / Cr[0]
Pixel clock and synchronization signals					
VDCE1_VI_CLK	DV_CLK				
VDCE1_VI_HSYNC	DV_HSYNC				
VDCE1_VI_VSYNC	DV_VSYNC				

37.8 D1M2(H) MIPI Video Input Interface (MIPI)

37.8.1 Overview of the RH850/D1L/D1M MIPI Video Input Interface

37.8.1.1 Number of Units

This microcontroller has the following number of units of the MIPI.

Each MIPI unit has one channel interface. “Number of channels” is used with the same meaning as “number of units” in this section.

Table 37.43 Number of Units

Product Name	D1L1	D1L2(H)	D1M1(H) D1M1-V2 D1M1A	D1M2(H)
Units	–	–	–	1
Names	–	–	–	MIPIn (n = 0)

Table 37.44 Index

Index	Meaning
n	Throughout this section, the individual MIPI units are identified by the index “n” (n = 0): for example, MIPInMODE is the MIPIn mode register.

37.8.1.2 Register Base Addresses

MIPI base addresses are listed in the following table.

MIPI register addresses are given as offsets from the base addresses in general.

Table 37.45 Register Base Address

Base Address Name	Base Address
<MIPI0_base>	FFFD 3400_H

37.8.1.3 Clock supply

The MIPI clock supply is shown in the following table.

Table 37.46 Clock Supply

Unit Name	Unit Clock Name	Supply Clock Name
MIPI0	MIPIPCLK	Clock Controller C_ISO_PCLK
	MIPIADCLK	Clock Controller C_ISO_MIPIADCLK
	MIPIPIXCLK	Clock Controller C_ISO_MIPIPIXCLK

37.8.1.4 Interrupts

MIPI interrupt requests are listed in the following table.

Table 37.47 MIPIn interrupt requests

MIPIn signals	Function	Connected to
MIPI0:		
	MIPI buffer overflow interrupt	Interrupt Controller INTMIPI0OVF
	MIPI controller interrupt	Interrupt Controller INTMIPI0CTL

37.8.1.5 Reset sources

MIPI reset sources are listed in the following table. MIPI is initialized by these reset sources.

Table 37.48 Reset sources

Unit Name	Reset Source
MIPI0	• Reset Controller SYSRES • reset upon wake-up from DEEPSTOP mode

37.8.1.6 External Input/Output Signals

External input/output signals of MIPI are listed below.

Table 37.49 I/O signals connections

Unit Signal Name	Outline	Alternative Port Pin Signal
MIPI0:		
MIPI0_VI_DATA0P	Data lane 0 signals	Port MIPI0_VI_DATA0P
MIPI0_VI_DATA0N		Port MIPI0_VI_DATA0N
MIPI0_VI_DATA1P	Data lane 1 signals	Port MIPI0_VI_DATA1P
MIPI0_VI_DATA1N		Port MIPI0_VI_DATA1N
MIPI0_VI_CLKP	Clock lane signals	Port MIPI0_VI_CLKP
MIPI0_VI_CLKN		Port MIPI0_VI_CLKN

37.8.2 Functional Overview

The MIPI Video Input Interface captures video data from a MIPI CSI-2 compliant Camera Serial Interface and passes the video data on to the Video Input Interface of the Video Data Controller.

Features summary

- Input and output color formats: 16-bit YUV422, 24-bit RGB888, 16-bit RGB565
- Maximum frame size: 1024 x 1024 pixels (24-bit color)
- Maximum frame rate: 60 frames/s
- MIPI CSI-2 receiver
- Lane module type:

  - MIPI CSI-2 receiver
  - The external MIPI CSI-2 master can send trigger events over the MIPI interface, which can issue interrupts.
- Data lane number: up to two lanes
- Data transfer bandwidth
  - Maximum data clock: max. 240 MHz in DDR mode
  - Transfer speed: max. 480 Mbit/s per lane, max. 960 Mbit/s in total
- Data consistency checks
  - ECC check on MIPI packet header (single bit errors correction, double bit errors detection)
  - CRC check on data packets
- Video output signal to the Video Data Controller
  - Regular video timing with HSYNC and VSYNC
  - Horizontal blank period: fixed to 16 pixel clock cycles
  - Burst output via two 4 KB output buffers
- Following MIPI CSI-2 features are not supported by this implementation of the MIPI Video Input Interface:
  - Virtual channel: This feature is not supported.
  - Embedded data: Transmitting embedded data to the Video Data Controller VDCE0 is not supported.
  - Low power data transfer (LPDT): This feature is not supported.

37.8.3 Functional Description

The following figure provides a functional block diagram of the MIPI Video Input Interface.

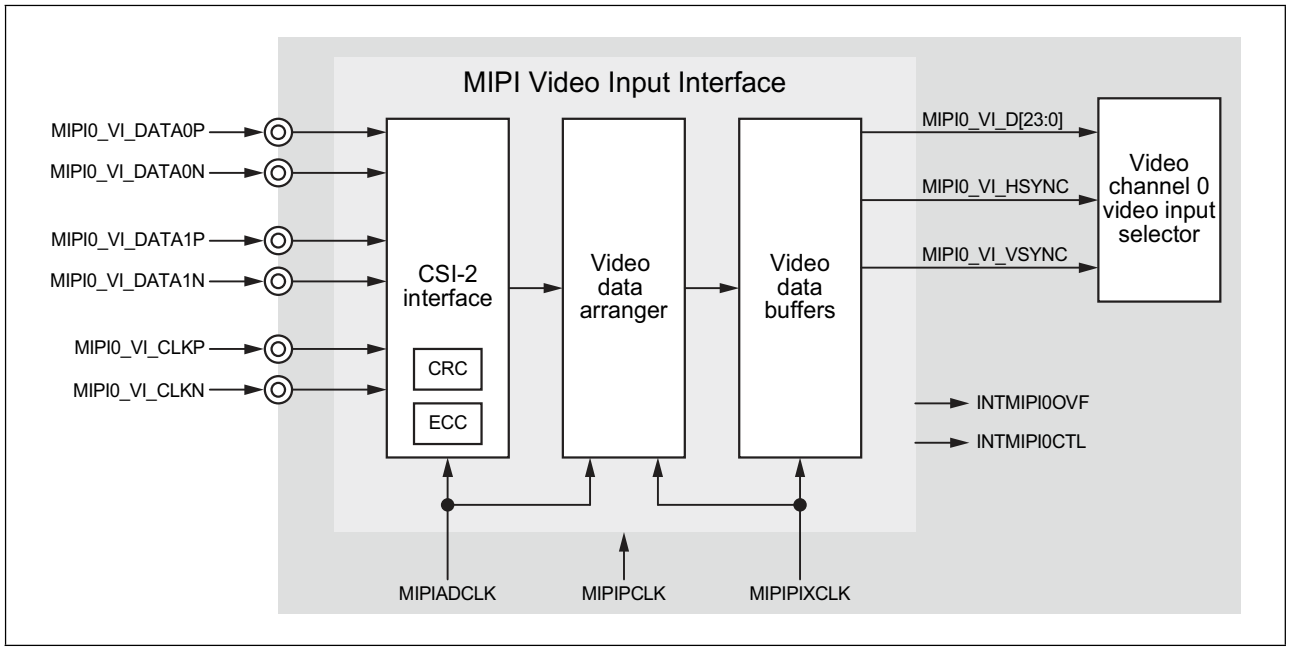


Figure 37.18 MIPI Video Input Interface block diagram

The video data, acquired via the CSI-2 interface, is arranged to be forwarded to the video input selector for the video channel 0 of the Video Data Controller.

The color format of the video data is passed through to the video channel 0, i.e. the color format is not changed.

The video signals (MIPI0_VI_D[23:0], MIPI0_VI_HSYNC and MIPI0_VI_VSYNC) are passed to the video input selector for the video channel 0 of the Video Data Controller via video data buffers.

The clock for the video channel 0 signals (MIPI0_VI_D[23:0], MIPI0_VI_HSYNC and MIPI0_VI_VSYNC) is the MIPIPIXCLK..
Refer to Section 37.7, Video Input selection for details about the video input selector.

37.8.4 MIPI Video Input Interface registers

The MIPI Video Input Interface is controlled and operated by the following registers:

For details on <MIPIn_base>, see Section 37.8.1.2, Register Base Addresses.

Table 37.50 MIPI register overview

Register name	Shortcut	Address
Operation control register	MIPInON	<MIPIn_base> + 00_H
Mode register	MIPInMODE	<MIPIn_base> + 04_H
Data delay register	MIPInDATA_DLY_CTL	<MIPIn_base> + 08_H
Reset register	MIPInRST_CTL	<MIPIn_base> + $0C_H$
SubLVDS buffer control	MIPInBUF_CTL	<MIPIn_base> + 10_H
$T_{CLK-SETTLE}$ and $T_{HS-SETTLE}$ register	MIPInSOT_COUNT	<MIPIn_base> + 14_H
PHY layer status register	MIPInRX_STATE	<MIPIn_base> + 18_H
Long packet word count definition register	MIPInWORD_COUNT	<MIPIn_base> + $1C_H$
LP buffer and noise removal timing register	MIPInLP_EN_ON_WC	<MIPIn_base> + 20_H
Line blanking period register	MIPInLINE_BLANK	<MIPIn_base> + 28_H
LP11 reset delay adjustment register	MIPInRESET_DLY_CTL0	<MIPIn_base> + $2C_H$
Interrupt status register	MIPInINTSTATUS	<MIPIn_base> + 40_H
Interrupt enable set register	MIPInINTENSET	<MIPIn_base> + 48_H
Interrupt enable clear register	MIPInINTENCLR	<MIPIn_base> + $4c_H$
Interrupt factor clear register	MIPInINTFFCLR	<MIPIn_base> + 50_H
T_{EOT} setting	MIPInEOT_COUNT	<MIPIn_base> + 74_H
Video input color format register	MIPInVIN_MODE	<MIPIn_base> + $9C_H$

37.8.4.1 MIPInON – Operation control register

This register controls start and stop of the MIPI interface.

Access: This register can be read/written in 32-bit units.

Address: <MIPIn_base> + 00_H

Initial value: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	MIPI_ON
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Table 37.51 MIPInON register contents

Bit position	Bit name	Function
31 to 1	Reserved	These bits are always read as 0. When written, write the initial value.
0	MIPI_ON	MIPI interface start/stop control 0: Stop the MIPI interface immediately 1: Start the MIPI interface immediately

37.8.4.2 MIPInMODE – Mode register

This register controls the mode of the MIPI interface.

Access: This register can be read/written in 32-bit units.

Address: <MIPIn_base> + 04_H

Initial value: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	0	0	DEL_LP11_NOISE	LANE_NUM[1:0]		0	SEL_PLIF	RSTN_CLK_ON	RSTN_DATA_ON	SOT_CLKSEL	0	0	0	0	DATA_TYPE[1:0]	
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	0	0	0	0	0	0	0	0	0	0	0	VTIM	0	0	ECC_EN	CRC_EN
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R/W	R/W

Table 37.52 MIPInMODE register contents (1/2)

Bit position	Bit name	Function
31 to 30	Reserved	These bits are always read as 0. When written, write the initial value.
29	DEL_LP11_NOISE	Noise filter for clock line. If this is switched on, noise occurring during transition from HS state to LP11 state is removed. 0: Noise removal disabled 1: Noise removal enabled
28 to 27	LANE_NUM[1:0]	Number of lanes. 00_B: 1 lane 01_B: 2 lanes All others: Setting prohibited
26	Reserved	These bits are always read as 0. When written, write the initial value.
25	SEL_PLIF	Select PHY layer interface synchronization method 0: Select FIFO method 1: Setting prohibited
24	RSTN_CLK_ON	Reset enable for clock lane for $T_{CLK\text{-}SETTLE}$ and T_{EOT} When this bit is 1(enable), the reset of clock lane is controlled after $T_{CLK\text{-}SETTLE}$ and T_{EOT}. This reset is available for noise removal from HS buffer during LP mode. This bit must be set to 1 before the MIPI transmission is enabled by MIPInON.MIPI_ON = 1. 0: Disable (prohibited when MIPInON.MIPI_ON = 1) Clock lane is not controlled after $T_{CLK\text{-}SETTLE}$ and T_{EOT}. Also, clock lane is not reset during LP mode. 1: Enable The reset of clock lane is released after $T_{CLK\text{-}SETTLE}$. $T_{CLK\text{-}SETTLE}$ is defined in MIPInSOT_COUNT.CLK_COUNT[7:0]. The reset of clock lane is activated after T_{EOT}. T_{EOT} is defined in MIPInEOT_COUNT.EOT_COUNT[7:0].

Table 37.52 MIPInMODE register contents (2/2)

Bit position	Bit name	Function
23	RSTN_ DATA_ON	Reset enable for data lane for $T_{HS-SETTLE}$ and T_{EOT} When this bit is 1(enable), the reset of data lane is controlled after $T_{HS-SETTLE}$ and T_{EOT}. This reset is available for noise removal from HS buffer during LP mode. This bit must be set to 1 before the MIPI transmission is enabled by MIPInON.MIPI_ON = 1. 0: Disable (prohibited when MIPInON.MIPI_ON = 1) Data lane is not controlled after $T_{HS-SETTLE}$ and T_{EOT}. Also, data lane is not reset during LP mode. 1: Enable The reset of data lane is released after $T_{HS-SETTLE}$. $T_{HS-SETTLE}$ is defined in MIPInSOT_COUNT.DATA_COUNT[7:0]. The reset of data lane is activated after T_{EOT}. T_{EOT} is defined in MIPInEOT_COUNT.EOT_COUNT[7:0].
22	SOT_CLKSEL	Select the clock that counts $T_{HS-SETTLE}$ 0: MIPIADCLK 1: External clock divided by 4
21 to 18	Reserved	These bits are always read as 0. When written, write the initial value.
17 to 16	DATA_ TYPE[1:0]	Input format specification 10_B: RAW8 All others: Setting prohibited
15 to 5	Reserved	These bits are always read as 0. When written, write the initial value.
4	VTIM	Timing of MIPI0_VI_VSYNC changes for using VDCE vertical position shifting 0: Enabled 1: Disabled
3 to 2	Reserved	This bit is always read as 0. When written, write the initial value.
1	ECC_EN	ECC check and correction of single bit errors of the header 0: Disabled 1: Enabled
0	CRC_EN	CRC check of the packet data 0: Disabled 1: Enabled

37.8.4.3 MIPInDATA_DLY_CTL – Data delay register

The register is for input data delay setting.

Access: This register can be read/written in 32-bit units.

Address: <MIPIn_base> + 08_H

Initial value: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	0	0	0	0	0	0	0	0	0	0	DLY_LANE1[2:0]			DLY_LANE0[2:0]		
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Table 37.53 MIPInDATA_DLY_CTL register contents

Bit position	Bit name	Function
31 to 6	Reserved	These bits are always read as 0. When written, write the initial value.
5 to 3	DLY_ LANE1[2:0]	Delay adjustment for data lane 1 Delay time = DLY_LANE1[2:0] x 100 ps
2 to 0	DLY_ LANE0[2:0]	Delay adjustment for data lane 0 Delay time = DLY_LANE0[2:0] x 100 ps

37.8.4.4 MIPInRST_CTL – Reset register

This register is used to reset the MIPI interface.

The reset must be released only if MIPI receiver is in STOP state.
So, please set the MIPI transceiver to STOP state (LP11) before releasing MIPI reset.
Thus follow the procedure below to reset the MIPI interface:

- Write MIPInRST_CTL = 0000 0003_H
- Set MIPI transceiver to STOP state (LP11)
- Write MIPInRST_CTL = 0000 0000_H

NOTE

In order to reduce power consumption it is recommended to set the MIPI interface into reset if it is not used.

Access: This register can be read/written in 32-bit units.

Address: <MIPIn_base> + $0C_H$

Initial value: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	OTHER_RST	PHY_RST
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Table 37.54 MIPInRST_CTL register contents

Bit position	Bit name	Function
31 to 2	Reserved	These bits are always read as 0. When written, write the initial value.
1	OTHER_RST	Reset MIPI interface, except PHY layer and PHY layer interface 0: Release reset 1: Reset
0	PHY_RST	Reset PHY layer and PHY layer interface 0: Release reset 1: Reset

CAUTION

All parts of the MIPI Interface must be reset or released from reset at the same time.

Thus set

- **MIPInRST_CTL = 0000 0003_H to set the MIPI Interface in reset**
- **MIPInRST_CTL = 0000 0000_H to release the MIPI Interface from reset**

37.8.4.5 MIPInBUF_CTL – SubLVDS buffer control

This register is used to control the lane buffers.

Access: This register can be read/written in 32-bit units.

Address: <MIPIn_base> + 10_H

Initial value: 0000 $7FFF_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MIPInBUF_CTL[31:16]															
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MIPInBUF_CTL[15:0]															
Initial value	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 37.55 MIPInBUF_CTL register contents

Bit position	Bit name	Function
31 to 0	MIPInBUF _CTL[31:0]	Lane buffer control • Set this register to 0000 0000_H if the MIPI interface is used. • Set this register to its initial value 0000 $7FFF_H$ if the MIPI interface is not used.

37.8.4.6 MIPInSOT_COUNT – $T_{CLK\text{-}SETTLE}$ and $T_{HS\text{-}SETTLE}$ register

This register determines the duration of $T_{CLK\text{-}SETTLE}$ and $T_{HS\text{-}SETTLE}$.

At the beginning of the HS transmission the LP receive buffers are switched off and the HS receive buffers and the line termination is switched on.

To mask possible analog effects that could occur during the switching a time is defined where the output from the HS receiver is ignored. Thus the time bridges the time until the switch from LP to HS has been settled.

For the data lane this time is $T_{HS\text{-}SETTLE}$ and is defined by MIPInSOT_COUNT.DATA_COUNT[7:0]. For the clock lane this time is $T_{CLK\text{-}SETTLE}$ and is defined by MIPInSOT_COUNT.CLK_COUNT[7:0].

(1) CLK_COUNT[7:0]

CLK_COUNT[7:0] must satisfy the following condition:

$$(95\ ns / T_{MIPIADCLK}) - 2 \leq CLK\_COUNT[7:0] \leq (300\ ns / T_{MIPIADCLK}) - 3$$

with

$T_{MIPIADCLK}$ $= 1/f_{MIPIADCLK}$ = cycle duration of the MIPIADCLK in ns

Example

for $f_{MIPIADCLK}$ = 160 MHz, i.e. $T_{MIPIADCLK}$ = 6.25 ns

$$(95\ ns / 6.25\ ns) - 2 \leq CLK\_COUNT[7:0] \leq (300\ ns / 6.25\ ns) - 3$$

$$13.2 \leq CLK\_COUNT[7:0] \leq 45$$

Thus CLK_COUNT[7:0] must be set between 14 and 45.

(2) DATA_COUNT[7:0]

DATA_COUNT[7:0] must satisfy the following condition

$$[(85\ ns + 6\ UI) / T_{HS\_SET\_CLK}] - 2 \leq DATA\_COUNT[7:0] \leq [(145\ ns + 10\ UI) / T_{HS\_SET\_CLK}] - 3$$

with

$T_{HS\_SET\_CLK}$ Count clock for T_{HS_SETTLE}, which is selected by MIPInMODE.SOT_CLKSEL:

- SOT_CLKSEL = 0: count clock is MIPIADCLK
  $T_{HS\_SET\_CLK} = T_{MIPIADCLK} = 1/f_{MIPI\_VI\_CLK}$
  = cycle duration of the MIPIADCLK in ns
- SOT_CLKSEL = 1: count clock is the MIPI interface input clock MIPI_VI_CLK/4
  $T_{HS\_SET\_CLK} = 4 \times T_{MIPI\_VI\_CLK} = 4/f_{MIPIADCLK}$
  = cycle duration of the MIPI_VI_CLK/4 in ns

UI Unit Interval, equal to the duration of any HS state on the clock lane, which is 1/2 clock cycle duration of the MIPI interface input clock MIPI_VI_CLK.

- $UI = T_{MIPI\_VI\_CLK}/2 = 2 \times f_{MIPI\_VI\_CLK}$
  = cycle duration of the 2 × MIPI_VI_CLK in ns

Example 1

for

- SOT_CLKSEL = 0
- $f_{MIPIADCLK}$ = 160 MHz → $T_{HS\_SET\_CLK}$ = 6.25 ns
- MIPI_VI_CLK = 240 MHz → UI = $T_{MIPI\_VI\_CLK}/2$ = 2.08 ns)

$$[(85\ ns + 6 \times 2.08\ ns) / 6.25\ ns] - 2 \leq DATA\_COUNT[7:0] \leq [(145\ ns + 10 \times 2.08\ ns) / 6.25] - 3$$

$$13.6 \leq DATA\_COUNT[7:0] \leq 23.5$$

Thus DATA_COUNT[7:0] must be set between 14 and 23.

Example 2

for

- SOT_CLKSEL = 1
- MIPI_VI_CLK = 240 MHz → $T_{HS\_SET\_CLK} = 4 \times T_{MIPI\_VI\_CLK}$ = 16.67 ns
- MIPI_VI_CLK = 240 MHz → UI = $T_{MIPI\_VI\_CLK}/2$ = 2.08 ns)

$$[(85\ ns + 6 \times 2.08\ ns) / 16.67\ ns] - 2 \leq DATA\_COUNT[7:0] \leq [(145\ ns + 10 \times 2.08\ ns) / 6.25] - 3$$

$$3.85 \leq DATA\_COUNT[7:0] \leq 6.95$$

Thus DATA_COUNT[7:0] must be set between 4 and 6.

Access: This register can be read/written in 32-bit units.

Address: <MIPIn_base> + 14_H

Initial value: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	0	0	0	0	0	0	0	0	DATA_COUNT[7:0]							
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	0	0	0	0	0	0	0	0	CLK_COUNT[7:0]							
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 37.56 MIPInSOT_COUNT register contents

Bit position	Bit name	Function
31 to 24	Reserved	These bits are always read as 0. When written, write the initial value.
23 to 16	DATA_ COUNT[7:0]	$T_{HS-SETTLE}$ setting of data lane
15 to 8	Reserved	These bits are always read as 0. When written, write the initial value.
7 to 0	CLK_ COUNT[7:0]	$T_{CLK-SETTLE}$ setting of clock lane

37.8.4.7 MIPInRX_STATE – PHY layer status register

This register indicates the STOP state of the PHY.

PHY STOP state is indicated by

RX_STOP_STATE_0 = RX_STOP_STATE_1 = RX_STOP_STATE_C = 1.

Verification of the STOP state is used to check release timing for MIPInRST_CTL.
Refer to Section 37.8.4.4, MIPInRST_CTL – Reset register for details.

Access: This register can be read in 32-bit units.

Address: <MIPIn_base> + 18_H

Initial value: 0000 0492_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Bit17	RX_STOP_STATE_C
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	0	0	0	0	Bit11	Bit10	0	Bit8	Bit7	0	Bit5	RX_STOP_STATE_1	0	Bit2	RX_STOP_STATE_0	0
Initial value	0	0	0	0	0	1	0	0	1	0	0	1	0	0	1	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 37.57 MIPInRX_STATE register contents

Bit position	Bit name	Function
31 to 18	Reserved	These bits are always read as 0.
17	Bit17	The read value of this bit is undefined.
16	RX_STOP_STATE_C	This bit indicates the clock lane status of PHY layer. 0: Not STOP 1: STOP **Note:** This status can not be confirmed when MIPInBUF_CTL = 0000 $7FFF_H$ or MIPInRST_CTL = 0000 0003_H.
15 to 12	Reserved	These bits are always read as 0.
11, 10	Bit11, Bit10	The read values of these bits are undefined.
9	Reserved	These bits are always read as 0.
8, 7	Bit8, Bit7	The read values of these bits are undefined.
6	Reserved	This bit is always read as 0.
5	Bit5	The read value of this bit is undefined.
4	RX_STOP_STATE_1	This bit indicates the data lane 1 status of PHY layer. 0: Not STOP 1: STOP
3	Reserved	This bit is always read as 0.
2	Bit2	The read value of this bit is undefined.
1	RX_STOP_STATE_0	This bit indicates the data lane 0 status of PHY layer. 0: Not STOP 1: STOP
0	Reserved	This bit is always read as 0. When written, write the initial value.

37.8.4.8 MIPInWORD_COUNT – Long packet word count definition register

This register allows to define the maximum size of long packets to be transferred to the Video Data Controller.

If enabled (FORCE_WC_EN = 1) the maximum packet word count of Long Packet will be fixed to FORCE_WC_NUM[15:0].
If packet word count is larger than FORCE_WC_NUM[15:0], FORCE_WC_NUM[15:0] size will be sent to Video Data Controller VDCE0.
In such case a CRC error will be generated. Therefore CRC check should be disabled (MIPInMODE.CRC_EN = 0), when FORCE_WC_EN = 1.

NOTE

By shortening Long Packets, while the image is sent to the Video Data Controller, the right side of the image can be cut.
Alternatively the image can also be cut by the Video Data Controller.

Access: This register can be read/written in 32-bit units.

Address: <MIPIn_base> + $1C_H$

Initial value: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	FORCE_WC_EN
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	FORCE_WC_NUM[15:0]															
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 37.58 MIPInWORD_COUNT register contents

Bit position	Bit name	Function
31 to 17	Reserved	These bits are always read as 0. When written, write the initial value.
16	FORCE_WC_EN	Enable Long Packet size limitation 0: Disable 1: Enable
15 to 0	FORCE_WC_NUM[15:0]	Word count value of Long Packet, if enabled by FORCE_WC_EN = 1. One word has 8 bit.

37.8.4.9 MIPInLP_EN_ON_WC – LP buffer and noise removal timing register

This register adjusts the timing and the width of noise removal when LP buffer is turned on.

LP buffer is turned on again after high speed transmission has ended.
Each high speed packet contains a word count value that indicates the length of the packet.

Switching on of LP buffer can be adjusted absolute to the beginning of the packet (LP_EN_ON_MODE = 1) or relative to the end of the packet (LP_EN_ON_MODE = 0). The adjustment value is given by LP_EN_ON_WC[15:0].

Noise reduction is then defined in an interval centered around this switching point. The width of the interval is given by 2* LP_DEL_NOISE_TIM[6:0].

Access: This register can be read/written in 32-bit units.

Address: <MIPIn_base> + 20_H

Initial value: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	0	0	0	0	0	0	0	LP_DEL_NOISE_MODE	LP_DEL_NOISE_TIM[6:0]							LP_EN_ON_MODE
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LP_EN_ON_WC[15:0]															
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 37.59 MIPInLP_EN_ON_WC register contents

Bit position	Bit name	Function
31 to 25	Reserved	These bits are always read as 0. When written, write the initial value.
24	LP_DEL_ NOISE_MODE	Enable noise removal mode. 1: Noise removal enabled 0: Noise removal disabled
23 to 17	LP_DEL_ NOISE_ TIM[6:0]	The count value for noise removal when LP_DEL_NOISE_MODE = 1. The recommended value of LP_DEL_NOISE_TIM[6:0] is 05_H.
16	LP_EN_ON_ MODE	Setting LP buffer on timing 0: LP buffer turns on when word count reached the calculated value (Word count in packet header) -LP_EN_ON_WC[15:0]. 1: LP buffer turns on after LP_EN_ON_WC[15:0] read words.
15 to 0	LP_EN_ON_ WC[15:0]	Set the word count (WC) when to (IO_MIPI_LP_EN_ * _OUT == 1) switch the LP BUFFER on. The recommended value of LP_EN_ON_WC[15:0] is 0020_H.

The noise reduction interval must not leave the data interval of the packet.
Thus LP_EN_ON_WC[15:0] and LP_DEL_NOISE_TIM[6:0] must comply to the following conditions:

- If noise removal enabled (LP_EN_ON_MODE = 1):

  - LP_EN_ON_WC[15:0] + LP_DEL_NOISE_TIM[6:0] + 16 < (wordcount in packet header)
  - LP_EN_ON_WC[15:0] – LP_DEL_NOISE_TIM[6:0] > 0
- If noise removal disabled (LP_EN_ON_MODE = 0):
  - LP_EN_ON_WC[15:0] – LP_DEL_NOISE_TIM[6:0] - 16 > 0
  - LP_EN_ON_WC[15:0] + LP_DEL_NOISE_TIM[6:0] < (wordcount in packet header)

NOTE

The word count in packet header is changed to MIPInWORD_COUNT.FORCE_WC_NUM[15:0], if Long Packet size limitation is enabled (MIPInWORD_COUNT.FORCE_WC_EN = 1).

37.8.4.10 MIPInLINE_BLANK – Line blanking period register

This register generates EOT (End of Transmission) timing.

MIPInLINE_BLANK controls the duration between the current line data (current Long Packet) and the next line data (next Long Packet).
This control is not related to MIPInWORD_COUNT control.

EOT timing is used in the following cases

- Initialization timing after a line data is transmitted.
- When MIPI mode moves from HS to LP under the condition that the number of words is less than WORD COUNT, EOT timing can be used to adjust the timing LP buffer turn on.

Access: This register can be read/written in 32-bit units.

Address: <MIPIn_base> + 28_H

Initial value: 0000 0004_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

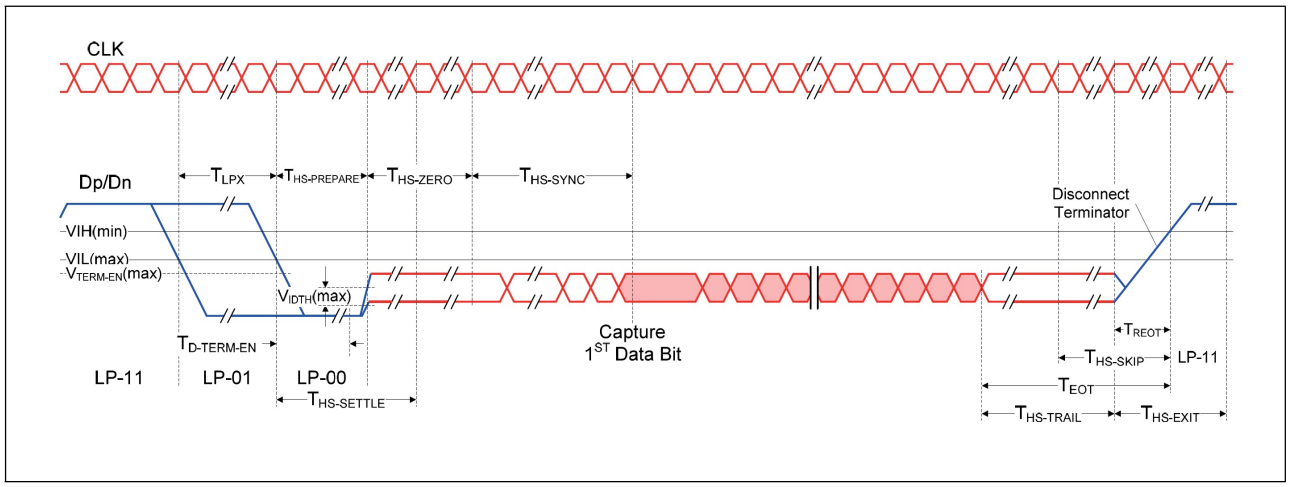
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	0	0	0	0	0	0	0	0	LINE_BLANK[7:0]							
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 37.60 MIPInLINE_BLANK register contents

Bit position	Bit name	Function
31 to 8	Reserved	These bits are always read as 0. When written, write the initial value.
7 to 0	LINE_ BLANK[7:0]	Interval until EOT in MIPIADCLK cycles

LINE_BLANK[7:0] must stay smaller than the minimum time until next HS transfer.
Thus it must satisfy the following condition:

- 12 * UI < MIPIADCLK cycles * (LINE_BLANK[7:0] + 2) < $T_{HS-EXIT}$ + T_{LPX} + $T_{HS-PREPARE}$ + $T_{HS-ZERO}$ − 12 * MIPIADCLK cycles
  UI means clock cycle / 2 in the HS mode (1 Data Bit Time).
- The setting of MIPInLINE_BLANK = 0 is prohibit.


CLK
Dp/Dn
VIH(min)
VIL(max)
VTERM-EN(max)
TLPX
THS-PREPARE
THS-ZERO
THS-SYNC
VIDTH(max)
TD-TERM-EN
LP-11
LP-01
LP-00
THS-SETTLE
Capture
1ST Data Bit
Disconnect
Terminator
TREOT
THS-SKIP
LP-11
TEOT
THS-TRAIL
THS-EXIT

37.8.4.11 MIPInRESET_DLY_CTL0 – LP11 reset delay adjustment register

This register selects the LP11 reset delay.

Access: This register can be read/written in 32-bit units.

Address: <MIPIn_base> + $2C_H$

Initial value: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	0	0	0	0	RESET_DLY_LANE1[11:0]											
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	0	0	0	0	RESET_DLY_LANE0[11:0]											
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 37.61 MIPInRESET_DLY_CTL0 register contents

Bit position	Bit name	Function
31 to 28	Reserved	These bits are always read as 0. When written, write the initial value.
27 to 16	RESET_DLY_ LANE1[11:0]	Control reset delay for data lane 1. **CAUTION** The initial value of RESET_DLY_LANE1[11:0] must be changed to $03F_H$ during the MIPI I/F initialization.
15 to 12	Reserved	These bits are always read as 0. When written, write the initial value.
11 to 0	RESET_DLY_ LANE0[11:0]	Control reset delay for data lane 0. **CAUTION** The initial value of RESET_DLY_LANE0[11:0] must be changed to $03F_H$ during the MIPI I/F initialization.

37.8.4.12 MIPInINTSTATUS – Interrupt status register

This register shows the status of interrupt.

For each interrupt a dedicated status bit is provided with the following meaning:

- interrupt status bit = 0: interrupt has not occurred
- interrupt status bit = 1: interrupt has occurred

De-assertion of an interrupt and clearing of its status bit can be executed via the MIPInINTFFCLR register.

NOTE

If an interrupt is disabled (by writing 1 to its corresponding mask bit in the MIPInINTENCLR register), its status bit in MIPInSTATUS remains 0, even if the respective interrupt condition is fulfilled.

Access: This register can be read in 32-bit units.

Address: <MIPIn_base> + 40_H

Initial value: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	0	0	0	0	ESC_ TRIG_ UK5	ESC_ TRIG_ UK4	ESC_ TRIG_ UK3	ESC_ TRIG_ RESET	0	0	0	0	0	0	PHWC_ ERR	CTL_ ERR_1
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ESC_ ERR_ 1_3	SOT_ SYNC_ ERR_1	SOT_ ERR_1	CTL_ ERR_0	ESC_ ERR_0	SOT_ SYNC_ ERR_0	SOT_ ERR_0	BUF_ OR_ ERR	0	0	FRAME SYNC_ ERR	0	CRC_ ERR	ECC_ SINGLE _ERR	ECC_ MULTI_ ERR	0
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 37.62 MIPInINTSTATUS register contents (1/2)

Bit position	Bit name	Function
31 to 28	Reserved	These bits are always read as 0. When written, write the initial value.
27	ESC_TRIG_ UK5	Unknown-5 trigger
26	ESC_TRIG_ UK4	Unknown-4 trigger
25	ESC_TRIG_ UK3	Unknown-3 trigger
24	ESC_TRIG_ RESET	Detect reset-trigger (remote application)
23 to 18	Reserved	These bits are always read as 0. When written, write the initial value.
17	PHWC_ERR	Detect packet header WORDCOUNT error
16	CTL_ERR_1	False Control Error for data lane 1
15	ESC_ERR_1_3	Escape mode entry command error for data lane 1
14	SOT_SYNC_ ERR_1	Start-of-Transmission (SoT) synchronization error for data lane 1
13	SOT_ERR_1	Start-of-Transmission (SoT) error for data lane 1

Table 37.62 MIPInINTSTATUS register contents (2/2)

Bit position	Bit name	Function
12	CTL_ERR_0	False Control Error for data lane 0
11	ESC_ERR_0	Escape mode entry command error for data lane 0
10	SOT_SYNC_ERR_0	Start-of-Transmission (SoT) synchronization error for data lane 0
9	SOT_ERR_0	Start-of-Transmission (SoT) error for data lane 0
8	BUF_OR_ERR	BUFFER overrun error
7 to 6	Reserved	These bits are always read as 0. When written, write the initial value.
5	FRAMESYNC_ERR	FRAME SYNC error
4	Reserved	These bits are always read as 0. When written, write the initial value.
3	CRC_ERR	CRC error detection
2	ECC_SINGLE_ERR	Single bit ECC error detection
1	ECC_MULTI_ERR	Multi bit ECC error detection
0	Reserved	These bits are always read as 0. When written, write the initial value.

Interrupt details

- Unknown-5,4,3 trigger (bit[27:25])
  - If Unknown-5,4,3 trigger commands are received, these interrupts are asserted.
  - Entry Command Patterns are 10100000, 00100001 and 01011101.
- Reset-trigger (Remote Application, bit[24])
  - If Reset-trigger commands are received, this interrupt are asserted.
  - Entry Command Pattern is 01100010.
- Packet Header word count error (bit[17])
  - Asserted when word count is lager than
    Actual Data Size + ((Ths-trail + Ths-exit) / (UI * 8)) * Number of Lane
    and receive next packet with the LP buffer tuerned on status. (LP buffer turn on timing is controllable by MIPInLP_EN_ON_WC register).
- False Control Error (bit[16], [12])
  - If a LP-Rqst (LP-10) is not followed by the remainder of a valid Escape or Turnaround sequence, a False Control Error is indicated. This error is also indicated if a HS-Rqst (LP-01) is not correctly followed by a Bridge State (LP-00).
- Escape Mode Entry Command Error (bit[15], [11])
  - If the receiving Lane Module does not recognize the received Entry Command for Escape mode an Escape mode Entry Command Error is indicated.
- SoT Sync Error (bit[14], [10])

  - If the SoT Leader sequence is corrupted in a way that proper synchronization cannot be expected, a SoT Sync Error is indicated.

  When SOT_SYNC_ERR is detected the MIPI module must be reset by using the MIPInRST_CTL register. See Section 37.8.4.4, MIPInRST_CTL – Reset register for details.

- SoT Error (bit[13], [9])
  - The Leader sequence for Start of High-Speed Transmission is fault tolerant for any single-bit error and some multi bit errors. Therefore, the synchronization may be usable, but confidence in the payload data is lower. If this situation occurs an SoT Error is indicated.
- BUFFER overrun error(bit[8])
  - Asserted when the overrun is occurred in the internal data FIFO of MIPI module.
- FRAMESYNC_ERR (bit[5])
  - Asserted when a Frame End (FE) is not paired with a Frame Start (FS) on the same virtual channel.
- CRC_ERR (bit[3])
  - Asserted when the computed CRC code is different than the received CRC code.
- ECC_SINGLE_ERR (bit[2])
  - Asserted when an ECC syndrome was computed and a single bit-error in the packet header was detected and corrected.
- ECC_MULTI_ERR (bit[1])
  - Asserted when an ECC syndrome was computed and two bit-errors are detected in the received packet header.

37.8.4.13 MIPInINTENSET – Interrupt enable set register

This register is used to enable the interrupts.

For each interrupt a dedicated interrupt enable bit is provided with the following function:

- at write:
  - interrupt enable bit = 0: no function
  - interrupt enable bit = 1: enable interrupt
- at read:
  - interrupt enable bit = 0: interrupt disabled
  - interrupt enable bit = 1: interrupt enabled

Disabling an interrupt can be executed via the MIPInINTENCLR register.

Enabling the interrupts

- Unknown-5,4,3 trigger (ESC_TRIG_UK5_EN, ESC_TRIG_UK4_EN, ESC_TRIG_UK3_EN = 1)
- Reset-trigger (ESC_TRIG_RESET_EN = 1)
- False Control Error (CTL_ERR_1_EN = 1, CTL_ERR_0_EN = 1)
- Escape Mode Entry Command Error (ESC_ERR_1_3_EN = 1, ESC_ERR_0_EN = 1)

should be used with the noise removal of clock line enabled (MIPInMODE.DEL_LP11_NOISE = 1).

Access: This register can be read/written in 32-bit units.

Address: <MIPIn_base> + 48_H

Initial value: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	0	0	0	0	ESC_TRIG_UK5_EN	ESC_TRIG_UK4_EN	ESC_TRIG_UK3_EN	ESC_TRIG_RESET_EN	0	0	0	0	0	0	PHWC_ERR_EN	CTL_ERR_1_EN
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ESC_ERR_1_3_EN	SOT_SYNC_ERR_1_EN	SOT_ERR_1_EN	CTL_ERR_0_EN	ESC_ERR_0_EN	SOT_SYNC_ERR_0_EN	SOT_ERR_0_EN	BUF_OR_ERR_EN	0	0	FRAME SYNC_ERR_EN	0	CRC_ERR_EN	ECC_SINGLE_ERR_EN	ECC_MULTI_ERR_EN	0
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R	R/W	R/W	R/W	R

Table 37.63 MIPInINTENSET register contents (1/2)

Bit position	Bit name	Function
31 to 28	Reserved	These bits are always read as 0. When written, write the initial value.
27	ESC_TRIG_UK5_EN	Unknown-5 trigger enable
26	ESC_TRIG_UK4_EN	Unknown-4 trigger enable
25	ESC_TRIG_UK3_EN	Unknown-3 trigger enable

Table 37.63 MIPInINTENSET register contents (2/2)

Bit position	Bit name	Function
24	ESC_TRIG_RESET_EN	Detect reset-trigger (remote application) enable This is only a regular interrupt triggered by a MIPI escape sequence. It is similar to the unknown trigger events and does not trigger any kind of reset.
23 to 18	Reserved	These bits are always read as 0. When written, write the initial value.
17	PHWC_ERR_EN	Detect packet header WORDCOUNT error enable
16	CTL_ERR_1_EN	False Control Error for data lane 1 enable
15	ESC_ERR_1_3_EN	Escape mode entry command error for data lane 1 enable
14	SOT_SYNC_ERR_1_EN	Start-of-Transmission (SoT) synchronization error for data lane 1 enable
13	SOT_ERR_1_EN	Start-of-Transmission (SoT) error for data lane 1 enable
12	CTL_ERR_0_EN	False Control Error for data lane 0 enable
11	ESC_ERR_0_EN	Escape mode entry command error for data lane 0 enable
10	SOT_SYNC_ERR_0_EN	Start-of-Transmission (SoT) synchronization error for data lane 0 enable
9	SOT_ERR_0_EN	Start-of-Transmission (SoT) error for data lane 0 enable
8	BUF_OR_ERR_EN	BUFFER overrun error enable
7 to 6	Reserved	These bits are always read as 0. When written, write the initial value.
5	FRAMESYNC_ERR_EN	FRAME SYNC error enable
4	Reserved	These bits are always read as 0. When written, write the initial value.
3	CRC_ERR_EN	CRC error detection enable
2	ECC_SINGLE_ERR_EN	Single bit ECC error detection enable
1	ECC_MULTI_ERR_EN	Multi bit ECC error detection enable
0	Reserved	These bits are always read as 0. When written, write the initial value.

37.8.4.14 MIPInINTENCLR – Interrupt enable clear register

This register is used to mask the interrupts.

For each interrupt a dedicated interrupt mask bit is provided with the following function:

- at write:
  - interrupt mask bit = 0: no function
  - interrupt mask bit = 1: disable (mask) interrupt
- at read: always 0

Enabling an interrupt can be executed via the MIPInINTENSET register.

Access: This register can be written in 32-bit units.

Address: <MIPIn_base> + $4C_H$

Initial value: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	0	0	0	0	ESC_TRIG_UK5_MASK	ESC_TRIG_UK4_MASK	ESC_TRIG_UK3_MASK	ESC_TRIG_RESET_MASK	0	0	0	0	0	0	PHWC_ERR_MASK	CTL_ERR_1_MASK
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	W	W	W	W	R	R	R	R	R	R	W	W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ESC_ERR_1_3_MASK	SOT_SYNC_ERR_1_MASK	SOT_ERR_1_MASK	CTL_ERR_0_MASK	ESC_ERR_0_MASK	SOT_SYNC_ERR_0_MASK	SOT_ERR_0_MASK	BUF_OR_ERR_MASK	0	0	FRAMESYNC_ERR_MASK	0	CRC_ERR_MASK	ECC_SINGLE_ERR_MASK	ECC_MULTI_ERR_MASK	0
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	R	R	W	R	W	W	W	R

Table 37.64 MIPInINTENCLR register contents (1/2)

Bit position	Bit name	Function
31 to 28	Reserved	These bits are always read as 0. When written, write the initial value.
27	ESC_TRIG_UK5_MASK	Unknown-5 trigger mask
26	ESC_TRIG_UK4_MASK	Unknown-4 trigger mask
25	ESC_TRIG_UK3_MASK	Unknown-3 trigger mask
24	ESC_TRIG_RESET_MASK	Detect reset-trigger (remote application) mask
23 to 18	Reserved	These bits are always read as 0. When written, write the initial value.
17	PHWC_ERR_MASK	Detect packet header WORDCOUNT error mask
16	CTL_ERR_1_MASK	False Control Error for data lane 1 mask
15	ESC_ERR_1_3_MASK	Escape mode entry command error for data lane 1 mask
14	SOT_SYNC_ERR_1_MASK	Start-of-Transmission (SoT) synchronization error for data lane 1 mask
13	SOT_ERR_1_MASK	Start-of-Transmission (SoT) error for data lane 1 mask

Table 37.64 MIPInINTENCLR register contents (2/2)

Bit position	Bit name	Function
12	CTL_ERR_0_MASK	False Control Error for data lane 0 mask
11	ESC_ERR_0_MASK	Escape mode entry command error for data lane 0 mask
10	SOT_SYNC_ERR_0_MASK	Start-of-Transmission (SoT) synchronization error for data lane 0 mask
9	SOT_ERR_0_MASK	Start-of-Transmission (SoT) error for data lane 0 mask
8	BUF_OR_ERR_MASK	BUFFER overrun error mask
7 to 6	Reserved	These bits are always read as 0. When written, write the initial value.
5	FRAMESYNC_ERR_MASK	FRAME SYNC error mask
4	Reserved	These bits are always read as 0. When written, write the initial value.
3	CRC_ERR_MASK	CRC error detection mask
2	ECC_SINGLE_ERR_MASK	Single bit ECC error detection mask
1	ECC_MULTI_ERR_MASK	Multi bit ECC error detection mask
0	Reserved	These bits are always read as 0. When written, write the initial value.

37.8.4.15 MIPInINTFFCLR – Interrupt factor clear register

This register is used to de-assert an interrupt and clear its status flag in the MIPInINTSTATUS register..

For each interrupt a dedicated interrupt bit is provided with the following function:

- at write:
  - interrupt factor clear bit = 0: no function
  - interrupt factor clear bit = 1: de-assert interrupt and clear its status bit in MIPInINTSTATUS
- at read: always 0

Access: This register can be written in 32-bit units.

Address: <MIPIn_base> + 50_H

Initial value: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	0	0	0	0	ESC_TRIG_UK5_CLR	ESC_TRIG_UK4_CLR	ESC_TRIG_UK3_CLR	ESC_TRIG_RESET_CLR	0	0	0	0	0	0	PHWC_ERR_CLR	CTL_ERR_1_CLR
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	W	W	W	W	R	R	R	R	R	R	W	W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ESC_ERR_1_3_CLR	SOT_SYNC_ERR_1_CLR	SOT_ERR_1_CLR	CTL_ERR_0_CLR	ESC_ERR_0_CLR	SOT_SYNC_ERR_0_CLR	SOT_ERR_0_CLR	BUF_OR_ERR_CLR	0	0	FRAME SYNC_ERR_CLR	0	CRC_ERR_CLR	ECC_SINGLE_ERR_CLR	ECC_MULTI_ERR_CLR	0
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	R	R	W	R	W	W	W	R

Table 37.65 MIPInINTFFCLR register contents (1/2)

Bit position	Bit name	Function
31 to 28	Reserved	These bits are always read as 0. When written, write the initial value.
27	ESC_TRIG_UK5_CLR	Unknown-5 trigger clear
26	ESC_TRIG_UK4_CLR	Unknown-4 trigger clear
25	ESC_TRIG_UK3_CLR	Unknown-3 trigger clear
24	ESC_TRIG_RESET_CLR	Detect reset-trigger (remote application) clear
23 to 18	Reserved	These bits are always read as 0. When written, write the initial value.
17	PHWC_ERR_CLR	Detect packet header WORDCOUNT error clear
16	CTL_ERR_1_CLR	False Control Error for data lane 1 enable clear
15	ESC_ERR_1_3_CLR	Escape mode entry command error for data lane 1 enable clear
14	SOT_SYNC_ERR_1_CLR	Start-of-Transmission (SoT) synchronization error for data lane 1 enable clear
13	SOT_ERR_1_CLR	Start-of-Transmission (SoT) error for data lane 1 enable clear
12	CTL_ERR_0_CLR	False Control Error for data lane 0 enable clear

Table 37.65 MIPInINTFFCLR register contents (2/2)

Bit position	Bit name	Function
11	ESC_ERR_0_CLR	Escape mode entry command error for data lane 0 enable clear
10	SOT_SYNC_ERR_0_CLR	Start-of-Transmission (SoT) synchronization error for data lane 0 enable clear
9	SOT_ERR_0_CLR	Start-of-Transmission (SoT) error for data lane 0 enable clear
8	BUF_OR_ERR_CLR	BUFFER overrun error enable clear
7 to 6	Reserved	These bits are always read as 0. When written, write the initial value.
5	FRAMESYNC_ERR_CLR	FRAME SYNC error clear
4	Reserved	These bits are always read as 0. When written, write the initial value.
3	CRC_ERR_CLR	CRC error detection clear
2	ECC_SINGLE_ERR_CLR	Single bit ECC error detection clear
1	ECC_MULTI_ERR_CLR	Multi bit ECC error detection clear
0	Reserved	These bits are always read as 0. When written, write the initial value.

37.8.4.16 MIPInEOT_COUNT – T_{EOT} setting

This register sets the reset timing of DES according T_{EOT}.

EOT_COUNT[7:0] refers to the number of MIPIADCLK cycles. It is the period from changing LP11 to beginning reset.

In the case that the data rate is more than 266 Mbps per one lane, this register should be set 0.

Otherwise, it is necessary to satisfy the following condition.

- MIPInMODE.SOT_CLKSEL = 0 (MIPIADCLK is $T_{HS-SETTLE}$ count clock):
  16 UI < $T_{MIPIADCLK}$ x EOT_COUNT[7:0]
  ($T_{MIPIADCLK}$ = MIPIADCLK cycle duration)
- MIPInMODE.SOT_CLKSEL = 1(external clock divided by 4 is $T_{HS-SETTLE}$ count clock):
  EOT_COUNT[7:0] = 3

Access: This register can be read/written in 32-bit units.

Address: <MIPIn_base> + 74_H

Initial value: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	0	0	0	0	0	0	0	0	EOT_COUNT[7:0]							
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 37.66 MIPInEOT_COUNT register contents

Bit position	Bit name	Function
31 to 8	Reserved	These bits are always read as 0. When written, write the initial value.
7 to 0	EOT_ COUNT[7:0]	Counter value indicating reset timing for EoT.

37.8.4.17 MIPInVIN_MODE – Video input color format register

This register selects the color format of the video data.

Access: This register can be read/written in 32-bit units.

Address: <MIPIn_base> + $9C_H$

Initial value: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	VIN_MODE[1:0]	
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Table 37.67 MIPInVIN_MODE register contents

Bit position	Bit name	Function
31 to 2	Reserved	These bits are always read as 0. When written, write the initial value.
1 to 0	VIN_ MODE[1:0]	Selection of the video data color format 00_B: YUV422 01_B: RGB888 10_B: RGB565 11_B: Setting prohibited

37.8.5 MIPI Operation

37.8.5.1 Start and stop sequence

MIPI port setup

(1) Select alternative port function: i.e. PMC40.PMC40_[5:0] = 11 1111_B for two data lanes

(2) Set ports as input: i.e. PM40.PM40_[5:0] = 11 1111_B for two data lanes

(3) Wait 20 µsec buffer stabilization time

(4) MIPI video source and clock setup
- Select video input channel 0 for MIPI CSI-2: VDCECTL.PXSL = 1
- Setup clocks
  – C_ISO_MIPIADCLK
  – C_ISO_MIPIPIXCLK

Start sequence

1. Reset by MIPInRST_CTL = 0000 0003_H
2. Set up following registers:
   - MIPInMODE
   - MIPInDATA_DLY_CTL
   - MIPInBUF_CTL
   - MIPInSOT_COUNT
   - MIPInWORD_COUNT
   - MIPInLP_EN_ON_WC
   - MIPInLINE_BLANK
   - MIPInRESET_DLY_CTL0
   - MIPInEOT_COUNT
   - MIPInVIN_MODE
3. Release reset by MIPInRST_CTL = 0000 0000_H
4. Enable MIPI interrupts by set up to the following registers:
   – MIPInINTENCLR
   – MIPInINTFFCLR
   – MIPInINTENSET
   Activate the following interrupts by the Interrupt Controller:
   – INTMIPI0OVF
   – INTMIPI0CTL
5. Enable MIPI transmission by MIPInON.MIPI_ON = 1.

Example start sequence

In the following a register set-up example is given for a MIPI CSI-2 video data stream with following parameters:

- 640 x 480 pixel size (VGA with VESA timing)
- 24 bpp (RGB888)

- using 2 data lanes of MIPI CSI-2
- using clock C_ISO_MIPIPIXCLK = 26.67 MHz
  (from C_ISO_MIPIADCLK = MIPIPLLCLK = PLL0 / 3 = 160 MHz;
  C_ISO_MIPIPIXCLK = MIPIPLLCLK / 6 = 26.67 MHz)

Register settings:

1. MIPInMODE = 2982 0003_H
2. MIPInDATA_DLY_CTL = 0000 0000_H
3. MIPInBUF_CTL = 0000 0000_H
4. MIPInSOT_COUNT = 0014 0014_H
5. MIPInWORD_COUNT = 0000 0000_H
6. MIPInLP_EN_ON_WC = 010A 0020_H
7. MIPInLINE_BLANK = 0000 0004_H
8. MIPInRESET_DLY_CTL0 = 003F $003F_H$
9. MIPInEOT_COUNT = 0000 0000_H
10. MIPInVIN_MODE = 0000 0001_H

CAUTION

Do not change the registers MIPInMODE, MIPInDATA_DLY_CTL, MIPInBUF_CTL, MIPInSOT_COUNT, MIPInWORD_COUNT, MIPInLP_EN_ON_WC, MIPInLINE_BLANK, MIPInRESET_DLY_CTL0, MIPInEOT_COUNT while the MIPI interface is active.

These registers may only be changed after the stop sequence.

Stop sequence

1. Disable MIPI Interrupts by the Interrupt Controller:
   – INTMIPI0OVF
   – INTMIPI0CTL
2. Disable the MIPI interface by MIPInON.MIPI_ON = 0.

37.8.5.2 Interrupt handling sequence

- INTMIPI0OVF interrupt handling:
  When MIPI buffer overflow is detected the MIPI module must be reset by using the MIPInRST_CTL register. See Section 37.8.4.4, MIPInRST_CTL – Reset register for details.
- INTMIPI0CTL interrupt handling:
  Evaluate interrupt cause by MIPInINTSTATUS interrupt status register and handle cause as described in Section 37.8.4.12, MIPInINTSTATUS – Interrupt status register.

37.8.5.3 Video Input Setup for MIPI

The MIPI Video Input Interface provides the video data to the Video Input channel 0 with the following attributes:

1. sync and data capture point
   - horizontal- and vertical- sync is to be captured on rising clock edge
   - data is to be captured on riding clock edge
2. horizontal blanking:
   - sync width fixed to 1 pixel clock
   - back porch fixed to 16 pixel clocks
   - active high horizontal sync
3. vertical blanking:
   - sync width fixed to 1 line
   - back porch fixed to 3 lines
   - active high vertical sync
   - first data line is the line with active sync

That means the following register settings have to be performed for the VDCE Video Input channel 0:

- DEMODE0.DE_4HS_EN = 1 when MIPInMODE.VTIM is 0.
- INP_SEL_CNT.INP_PXD_EDGE = 0 (rising edge)
- INP_SEL_CNT.INP_VS_EDGE = 0 (rising edge)
- INP_SEL_CNT.INP_HS_EDGE = 0 (rising edge)
- SC0_SCL0_DS2.SC0_RES_VS = 4 (refer to note below)
  (1 line vsync + 3 line backporch = 4 lines)
- SC0_SCL0_DS2.SC0_RES_VW = 484 (refer to note below)
  (height of video signal + 4 = 484 lines at VGA)
- SC0_SCL0_DS3.SC0_RES_HS = 17 (1 pixel hsync + 16 pixel backporch = 17 pixel)
- SC0_SCL0_DS3.SC0_RES_HW = 640 (width of video signal = 640 pixels at VGA)

NOTE

The frame sync (vertical sync and horizontal sync) information is contained in the actual data packets (long packets) of the MIPI CSI-2 data stream.

By this fact the vertical sync is provided to the VDCE video input channel 0 together with the first line of video data. As the VDCE video input requires 4 lines from the reception of VSYNC to the capture of the first line (min. value of SC0_SCL0_DS2.SC0_RES_VS register is 4) the first 4 lines can't be captured.

However, this issue can be avoided by setting DEMODE0.DE_4HS_EN to 1 in VDCE0 and MIPInMODE.VTIM to 0. By this setting, VDC automatically shift 4 lines after VSYNC detection between VSYNC and the first HSYNC from MIPI.

Also, when DEMODE0.DE_4HS_EN is set to 1, the detection of the end of line is available by setting DEMODE0.DE_VLAST_EN. For the detail, please refer DEMODE0 and DEMODE1 in VDCE chapter.

When MIPInMODE.VTIM is 1, the MIPI CSI-2 video data stream needs to contain 4 lines of dummy data as first active data lines.

I.e. when a 640 x 480 pixels size video input source (VGA) shall be captured by the VDCE video input via MIPI CSI-2, provide a 640 x 484 pixels size video source with lines 1 to 4 containing dummy data and lines 5 to 484 containing the actual video data.

Application hint: This can be done by providing a data enable signal to the MIPI CSI-2 generating video converter, that starts 4 lines earlier than the actual video data.

37.8.5.4 Conditions for MIPI video clock selection

The MIPI Video Input Interface can select the pixel clock to the Video Input channel 0 (C_ISO_MIPIPIXCLK) generated from the clock source C_ISO_MIPIADCLK by division.

The selection of this divider (CKSC_IMIPIPIXD_CTL) needs to fulfill the following two conditions.

(1) Condition 1: VIN pixel clock (restriction of MIPI Video data buffers)

- C_ISO_MIPIPIXCLK > VIN pixel clock
- (C_ISO_MIPIADCLK / DIV) > VIN pixel clock

with DIV is 3, 6 or 12. (Refer to Section 37.5.7.29, CKSC_IMIPIPIXD_CTL — C_ISO_MIPIPIXCLK clock divider register (D1M2(H) only))

Example:

- C_ISO_MIPIADCLK = 160 MHz
- QVGA 240 x 320 (native video data timing of data embedded into MIPI CSI-2 video stream)

⇒ VIN pixel clock = 5 MHz
(160 MHz / 12) = 13.34 MHz > 5 MHz

⇒ Select DIV = 12

(2) Condition 2: max. HSYNC width (restriction of VDCE)

- (C_ISO_MIPIPIXCLK × VIN total width) / VIN pixel clock ≤ 2015 pixels
- ((C_ISO_MIPIADCLK / DIV) × VIN total width) / VIN pixel clock ≤ 2015 pixels

with DIV is 3, 6 or 12. (Refer to Section 37.5.7.29, CKSC_IMIPIPIXD_CTL — C_ISO_MIPIPIXCLK clock divider register (D1M2(H) only))

Example:

- C_ISO_MIPIADCLK = 160 MHz
- QVGA 240 x 320 (native video data timing of data embedded into MIPI CSI-2 video stream)

⇒ VIN total width = 320 + 84 = 404 pixel

⇒ VIN pixel clock = 5 MHz
((160MHz / 12) * 404 pixel) / 5MHz = 1078 <= 2015 pixels

⇒ Select DIV = 12

37.8.5.5 VSYNC Timing from MIPI to VDCE

The following figure show the difference of VSYNC output timing by MIPInMODE.VTIM.

When MIPInMODE.VTIM is 0, VSYNC is asserted before almost 1 line from the first line output. Also, VSYNC is asserted until next data come to data buffer.

When MIPInMODE.VTIM is 1, VSYNC is output at the same timing as HSYNC to VDCE.

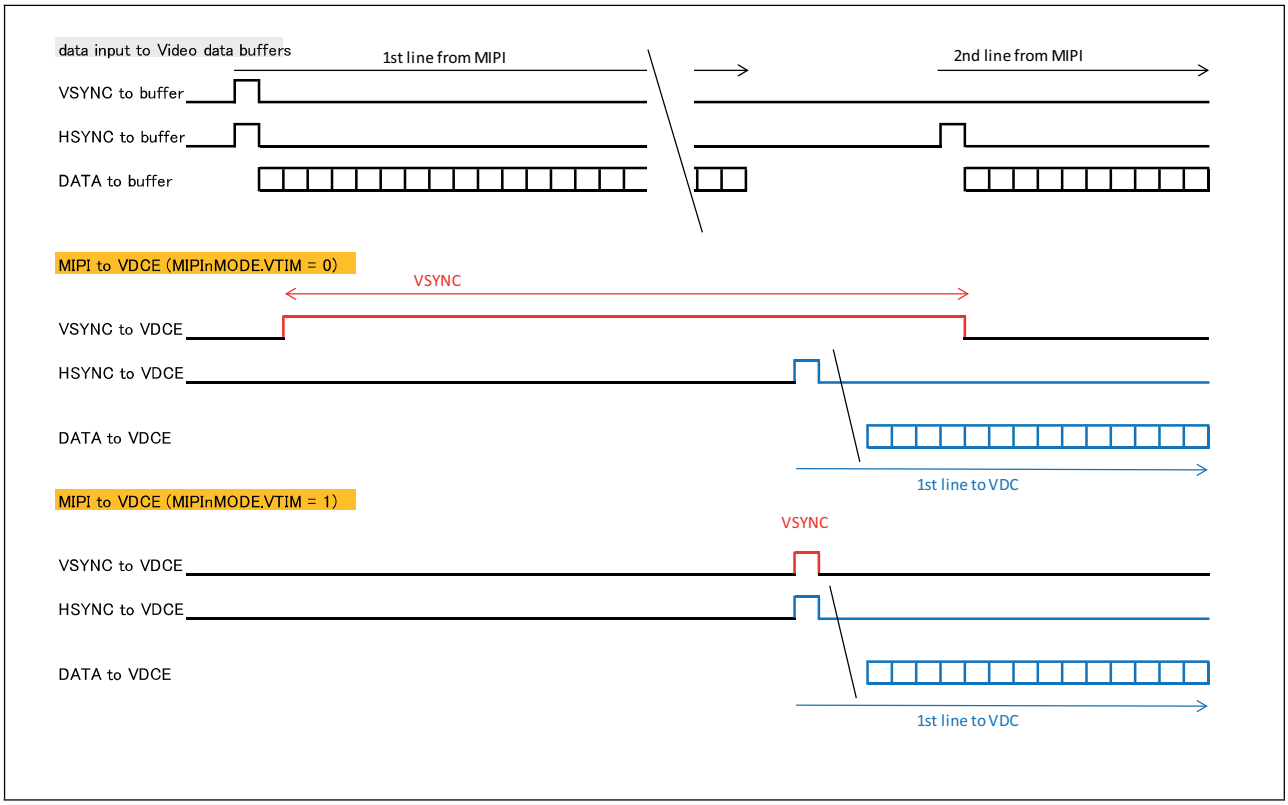


Figure 37.19 VSYNC Timing from MIPI to VDCE

NOTE

- MIPInMODE.VTIM is 1, DEMODE0 and 1 in VDCE0 register can be available.
- VDCE setting except DEMODE0 and 1 for MIPI is the same regardless MIPInMODE.VTIM setting.

MIPI setting

1. Fix the capturing position
   - SC_SCL0_DS2.SC_RES_VS = 4
   - SC_SCL0_DS3.SC_RES_HS = 17
2. DEMODE setting

   [No line detection]
   - DEMODE0=0x00000010

   [line detection after the last line finished]

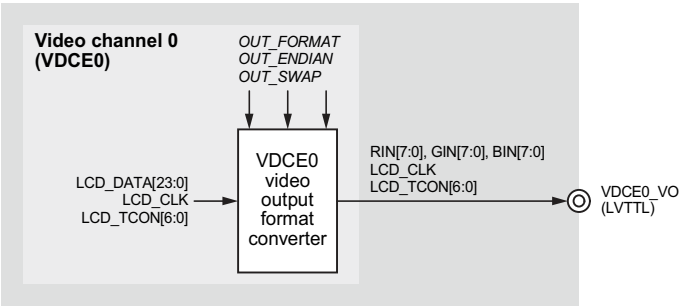
- DEMODE0=0x00000014
- DEMODE1= active line number
  e.g) when active vertical width is 480, DEMODE1 is 480.
- SC_SCL0_INT.SC_RES_LINE = active line number + 4
  e.g) when active vertical width is 480, SC_RES_LINE is 484.

3. Other VDCE setting is the same as video input setting with HSYNC

37.9 Video Output selection and RSDS control

37.9.1 D1L2(H) and D1M1(H) Video output selection

The diagram below illustrates the video output signal connections.



Note: The Video Data Controllers VDCEn include facilities to change the mapping of the color output data LCD_DATA[23:0] to the VDCEn output data RIN[7:0], GIN[7:0] and BIN[7:0]. Refer to Section 38, Video Data Controller E (VDCE) for details about the OUT_FORMAT, OUT_ENDIAN and OUT_SWAP settings.

Figure 37.20 D1L2(H) and D1M1(H) video output signals

The following table shows the assignment between the various video output signals for following settings:

- VDCEn video output format converter settings:
  - OUT_SET.OUT_FORMAT[1:0] = 00_B
  - OUT_SET.OUT_ENDIAN_ON = 0
  - OUT_SET.OUT_SWAP_ON = 0

  For other bit allocations of LCD_DATA[23:0] to RIN[7:0], GIN[7:0] and BIN[7:0] refer to Section 38, Video Data Controller E (VDCE).

Table 37.68 D1L2(H) and D1M1(H) Video output format selection example (1/2)

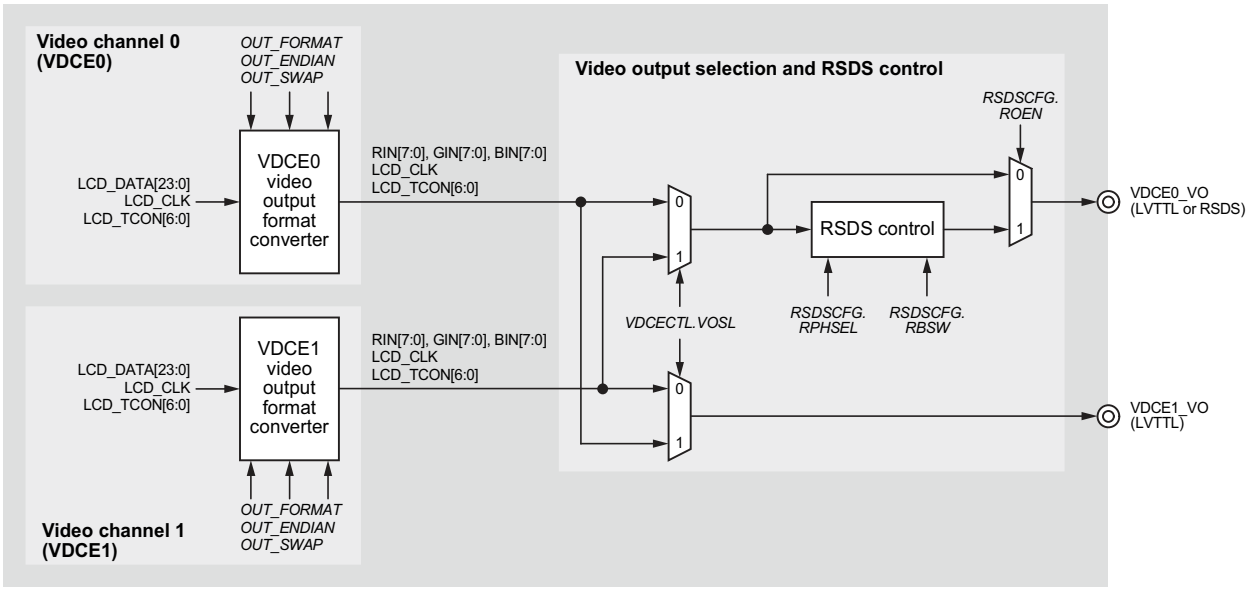
VDCE0 video output data	VDCE0 video output format converter output	Port signal name	LVTTL signal
Color data signals			
LCD_DATA23	RIN[7]	VDCE0_VO_DATA23	R[7]
LCD_DATA22	RIN[6]	VDCE0_VO_DATA22	R[6]
LCD_DATA21	RIN[5]	VDCE0_VO_DATA21	R[5]
LCD_DATA20	RIN[4]	VDCE0_VO_DATA20	R[4]
LCD_DATA19	RIN[3]	VDCE0_VO_DATA19	R[3]
LCD_DATA18	RIN[2]	VDCE0_VO_DATA18	R[2]
LCD_DATA17	RIN[1]	VDCE0_VO_DATA17	R[1]
LCD_DATA16	RIN[0]	VDCE0_VO_DATA16	R[0]
LCD_DATA15	GIN[7]	VDCE0_VO_DATA15	G[7]
LCD_DATA14	GIN[6]	VDCE0_VO_DATA14	G[6]
LCD_DATA13	GIN[5]	VDCE0_VO_DATA13	G[5]
LCD_DATA12	GIN[4]	VDCE0_VO_DATA12	G[4]
LCD_DATA11	GIN[3]	VDCE0_VO_DATA11	G[3]

Table 37.68 D1L2(H) and D1M1(H) Video output format selection example (2/2)

VDCE0 video output data	VDCE0 video output format converter output	Port signal name	LVTTL signal
LCD_DATA10	GIN[2]	VDCE0_VO_DATA10	G[2]
LCD_DATA9	GIN[1]	VDCE0_VO_DATA9	G[1]
LCD_DATA8	GIN[0]	VDCE0_VO_DATA8	G[0]
LCD_DATA7	BIN[7]	VDCE0_VO_DATA7	B[7]
LCD_DATA6	BIN[6]	VDCE0_VO_DATA6	B[6]
LCD_DATA5	BIN[5]	VDCE0_VO_DATA5	B[5]
LCD_DATA4	BIN[4]	VDCE0_VO_DATA4	B[4]
LCD_DATA3	BIN[3]	VDCE0_VO_DATA3	B[3]
LCD_DATA2	BIN[2]	VDCE0_VO_DATA2	B[2]
LCD_DATA1	BIN[1]	VDCE0_VO_DATA1	B[1]
LCD_DATA0	BIN[0]	VDCE0_VO_DATA0	B[0]
Pixel clock			
LCD_CLK	LCD_CLK	VDCE0_VO_CLKP	VDCE0_VO_CLKP
Timing Controller signals			
LCD_TCON0	LCD_TCON0	VDCE0_VO_TCON0	
LCD_TCON1	LCD_TCON1	VDCE0_VO_TCON1	
LCD_TCON2	LCD_TCON2	VDCE0_VO_TCON2	
LCD_TCON3	LCD_TCON3	VDCE0_VO_TCON3	
LCD_TCON4	LCD_TCON4	VDCE0_VO_TCON4	
LCD_TCON5	LCD_TCON5	VDCE0_VO_TCON5	
LCD_TCON6	LCD_TCON6	VDCE0_VO_TCON6	

37.9.2 D1M2(H) Video output selection

The diagram below illustrates the video output selection options.



Note: The Video Data Controllers VDCEn include facilities to change the mapping of the color output data LCD_DATA[23:0] to the VDCEn output data RIN[7:0], GIN[7:0] and BIN[7:0]. Refer to Section 38, Video Data Controller E (VDCE) for details about the OUT_FORMAT, OUT_ENDIAN and OUT_SWAP settings.

Figure 37.21 D1M2(H) video output selections

By use of the VDCECTL.VOSL selector the output signals from the video channels 0 and 1 can be swapped, so that any of both can be used for the RSDS output. The other video output is output as LVTTL signals.

Table 37.69 D1M2(H) Video output selections

VDCECTL.VOSL	RSDSCFG.ROEN	Ports VDCE0_VO	Ports VDCE1_VO
0	0	Video channel 0: LVTTL	Video channel 1: LVTTL
	1	Video channel 0: RSDS	
1	0	Video channel 1: LVTTL	Video channel 0: LVTTL
	1	Video channel 1: RSDS	

37.9.2.1 Video data and clock output settings

The following tables show the assignment between the video data and clock output signals for different settings endian and R – B swap (red channel – blue channel swapping).

Basic video output selection and RSDS:

- RSDSCFG.ROEN = 1: VDCE0_VO as RSDS
- VDCECTL.VOSL determines which video output channel is output as RSDS via VDCE0_VO (no impact on the tables)
- The tables show the assignment for RGB888 output format, selected by OUT_SET.OUT_FORMAT[1:0] = 00_B. For other formats refer to Section 38, Video Data Controller E (VDCE).

(a) VDCEn video output format converter settings (no R – B swap, little endian)

– OUT_SET.OUT_FORMAT[1:0] = 00_B (RGB888 output)

– OUT_SET.OUT_ENDIAN_ON = 0 (little endian)

– OUT_SET.OUT_SWAP_ON = 0 (no R – B swap)

Table 37.70 D1M2(H) Video output format selection (no R – B swap, little endian)

VDCEn video output data	VDCEn video output format converter output	Port signal name	LVTTL	RSDS
Color data signals (LCD_DATA[23:0] from video data channel selected by VDCECTL.VOSL)				
LCD_DATA23	RIN[7]	VDCE0_VO_DATA23	R[7]	RSDS_R[3]_P
LCD_DATA22	RIN[6]	VDCE0_VO_DATA22	R[6]	RSDS_R[3]_N
LCD_DATA21	RIN[5]	VDCE0_VO_DATA21	R[5]	RSDS_R[2]_P
LCD_DATA20	RIN[4]	VDCE0_VO_DATA20	R[4]	RSDS_R[2]_N
LCD_DATA19	RIN[3]	VDCE0_VO_DATA19	R[3]	RSDS_R[1]_P
LCD_DATA18	RIN[2]	VDCE0_VO_DATA18	R[2]	RSDS_R[1]_N
LCD_DATA17	RIN[1]	VDCE0_VO_DATA17	R[1]	RSDS_R[0]_P
LCD_DATA16	RIN[0]	VDCE0_VO_DATA16	R[0]	RSDS_R[0]_N
LCD_DATA15	GIN[7]	VDCE0_VO_DATA15	G[7]	RSDS_G[3]_P
LCD_DATA14	GIN[6]	VDCE0_VO_DATA14	G[6]	RSDS_G[3]_N
LCD_DATA13	GIN[5]	VDCE0_VO_DATA13	G[5]	RSDS_G[2]_P
LCD_DATA12	GIN[4]	VDCE0_VO_DATA12	G[4]	RSDS_G[2]_N
LCD_DATA11	GIN[3]	VDCE0_VO_DATA11	G[3]	RSDS_G[1]_P
LCD_DATA10	GIN[2]	VDCE0_VO_DATA10	G[2]	RSDS_G[1]_N
LCD_DATA9	GIN[1]	VDCE0_VO_DATA9	G[1]	RSDS_G[0]_P
LCD_DATA8	GIN[0]	VDCE0_VO_DATA8	G[0]	RSDS_G[0]_N
LCD_DATA7	BIN[7]	VDCE0_VO_DATA7	B[7]	RSDS_B[3]_P
LCD_DATA6	BIN[6]	VDCE0_VO_DATA6	B[6]	RSDS_B[3]_N
LCD_DATA5	BIN[5]	VDCE0_VO_DATA5	B[5]	RSDS_B[2]_P
LCD_DATA4	BIN[4]	VDCE0_VO_DATA4	B[4]	RSDS_B[2]_N
LCD_DATA3	BIN[3]	VDCE0_VO_DATA3	B[3]	RSDS_B[1]_P
LCD_DATA2	BIN[2]	VDCE0_VO_DATA2	B[2]	RSDS_B[1]_N
LCD_DATA1	BIN[1]	VDCE0_VO_DATA1	B[1]	RSDS_B[0]_P
LCD_DATA0	BIN[0]	VDCE0_VO_DATA0	B[0]	RSDS_B[0]_N
Pixel clock (LCD_CLK from video data channel selected by VDCECTL.VOSL)*1				
LCD_CLK	LCD_CLK	VDCE0_VO_CLKP	VDCE0_VO_CLKP	RSDS_CLK_P
		VDCE0_VO_CLKN	–	RSDS_CLK_N

Note 1. The clock signals are not affected by R – B swap and endian selection.

(b) VDCEn video output format converter settings (R – B swap, little endian)

– OUT_SET.OUT_FORMAT[1:0] = 00_B (RGB888 output)

– OUT_SET.OUT_ENDIAN_ON = 0 (little endian)

– OUT_SET.OUT_SWAP_ON = 1 (R – B swap)

Table 37.71 D1M2(H) Video output format selection (R – B swap, little endian)

VDCEn video output data	VDCEn video output format converter output	Port signal name	LVTTL	RSDS
Color data signals (LCD_DATA[23:0] from video data channel selected by VDCECTL.VOSL)				
LCD_DATA23	BIN[7]	VDCE0_VO_DATA23	B[7]	RSDS_B[3]_P
LCD_DATA22	BIN[6]	VDCE0_VO_DATA22	B[6]	RSDS_B[3]_N
LCD_DATA21	BIN[5]	VDCE0_VO_DATA21	B[5]	RSDS_B[2]_P
LCD_DATA20	BIN[4]	VDCE0_VO_DATA20	B[4]	RSDS_B[2]_N
LCD_DATA19	BIN[3]	VDCE0_VO_DATA19	B[3]	RSDS_B[1]_P
LCD_DATA18	BIN[2]	VDCE0_VO_DATA18	B[2]	RSDS_B[1]_N
LCD_DATA17	BIN[1]	VDCE0_VO_DATA17	B[1]	RSDS_B[0]_P
LCD_DATA16	BIN[0]	VDCE0_VO_DATA16	B[0]	RSDS_B[0]_N
LCD_DATA15	GIN[7]	VDCE0_VO_DATA15	G[7]	RSDS_G[3]_P
LCD_DATA14	GIN[6]	VDCE0_VO_DATA14	G[6]	RSDS_G[3]_N
LCD_DATA13	GIN[5]	VDCE0_VO_DATA13	G[5]	RSDS_G[2]_P
LCD_DATA12	GIN[4]	VDCE0_VO_DATA12	G[4]	RSDS_G[2]_N
LCD_DATA11	GIN[3]	VDCE0_VO_DATA11	G[3]	RSDS_G[1]_P
LCD_DATA10	GIN[2]	VDCE0_VO_DATA10	G[2]	RSDS_G[1]_N
LCD_DATA9	GIN[1]	VDCE0_VO_DATA9	G[1]	RSDS_G[0]_P
LCD_DATA8	GIN[0]	VDCE0_VO_DATA8	G[0]	RSDS_G[0]_N
LCD_DATA7	RIN[7]	VDCE0_VO_DATA7	R[7]	RSDS_R[3]_P
LCD_DATA6	RIN[6]	VDCE0_VO_DATA6	R[6]	RSDS_R[3]_N
LCD_DATA5	RIN[5]	VDCE0_VO_DATA5	R[5]	RSDS_R[2]_P
LCD_DATA4	RIN[4]	VDCE0_VO_DATA4	R[4]	RSDS_R[2]_N
LCD_DATA3	RIN[3]	VDCE0_VO_DATA3	R[3]	RSDS_R[1]_P
LCD_DATA2	RIN[2]	VDCE0_VO_DATA2	R[2]	RSDS_R[1]_N
LCD_DATA1	RIN[1]	VDCE0_VO_DATA1	R[1]	RSDS_R[0]_P
LCD_DATA0	RIN[0]	VDCE0_VO_DATA0	R[0]	RSDS_R[0]_N
Pixel clock (LCD_CLK from video data channel selected by VDCECTL.VOSL)*1				
LCD_CLK	LCD_CLK	VDCE0_VO_CLKP	VDCE0_VO_CLKP	RSDS_CLK_P
		VDCE0_VO_CLKN	–	RSDS_CLK_N

Note 1. The clock signals are not affected by R – B swap and endian selection.

(c) VDCEn video output format converter settings (no R – B swap, big endian)

– OUT_SET.OUT_FORMAT[1:0] = 00_B (RGB888 output)

– OUT_SET.OUT_ENDIAN_ON = 1 (big endian)

– OUT_SET.OUT_SWAP_ON = 0 (no R – B swap)

Table 37.72 D1M2(H) Video output format selection (no R – B swap, big endian)

VDCEn video output data	VDCEn video output format converter output	Port signal name	LVTTL	RSDS
Color data signals (LCD_DATA[23:0] from video data channel selected by VDCECTL.VOSL)				
LCD_DATA23	RIN[0]	VDCE0_VO_DATA23	R[0]	RSDS_R[0]_P
LCD_DATA22	RIN[1]	VDCE0_VO_DATA22	R[1]	RSDS_R[0]_N
LCD_DATA21	RIN[2]	VDCE0_VO_DATA21	R[2]	RSDS_R[1]_P
LCD_DATA20	RIN[3]	VDCE0_VO_DATA20	R[3]	RSDS_R[1]_N
LCD_DATA19	RIN[4]	VDCE0_VO_DATA19	R[4]	RSDS_R[2]_P
LCD_DATA18	RIN[5]	VDCE0_VO_DATA18	R[5]	RSDS_R[2]_N
LCD_DATA17	RIN[6]	VDCE0_VO_DATA17	R[6]	RSDS_R[3]_P
LCD_DATA16	RIN[7]	VDCE0_VO_DATA16	R[7]	RSDS_R[3]_N
LCD_DATA15	GIN[0]	VDCE0_VO_DATA15	G[0]	RSDS_G[0]_P
LCD_DATA14	GIN[1]	VDCE0_VO_DATA14	G[1]	RSDS_G[0]_N
LCD_DATA13	GIN[2]	VDCE0_VO_DATA13	G[2]	RSDS_G[1]_P
LCD_DATA12	GIN[3]	VDCE0_VO_DATA12	G[3]	RSDS_G[1]_N
LCD_DATA11	GIN[4]	VDCE0_VO_DATA11	G[4]	RSDS_G[2]_P
LCD_DATA10	GIN[5]	VDCE0_VO_DATA10	G[5]	RSDS_G[2]_N
LCD_DATA9	GIN[6]	VDCE0_VO_DATA9	G[6]	RSDS_G[3]_P
LCD_DATA8	GIN[7]	VDCE0_VO_DATA8	G[7]	RSDS_G[3]_N
LCD_DATA7	BIN[0]	VDCE0_VO_DATA7	B[0]	RSDS_B[0]_P
LCD_DATA6	BIN[1]	VDCE0_VO_DATA6	B[1]	RSDS_B[0]_N
LCD_DATA5	BIN[2]	VDCE0_VO_DATA5	B[2]	RSDS_B[1]_P
LCD_DATA4	BIN[3]	VDCE0_VO_DATA4	B[3]	RSDS_B[1]_N
LCD_DATA3	BIN[4]	VDCE0_VO_DATA3	B[4]	RSDS_B[2]_P
LCD_DATA2	BIN[5]	VDCE0_VO_DATA2	B[5]	RSDS_B[2]_N
LCD_DATA1	BIN[6]	VDCE0_VO_DATA1	B[6]	RSDS_B[3]_P
LCD_DATA0	BIN[7]	VDCE0_VO_DATA0	B[7]	RSDS_B[3]_N
Pixel clock (LCD_CLK from video data channel selected by VDCECTL.VOSL)*1				
LCD_CLK	LCD_CLK	VDCE0_VO_CLKP	VDCE0_VO_CLKP	RSDS_CLK_P
		VDCE0_VO_CLKN	–	RSDS_CLK_N

Note 1. The clock signals are not affected by R – B swap and endian selection.

(d) VDCEn video output format converter settings (R – B swap, big endian)

– OUT_SET.OUT_FORMAT[1:0] = 00_B (RGB888 output)

– OUT_SET.OUT_ENDIAN_ON = 1 (big endian)

– OUT_SET.OUT_SWAP_ON = 1 (R – B swap)

Table 37.73 D1M2(H) Video output format selection (R – B swap, big endian)

VDCEn video output data	VDCEn video output format converter output	Port signal name	LVTTL	RSDS
Color data signals (LCD_DATA[23:0] from video data channel selected by VDCECTL.VOSL)				
LCD_DATA23	BIN[0]	VDCE0_VO_DATA23	B[0]	RSDS_B[0]_P
LCD_DATA22	BIN[1]	VDCE0_VO_DATA22	B[1]	RSDS_B[0]_N
LCD_DATA21	BIN[2]	VDCE0_VO_DATA21	B[2]	RSDS_B[1]_P
LCD_DATA20	BIN[3]	VDCE0_VO_DATA20	B[3]	RSDS_B[1]_N
LCD_DATA19	BIN[4]	VDCE0_VO_DATA19	B[4]	RSDS_B[2]_P
LCD_DATA18	BIN[5]	VDCE0_VO_DATA18	B[5]	RSDS_B[2]_N
LCD_DATA17	BIN[6]	VDCE0_VO_DATA17	B[6]	RSDS_B[3]_P
LCD_DATA16	BIN[7]	VDCE0_VO_DATA16	B[7]	RSDS_B[3]_N
LCD_DATA15	GIN[0]	VDCE0_VO_DATA15	G[0]	RSDS_G[0]_P
LCD_DATA14	GIN[1]	VDCE0_VO_DATA14	G[1]	RSDS_G[0]_N
LCD_DATA13	GIN[2]	VDCE0_VO_DATA13	G[2]	RSDS_G[1]_P
LCD_DATA12	GIN[3]	VDCE0_VO_DATA12	G[3]	RSDS_G[1]_N
LCD_DATA11	GIN[4]	VDCE0_VO_DATA11	G[4]	RSDS_G[2]_P
LCD_DATA10	GIN[5]	VDCE0_VO_DATA10	G[5]	RSDS_G[2]_N
LCD_DATA9	GIN[6]	VDCE0_VO_DATA9	G[6]	RSDS_G[3]_P
LCD_DATA8	GIN[7]	VDCE0_VO_DATA8	G[7]	RSDS_G[3]_N
LCD_DATA7	RIN[0]	VDCE0_VO_DATA7	R[0]	RSDS_R[0]_P
LCD_DATA6	RIN[1]	VDCE0_VO_DATA6	R[1]	RSDS_R[0]_N
LCD_DATA5	RIN[2]	VDCE0_VO_DATA5	R[2]	RSDS_R[1]_P
LCD_DATA4	RIN[3]	VDCE0_VO_DATA4	R[3]	RSDS_R[1]_N
LCD_DATA3	RIN[4]	VDCE0_VO_DATA3	R[4]	RSDS_R[2]_P
LCD_DATA2	RIN[5]	VDCE0_VO_DATA2	R[5]	RSDS_R[2]_N
LCD_DATA1	RIN[6]	VDCE0_VO_DATA1	R[6]	RSDS_R[3]_P
LCD_DATA0	RIN[7]	VDCE0_VO_DATA0	R[7]	RSDS_R[3]_N
Pixel clock (LCD_CLK from video data channel selected by VDCECTL.VOSL)				
LCD_CLK	LCD_CLK	VDCE0_VO_CLKP	VDCE0_VO_CLKP	RSDS_CLK_P
		VDCE0_VO_CLKN	–	RSDS_CLK_N

37.9.2.2 Timing Controller output signals assignment

The following table shows the relation between the VDECn TCON output signals and the port signal names.

Table 37.74 D1M2(H) Timing Controller output signals assignment

VDCEn video output data	VDCEn video output format converter output	Port signal name
LCD_TCON0	LCD_TCON0	VDCEn_VO_TCON0
LCD_TCON1	LCD_TCON1	VDCEn_VO_TCON1
LCD_TCON2	LCD_TCON2	VDCEn_VO_TCON2
LCD_TCON3	LCD_TCON3	VDCEn_VO_TCON3
LCD_TCON4	LCD_TCON4	VDCEn_VO_TCON4
LCD_TCON5	LCD_TCON5	VDCEn_VO_TCON5
LCD_TCON6	LCD_TCON6	VDCEn_VO_TCON6

37.9.3 D1M1A Video output configuration

37.9.3.1 D1M1A video output mode selection

Parallel 16-/18-/24-bit RGB, VODDR, Open LDI and Serial 8-bit RGB are assigned to the same alternative ports.

Therefore, these output functions must be selected by RSDSCFG register.

NOTE

P42_[15:11] functions (VDCE0_VO_TCONx) are not changed by RSDSCFG register.

Table 37.75 Pin assignment of P43_0-1, P44_0-11, P45_0-13 (1/2)

		Use case			
		Parallel RGB*1	**VODDR**	**Open LDI*2**	**Serial RGB*2*4**
Configuration					
RSDSCFG register	VODDR_OEN	0	1	0	0
	OLDI0_OEN	0	0	1	–
	SRGB1_EN	0	0	–*3	1
	Pin name	**Pin functions**			
1st alternative assignment	P43_0	VDCE0_TCON2 (HS)	VODDR_TCON2 (HS)		VDCE1_VO_TCON2 (HS)
	P43_1	VDCE0_TCON0 (VS)	VODDR_TCON0 (VS)		VDCE1_VO_TCON0 (VS)
	P44_0	VDCE0_VO_DATA23	VODDR_LCDOUT23		VDCE1_VO_DATA7
	P44_1	VDCE0_VO_DATA22	VODDR_LCDOUT22		VDCE1_VO_DATA6
	P44_2	VDCE0_VO_DATA21	VODDR_LCDOUT21		VDCE1_VO_DATA5
	P44_3	VDCE0_VO_DATA20	VODDR_LCDOUT20		VDCE1_VO_DATA4
	P44_4	VDCE0_VO_DATA19	VODDR_LCDOUT19		VDCE1_VO_DATA3
	P44_5	VDCE0_VO_DATA18	VODDR_LCDOUT18		VDCE1_VO_DATA2
	P44_6	VDCE0_VO_DATA17	VODDR_LCDOUT17		VDCE1_VO_DATA1
	P44_7	VDCE0_VO_DATA16	VODDR_LCDOUT16		VDCE1_VO_DATA0
	P44_8	VDCE0_VO_DATA15	VODDR_LCDOUT15		VDCE1_VO_CLK
	P44_9	VDCE0_VO_DATA14	VODDR_LCDOUT14		
	P44_10	VDCE0_VO_DATA13	VODDR_LCDOUT13		
	P44_11	VDCE0_VO_DATA12	VODDR_LCDOUT12		
	P45_0	VDCE0_VO_CLKP	VODDR_OUT0_CLK	OLDI0_CH0_CLKP	
	P45_1	VDCE0_VO_CLKN	VODDR_OUT1_CLK	OLDI0_CH0_CLKN	
	P45_2	VDCE0_VO_DATA11	VODDR_LCDOUT11	OLDI0_CH1_P	
	P45_3	VDCE0_VO_DATA10	VODDR_LCDOUT10	OLDI0_CH1_N	
	P45_4	VDCE0_VO_DATA9	VODDR_LCDOUT9	OLDI0_CH2_P	
	P45_5	VDCE0_VO_DATA8	VODDR_LCDOUT8	OLDI0_CH2_N	
	P45_6	VDCE0_VO_DATA7	VODDR_LCDOUT7	OLDI0_CH3_P	
	P45_7	VDCE0_VO_DATA6	VODDR_LCDOUT6	OLDI0_CH3_N	
	P45_8	VDCE0_VO_DATA5	VODDR_LCDOUT5	OLDI0_CH4_P	
	P45_9	VDCE0_VO_DATA4	VODDR_LCDOUT4	OLDI0_CH4_N	
	P45_10	VDCE0_VO_DATA3	VODDR_LCDOUT3		
	P45_11	VDCE0_VO_DATA2	VODDR_LCDOUT2		
	P45_12	VDCE0_VO_DATA1	VODDR_LCDOUT1		
	P45_13	VDCE0_VO_DATA0	VODDR_LCDOUT0		

Table 37.75 Pin assignment of P43_0-1, P44_0-11, P45_0-13 (2/2)

		Use case			
		Parallel RGB*1	VODDR	Open LDI*2	Serial RGB*2*4
2nd alternative assignment	P43_0	VDCE0_VO_TCON3 (DE)	VODDR_VO_TCON3 (DE)*5		VDCE1_VO_TCON3 (DE)

Note 1. VDCE0 and VDCE1 output can be swapped by VDCECTL.VOSL.

Note 2. Pins not used for the selected video output mode can be used for other functions.

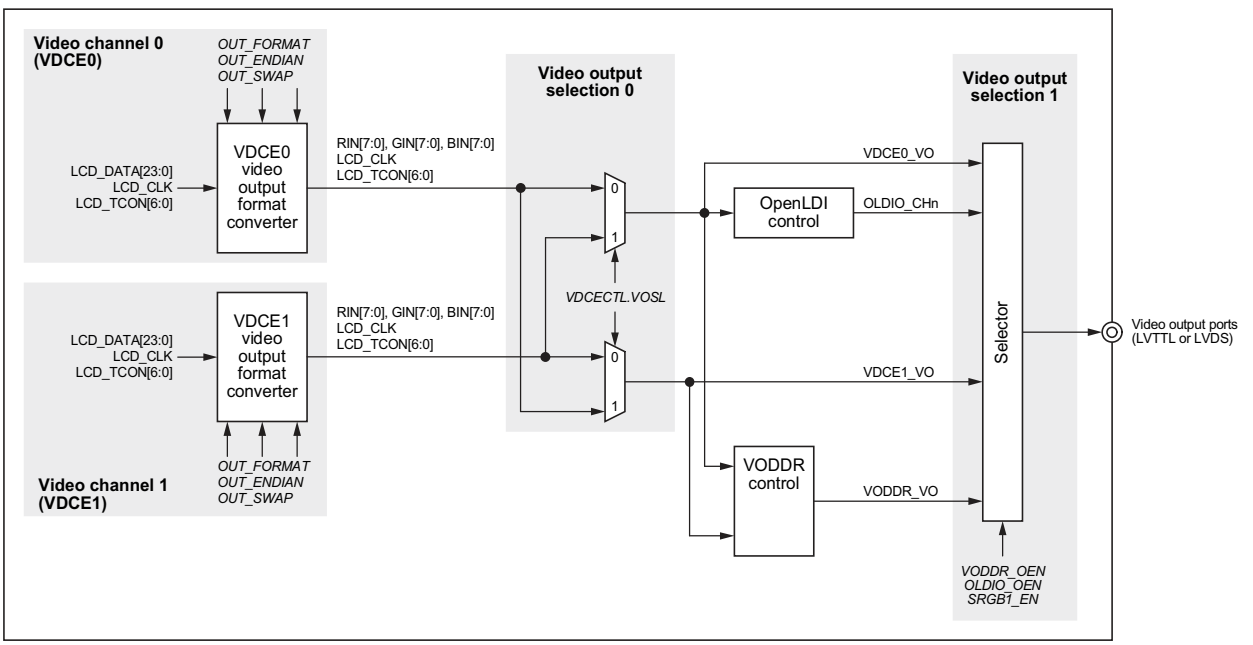
Note 3. When OpenLDI is not used, OLDI0_OEN must be set to 0.

Note 4. Serial RGB is only available for VDCE1.

Note 5. Another alternative, when all 3 sync signals (HS+VS and DE) are required for VODDR at the same time, is to use VDCE0_VO_TCON3 (DE) on P42_11 (AF1 Out). This port pin selection is not controlled by the RSDSCFG register.

37.9.3.2 D1M1A Video output selection

The diagram below illustrates the video output selection options.



Note 1. The Video Data Controllers VDCEn include facilities to change the mapping of the color output data LCD_DATA[23:0] to the VDCEn output data RIN[7:0], GIN[7:0] and BIN[7:0]. Refer to Section 38, Video Data Controller E (VDCE) for details about the OUT_FORMAT, OUT_ENDIAN and OUT_SWAP settings.

Note 2. When VODDR is used, VDCECTL.VOSL must be set to 0.

Figure 37.22 D1M1A video output selections

By use of the VDCECTL.VOSL selector the output signals from the video channels 0 and 1 can be swapped, so that any of both can be used for the OpenLDI output. The other video output is output as LVTTL signals.

Table 37.76 D1M1A video output selections

VDCECTL.	RSDSCFG.			
VOSL	VODDR _OEN	OLDIO _OEN	SRGB1 _OEN	Video output ports
0	0	0	0	Video Channel 0: Parallel RGB
	1			VODDR: VODDR output
	0	1	– *1	OpenLDI (Video Channel 0): OpenLDI (LVDS) output
		– *1	1	Video Channel 1: Serial RGB
1	0	0	0	Video Channel 1: Parallel RGB
	1			VODDR: VODDR output
	0	1	– *1	OpenLDI (Video Channel 1): OpenLDI (LVDS) output

Note 1. OpenLDI output ports and Serial RGB ports from VDCE1_VO are not overlapping.

37.9.4 D1M1-V2 Video output configuration

37.9.4.1 D1M1-V2 video output mode selection

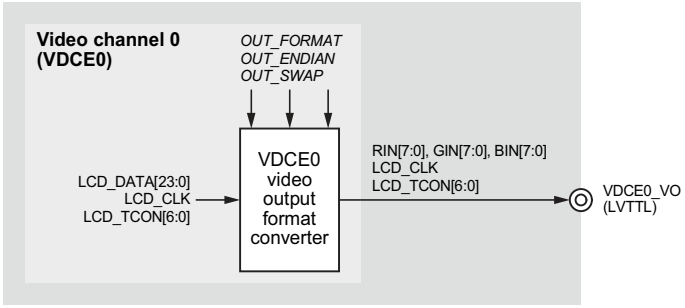
Parallel 16-/18-/24-bit RGB and Serial 8-bit RGB are assigned to the same alternative ports.
These output function must be selected by the OUT_SET.OUT_FRQ_SEL[1:0] register.

Table 37.77 Pin assignment of P44_0-11, P45_0-13

		Use case	
		Parallel RGB	**Serial RGB**
Configuration			
OUT_SET register	OUT_FRQ_SEL[1:0]	0	1, 2
	Pin name	**Pin functions**	
1st alternative assignment	P44_0	VDCE0_VO_DATA23	
	P44_1	VDCE0_VO_DATA22	
	P44_2	VDCE0_VO_DATA21	
	P44_3	VDCE0_VO_DATA20	
	P44_4	VDCE0_VO_DATA19	
	P44_5	VDCE0_VO_DATA18	
	P44_6	VDCE0_VO_DATA17	
	P44_7	VDCE0_VO_DATA16	
	P44_8	VDCE0_VO_DATA15	
	P44_9	VDCE0_VO_DATA14	
	P44_10	VDCE0_VO_DATA13	
	P44_11	VDCE0_VO_DATA12	
	P45_0	VDCE0_VO_CLKP	VDCE0_VO_CLKP
	P45_1	VDCE0_VO_CLKN	VDCE0_VO_CLKN
	P45_2	VDCE0_VO_DATA11	
	P45_3	VDCE0_VO_DATA10	
	P45_4	VDCE0_VO_DATA9	
	P45_5	VDCE0_VO_DATA8	
	P45_6	VDCE0_VO_DATA7	VDCE0_VO_DATA7
	P45_7	VDCE0_VO_DATA6	VDCE0_VO_DATA6
	P45_8	VDCE0_VO_DATA5	VDCE0_VO_DATA5
	P45_9	VDCE0_VO_DATA4	VDCE0_VO_DATA4
	P45_10	VDCE0_VO_DATA3	VDCE0_VO_DATA3
	P45_11	VDCE0_VO_DATA2	VDCE0_VO_DATA2
	P45_12	VDCE0_VO_DATA1	VDCE0_VO_DATA1
	P45_13	VDCE0_VO_DATA0	VDCE0_VO_DATA0

37.9.4.2 D1M1-V2 Video output selection

The diagram below illustrates the video output selection options.



Note: The Video Data Controllers VDCEn include facilities to change the mapping of the color output data LCD_DATA[23:0] to the VDCEn output data RIN[7:0], GIN[7:0] and BIN[7:0]. Refer to Section 38, Video Data Controller E (VDCE) for details about the OUT_FORMAT, OUT_ENDIAN and OUT_SWAP settings.

Figure 37.23 D1M1-V2 video output signals

37.9.5 Video output clock inversion in LVTTL mode

If the video data VDCEn_VO_DATA[23:0] is output in LVTTL mode, the video data changes at the rising edge of the video output clock VDCEn_VO_CLK.

By use of the port output inversion function of the port control the video output clock can be inverted, thus the video data changes at the falling edge of the video output clock.

- VDCE0_VO_CLK
  - PINV45.PINV45_0 = 0: not inverted
    VDCE0_VO_DATA[23:0] changes at the rising edge of VDCE0_VO_CLK
  - PINV45.PINV45_0 = 1: inverted
    VDCE0_VO_DATA[23:0] changes at the falling edge of VDCE0_VO_CLK
- VDCE1_VO_CLK
  - PINV47.PINV47_8 = 0: not inverted
    VDCE1_VO_DATA[23:0] changes at the rising edge of VDCE1_VO_CLK
  - PINV47.PINV47_8 = 1: inverted
    VDCE1_VO_DATA[23:0] changes at the falling edge of VDCE1_VO_CLK

37.9.6 D1M2(H) RSDS control

37.9.6.1 Phase shift

RSDSCFG.RPHSL[1:0] selects a phase shift in 90 ° steps between the RSDS output clock VDCE0_VO_CLKP and the RSDS data output.

While RSDS data are output to terminal with 1 cycle delay from VDCE output, TCON signals are not delayed to terminal.
Therefore, when RSDS output is used, TCON signals should be delayed for 1 cycle. For this adjustment, please refer to Section 38.8.1.9, LCD TCON.

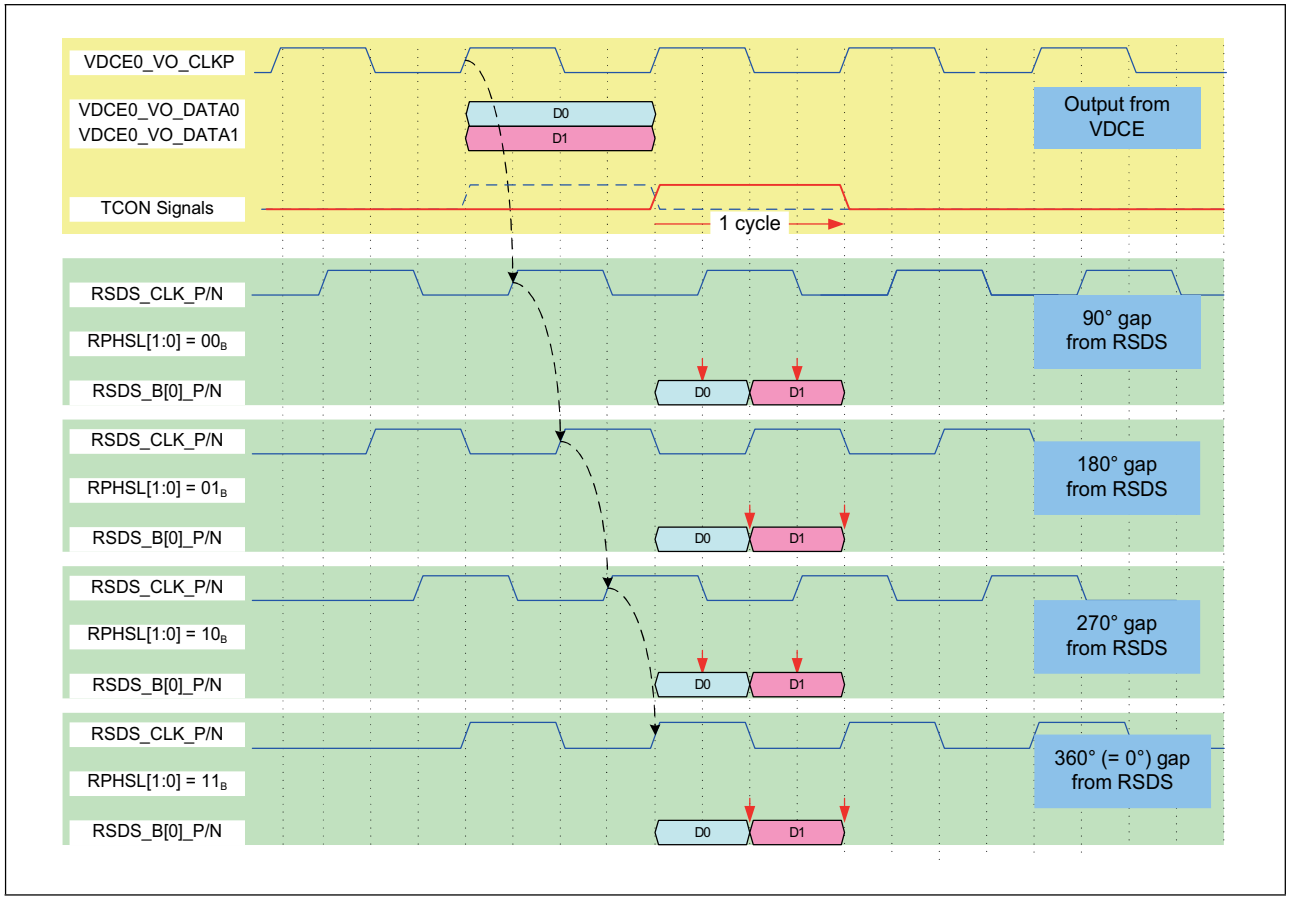


Figure 37.24 RSDS clock phase shift

37.9.6.2 RSDS data bit swap

RSDSCFG.RBSW selects a bit swap of the bits on a single RSDS data line:

- RBSW = 0: no swap, even number bit (i.e. data bits 0, 2, 4, 6) first
- RBSW = 1: swap, odd number bit (i.e. data bits 1, 3, 5, 7) first

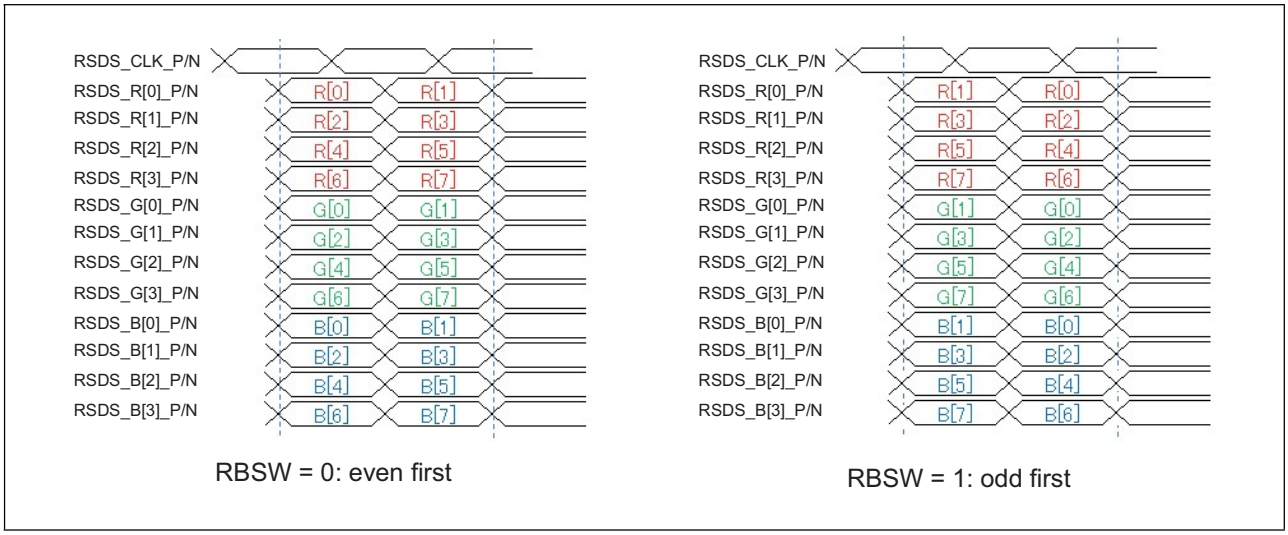


Figure 37.25 RSDS data bitswap

37.9.6.3 RSDS setup

1. Select alternative port function:
   - PMC44.PM44_[11:0] = FFF_H
   - PMC45.PM45_[13:0] = $3FFF_H$
2. Set ports as output:
   - PM44.PM44_[11:0] = FFF_H
   - PM45.PM45_[13:0] = $3FFF_H$
3. Set RSDSCFG.ROEN = 1 to select RSDS output
4. Set RSSCFG.RPHSL[1:0] and RSSCFG.RBSW as required.
5. Wait 10 µsec RSDS buffer stabilization time.

37.10 Video and graphics functions control registers

Table 37.78 List of video and graphics functions control registers

Module Name	Register Name	Symbol	Address
SELB	VDCE control registers	VDCECTL	FFC0 601C$_H$
SELB	RSDS configuration register	RSDSCFG	FFC0 6020$_H$
SELB	Sprite Engine update timing control register	SPEAUPDEN	FFC0 6048$_H$

NOTES

1. In the header files the names of the above registers are defined in the following format: <ModuleName> + <Symbol>.
   <ModuleName> and <Symbol> are defined in the above table.
2. Memory access right after register access has possibility to cause the racing between two accesses via different routes.
   Therefore, the last accessed register value must be confirmed at least before memory access. Refer to Section 3.4.1, Synchronization of Store Instruction Completion and Subsequent Instruction Generation for details.

37.10.1 Register details

37.10.1.1 VDCECTL — VDCE control register

Access: This register can be read/written in 32-bit units.

Address: FFC0 601C$_H$

Initial value: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0*1
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	0	0	0	0	0	0	0	0	0	VI1CTL	UPDT1	UPDT0	PXSL	VOSL	VISL1	VISL0
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Note 1. The initial value “0” of bit 16 must be changed to “1” and must not be changed afterwards.

Table 37.79 VDCECTL register contents

Bit Position	Bit Name	Function
31 to 17	Reserved	When written, write the initial value.
16	–	**CAUTION** The initial value “0” of bit 16 must be changed to “1” and must not be changed afterwards.
15 to 7	Reserved	When written, write the initial value.
6	VI1CTL	Selection of C_ISO_VI1PIXCLK (for D1M2(H) devices only) 0: C_ISO_VI1PIXCLK = VDCE1_VI_CLK 1: C_ISO_VI1PIXCLK = C_ISO_MIPIPIXCLK
5	UPDT1	Sprite Engine software update control This bit is used to generate the VUPDATE1 signal towards the Sprite Engine in sprite definition registers in software mode.
4	UPDT0	Sprite Engine software update control This bit is used to generate the VUPDATE0 signal towards the Sprite Engine in sprite definition registers in software mode.
3	PXSL	Video Input selection for VDCE0 (D1M2(H) only) 0: LVTTL 1: MIPI
2	VOSL	Video Output swap (for D1M2(H) and D1M1A devices only) 0: VDCE0 -> VO0 (LVTTL/RSDS) VDCE1 -> VO1 (LVTTL) 1: VDCE0 -> VO1 (LVTTL) VDCE1 -> VO0 (LVTTL/RSDS)
1	VISL1	Video Input color format for video channel 1 (for D1M2(H) devices only) 0: 8-bit ITU656 1: 24-bit RGB888
0	VISL0	Video Input color format for video channel 0 0: 8-bit ITU656 1: 24-bit RGB888

37.10.1.2 RSDSCFG — RSDS and other video output control register

Access: This register can be read/written in 32-bit units.

Address: FFC0 6020$_H$

Initial value: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	0	0	0	0	0	0	0	SRGB1_OEN	0	0	OLDI_OEN	VODDR_OEN	ROEN	RPHSL[1:0]		RBSW
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Table 37.80 RSDSCFG register contents

Bit Position	Bit Name	Function
31 to 9	Reserved	When written, write the initial value.
8	SRGB1_OEN	VDCE1 serial RGB output enabled/disable (D1M1A only) 0: VDCE1 serial RGB output disabled 1: VDCE1 serial RGB enabled **Note:** When this bit is set to 1, VODDR_OEN must be set to 0.
7 to 6	Reserved	When written, write the initial value.
5	OLDI_OEN	Open LDI output enabled/disable (D1M1A only) 0: Open LDI output disabled 1: Open LDI output enabled **Note:** When this bit is set to 1, VODDR_OEN must be set to 0. When Open LDI is not used, OLDI0_OEN must be set to 0.
4	VODDR_OEN	VODDR output enabled/disable (D1M1A only) VODDR_OEN selects the clock source for the Video Output Interface n (VDCEn) 0: VODDR output disabled VDCE0 source clock: CKSC_IVDCE0VOS_CTL output VDCE1 source clock: CKSC_IVDCE1VOS_CTL output 1: VODDR output enabled VDCE0 source clock: VODDR output VODDR_PIXCLK0 VDCE1 source clock: VODDR output VODDR_PIXCLK1 About clock generation from VODDR refer to Section 37.13, Video Output DDR format converter (VODDR) (D1M1A only). **Note:** When this bit is set to 1, SRGB1_OEN and OLDI0_OEN must be set to 0.
3	ROEN	RSDS output enabled/disable (D1M2(H) only) 0: RSDS output disabled (LVTTL output enabled) 1: RSDS output enabled (LVTTL output disabled)
2, 1	RPHSL[1:0]	RSDS output clock phase shift selection (D1M2(H) only) 00_B: 90° 01_B: 180° 10_B: 270° 11_B: 360°

Table 37.80 RSDSCFG register contents

Bit Position	Bit Name	Function
0	RBSW	RSDS bit swap selection (D1M2(H) only) 0: even bit first 1: odd bit first

37.10.1.3 SPEAUPDEN — Sprite Engine update timing control register

Access: This register can be read/written in 32-bit units.

Address: FFC0 6048_H

Initial value: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	UPDEN[15:0]															
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 37.81 SPEAUPDEN register contents

Bit Position	Bit Name	Function
31 to 16	Reserved	When written, write the initial value.
15 to 0	UPDEN[15:0]	Each of these bits enable the update of the Sprite Engine registers by the respective VE signal. 0: update by VE signal disabled 1: update by VE signal enabled Following the assignment of the bits to the VE sources • VDCE0 VE sources UPDEN0: VE of Image Synthesizer 00 UPDEN1: VE of Image Synthesizer 01 UPDEN2: VE of Image Synthesizer 02 UPDEN3: VE of Image Synthesizer 03 UPDEN4: VE of Output Image Generator UPDEN5: VE after alpha blending of Image Synthesizer 02 and Image Synthesizer 03 • VDCE1 VE sources (D1M2(H) and D1M1A only) UPDEN8: VE of Image Synthesizer 10 UPDEN9: VE of Image Synthesizer 11 UPDEN10: VE of Image Synthesizer 12 UPDEN11: VE of Image Synthesizer 13 UPDEN13: VE after alpha blending of Image Synthesizer 12 and Image Synthesizer 13 UPDEN[6,7,12,14,15]: Reserved

37.11 Video Output Warping Engine (VOWE) Ring Buffer

The Video Output Warping Engine (VOWE) writes rendered data to the memory. The Output Image Renderer (OIR) reads this data and forwards it to the Video Output module.

Since both, writing to the memory by the Video Output Warping Engine and reading the data by the Output Image Renderer is performed on the same timebase, which is determined by the frame geometry, framerate and pixel clock, it is in most cases not necessary to reserve memory for a complete output frame, since the OIR reads the VOWE data already before a complete frame has been stored.

A Ring Buffer in the memory that holds enough VOWE data for the OIR 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.

The Ring Buffer location in the memory is defined by its base address and mask.

The Ring Buffer base address RB_BASE_ADDR is defined by VOWEMAC.RBBADR[31:10].
It must be aligned to 1 KB boundaries, thus the lower 9 bit of the base address have to be all 0.

The mask is defined by VOWEMMC.RBMASK[31:0] = RB_SIZE - 1.
RB_SIZE denotes the Ring Buffer size.

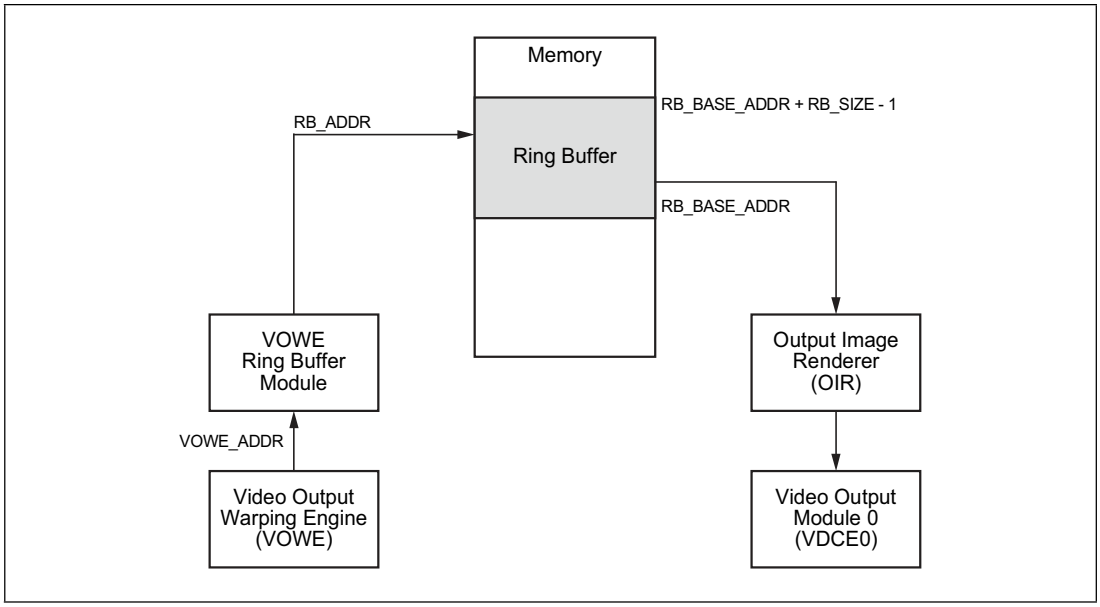


Figure 37.26 VOWE Ring Buffer module

The Ring Buffer address RB_ADDR, that is the address to write data the buffer in the memory, is calculated by

RB_ADDR = RB_BASE_ADDR + (VOWE_ADDR mod RB_SIZE)

VOWE_ADDR denotes the write address, issued by the VOWE.
“mod” denotes the modulo operator.

Example:

- horizontal size: 2048 32-bit pixel -> line stride = 2048 x 4 byte = 8 KB
- Ring Buffer shall hold 16 lines:
  -> RB_SIZE = 16 x 8 KB = 128 KB = 0002 0000_H
  -> VOWEMMC.RBMASK[31:0] = RB_SIZE - 1 = 0001 $FFFF_H$
- Ring Buffer base address RB_BASE_ADDR = VOWEMAC.RBBADR[31:0] = 4001 0000_H

If VOWE_ADDR = 1234 5670_H:

RB_ADDR = 4001 0000_H + (1234 5678_H mod 0002 0000_H)
= 4001 0000_H + 0000 5678_H = 4001 5678_H

Output Image Renderer loop addressing

The Output Image Renderer features a loop addressing function, that is to be used when reading the data from the Ring Buffer. Refer to Section 38.7, Output Image Generator for details.

37.11.1 VOWE Ring Buffer control registers

Table 37.82 List of the VOWE Ring Buffer control registers

Module Name	Register Name	Symbol	Address
SELB	VOWE Ring Buffer address offset register	VOWEMAC	FFC0 6040_H
SELB	VOWE Ring Buffer mask register	VOWEMMC	FFC0 6044_H

NOTES

1. In the header files the names of the above registers are defined in the following format:
   <ModuleName> + <Symbol>.
   <ModuleName> and <Symbol> are defined in the above table.
2. Memory access right after register access has possibility to cause the racing between two accesses via different routes.
   Therefore, the last accessed register value must be confirmed at least before memory access. Refer to Section 3.4.1, Synchronization of Store Instruction Completion and Subsequent Instruction Generation for details.

37.11.2 Register details

37.11.2.1 VOWEMAC — VOWE Ring Buffer base address register

Access: This register can be read/written in 32-bit units.

Address: FFC0 6040_H

Initial value: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RBBADR[31:16]															
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RBBADR[15:10]						0	0	0	0	0	0	0	0	0	0
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R

Table 37.83 VOWEMAC register contents

Bit Position	Bit Name	Function
31 to 10	RBBADR[31:10]	Video Output Warping Engine Ring Buffer base address It must be aligned to 1 KB boundaries, thus the lower 10 bits of the base address are always 0.
9 to 0	Reserved	When written, write the initial value.

37.11.2.2 VOWEMMC — VOWE Ring Buffer mask register

Access: This register can be read/written in 32-bit units.

Address: FFC0 6044_H

Initial value: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RBMASK[31:16]															
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RBMASK[15:0]															
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 37.84 VOWEMMC register contents

Bit Position	Bit Name	Function
31 to 0	RBMASK[31:0]	Video Output Warping Engine Ring Buffer size RBMASK[31:0] = Ring Buffer size - 1

37.12 Open LDI Interface (D1M1A only)

37.12.1 Overview of Open LDI Interface

37.12.1.1 Units

This microcontroller has the following number of units of the Open LDI Interface.

Table 37.85 Number of Units

Product Name	D1L1	D1L2(H)	D1M1(H) D1M1-V2	D1M1A	D1M2(H)
Number of Units	–	–	–	1	–
Name	–	–	–	OLDIn (n = 0)	–

37.12.1.2 Indices

Following indices are used in this section:

Table 37.86 Indices

Index	Meaning
n	The individual Open LDI Interface units are generically indicated by the index “n”.

37.12.1.3 Register addresses

All OLDIn register addresses are given as address offsets from the individual base addresses <OLDIn_base>.

These base addresses are listed in the following table.

Table 37.87 Register base addressed <OLDIn_base>

OLDIn unit	<OLDIn_base> address
OLDI0	F200 2000H

37.12.1.4 Clock supply

All Open LDI Interfaces provide following clock inputs.

Table 37.88 Clock supply

OLDIn unit	OLDIn clock	Connected to
OLDI0	APB bus clock PCLK	Clock Controller C_ISO_PCLK
	Operation clock OLDICLK	Clock Controller CKSC_IVOEXS_CTL

37.12.1.5 Interrupts

The Open LDI Interface units do not generate any interrupt requests.

37.12.1.6 Reset sources

The Open LDI and their registers are initialized by the following reset signal:

Table 37.89 Reset sources

OLDIn unit	Reset signal
OLDI0	Reset Controller SYSRES

37.12.1.7 External Input/Output Signals

The external input/output signals of Open LDI are listed below.

Table 37.90 Video output signals connection

OLDIn signal	Function	Connected to *1
OLDI0_CH0_CLKP	OLDI0 differential clock (pos)	Video output mode selection OLDI0_CH0_CLKP
OLDI0_CH0_CLKN	OLDI0 differential clock (neg)	Video output mode selection OLDI0_CH0_CLKN
OLDI0_CH1_P	OLDI0 differential Ch1 data (pos)	Video output mode selection OLDI0_CH1_P
OLDI0_CH1_N	OLDI0 differential Ch1 data (neg)	Video output mode selection OLDI0_CH1_N
OLDI0_CH2_P	OLDI0 differential Ch2 data (pos)	Video output mode selection OLDI0_CH2_P
OLDI0_CH2_N	OLDI0 differential Ch2 data (neg)	Video output mode selection OLDI0_CH2_N
OLDI0_CH3_P	OLDI0 differential Ch3 data (pos)	Video output mode selection OLDI0_CH3_P
OLDI0_CH3_N	OLDI0 differential Ch3 data (neg)	Video output mode selection OLDI0_CH3_N
OLDI0_CH4_P	OLDI0 differential Ch4 data (pos)	Video output mode selection OLDI0_CH4_P
OLDI0_CH4_N	OLDI0 differential Ch4 data (neg)	Video output mode selection OLDI0_CH4_N

Note 1. The video output signals can be selected from several kinds of mode. Refer to Section 37.9.3.1, D1M1A video output mode selection.

37.12.2 Open LDI Overview

The Open LDI module converts an RGB signal output by the Video Output Controller E Unit module to the Open LDI format and outputs those signals.

The Open LDI interface supports 8 output data formats with the conversion formats selected by register settings. The output control signals can also be selected freely.

37.12.2.1 Features

This module has the following features.

- Output pins:
  Five differential output pairs (4 data and 1 clock) that conform to the TIA/EIA-644 standard.
- Operating frequency: maximum operation clock (OLDICLK) frequency 240 MHz
- Maximum dot clock frequency of 34 MHz
- Supports eight output data formats

37.12.2.2 Block Diagram

The figure below shows the block diagram of the Open LDI module.

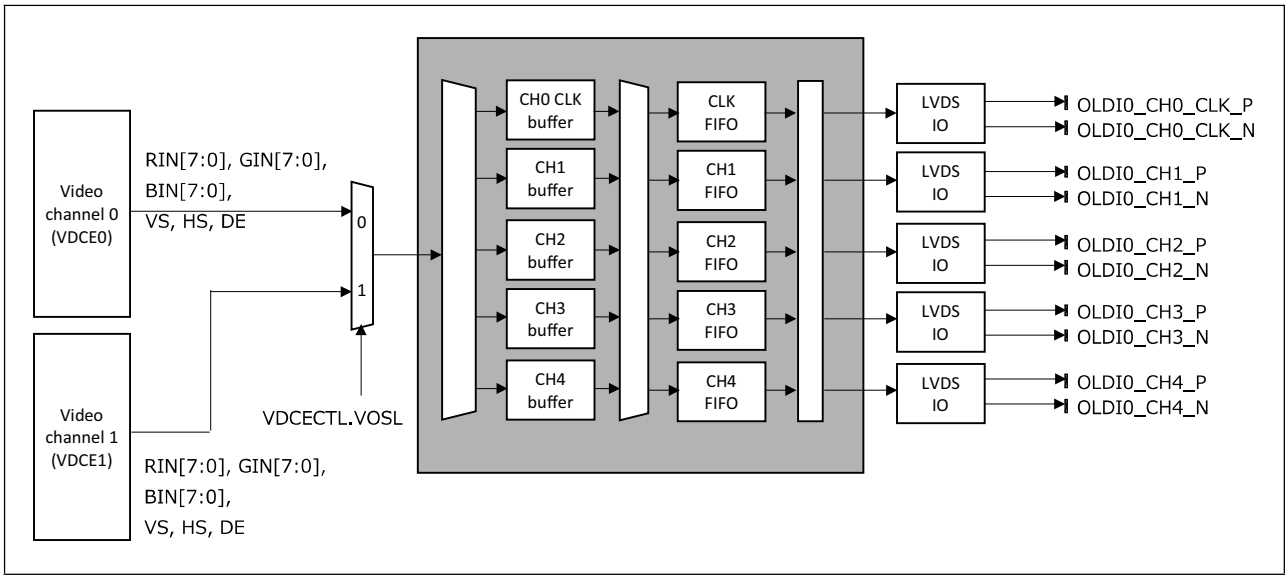


Figure 37.27 Open LDI Block Diagra

Table 37.91 Open LDI selections

OLDIn unit	VDCECTL.VOSL	Function
OLDI0	0	Video channel 0 (VDCE0)
	1	Video channel 1 (VDCE1)

37.12.2.3 Output signals

Table 37.92 Output signals

Pin Name	I/O	Connected to
OLDI0_CH0_CLKP	Output	OLDI0 differential clock (pos)
OLDI0_CH0_CLKN	Output	OLDI0 differential clock (neg)
OLDI0_CH1_P	Output	OLDI0 differential Ch1 data (pos)
OLDI0_CH1_N	Output	OLDI0 differential Ch1 data (neg)
OLDI0_CH2_P	Output	OLDI0 differential Ch2 data (pos)
OLDI0_CH2_N	Output	OLDI0 differential Ch2 data (neg)
OLDI0_CH3_P	Output	OLDI0 differential Ch3 data (pos)
OLDI0_CH3_N	Output	OLDI0 differential Ch3 data (neg)
OLDI0_CH4_P	Output	OLDI0 differential Ch4 data (pos)
OLDI0_CH4_N	Output	OLDI0 differential Ch4 data (neg)

The state of the Open LDI pin is determined by the OLDInCR register.

37.12.3 Register Description

The table below lists the register configuration.

Register Access Size

All registers can be accessed in 32-bit units.

Correct device operation is not guaranteed if any access size other than 32-bit is used to access these registers.

<OLDIn_base>

The base address <OLDIn_base> of the OLDIn is defined in section 37.12.1.3, Register addresses.

Table 37.93 Register Configuration

Register Name	Symbol	Address
OLDI control register 0	OLDInCR0	<OLDIn_base> + 00_H
OLDI control register 1	OLDInCR1	<OLDIn_base> + 04_H
OLDI CTR control register	OLDInCTRCR	<OLDIn_base> + $0C_H$
OLDI control register	OLDInCHCR	<OLDIn_base> + 10_H
OLDI skew control register	OLDInSKEWCTR	<OLDIn_base> + 70_H

37.12.3.1 OLDInCR0 – Open LDI control register 0

This register controls Open LDI function.

Access: This register can be read/written in 32-bit units.

Address: <OLDIn_base> + 00_H

Initial value: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	–	–	–	–	LVMD[3:0]				–	–	–	–	–	–	LVEN	LVRES
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R/W	R/W

Table 37.94 OLDInCR0 register contents

Bit position	Bit name	Function
31 to 12	–	These bits are always read as 0. When written, write the initial value.
11 to 8	LVMD[3:0]	OpenLDI Mode Selects the Open LDI module output data format (see Figure 37.28, Output Data Format). 0000: MODE0 0001: MODE1 0010: MODE2 0011: MODE3 0100: MODE4 0101: MODE5 0110: MODE6 0111: MODE7 All other values: Setting prohibited
7 to 2	–	These bits are always read as 0. When written, write the initial value.
1	LVEN	Open LDI Enable Bit Controls the Open LDI operation. 0: Stopped 1: Normal operation Once Open LDI operation is started, do not change this bit to 0.
0	LVRES	Open LDI Reset Bit Controls the Open LDI output. 0: Output off 1: Output on Once Open LDI operation is started, do not change this bit to 0.

37.12.3.2 OLDInCR1 – Open LDI control register 1

This register controls Open LDI function.

Access: This register can be read/written in 32-bit units.

Address: <OLDIn_base> + 04_H

Initial value: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	–	–	–	–	–	–	CH4STBY[1:0]		CH3STBY[1:0]		CH2STBY[1:0]		CH1STBY[1:0]		CLKSTBY[1:0]	
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 37.95 OLDInCR1 register contents

Bit position	Bit name	Function
31 to 10	–	These bits are always read as 0. When written, write the initial value.
9 to 8	CH4STBY[1:0]	CH4 Control Controls the CH4 pin. 00: Standby mode 01: Output off mode 11: Operating mode Other settings are prohibited. Once Open LDI operation is started, do not change this bit0 to 0.
7 to 6	CH3STBY[1:0]	CH3 Control Controls the CH3 pin. 00: Standby mode 01: Output off mode 11: Operating mode Other settings are prohibited. Once Open LDI operation is started, do not change this bit0 to 0.
5 to 4	CH2STBY[1:0]	CH2 Control Controls the CH2 pin. 00: Standby mode 01: Output off mode 11: Operating mode Other settings are prohibited. Once Open LDI operation is started, do not change this bit0 to 0.
3 to 2	CH1STBY[1:0]	CH1 Control Controls the CH1 pin. 00: Standby mode 01: Output off mode 11: Operating mode Other settings are prohibited. Once Open LDI operation is started, do not change this bit0 to 0.
1 to 0	CLKSTBY[1:0]	CLK Control Controls the CLK pin. 00: Standby mode 01: Output off mode 11: Operating mode Other settings are prohibited. Once Open LDI operation is started, do not change this bit0 to 0.

37.12.3.3 OLDInCTRCR – Open LDI CTR control register

This register controls Open LDI function.

Access: This register can be read/written in 32-bit units.

Address: <OLDIn_base> + $0C_H$

Initial value: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	–	–	–	–	–	CTR2SEL[2:0]			–	CTR1SEL[2:0]			–	CTR0SEL[2:0]		
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

Table 37.96 OLDInCTRCR register contents

Bit position	Bit name	Function
31 to 11	–	These bits are always read as 0. When written, write the initial value.
10 to 8	CTR2SEL[2:0]	Ctrl2 Select Selects data to be output to Ctrl2 (see section 37.12.4.3, CH Selection). 000: DE 011: HSYNC 100: VSYNC Other settings are prohibited.
7	–	These bits are always read as 0. When written, write the initial value.
6 to 4	CTR1SEL[2:0]	Ctrl1 Select Selects data to be output to Ctrl1 (see section 37.12.4.3, CH Selection). 000: VSYNC 001: DE 100: HSYNC Other settings are prohibited.
3	–	These bits are always read as 0. When written, write the initial value.
2 to 0	CTR0SEL[2:0]	Ctrl0 Select Selects data to be output to Ctrl0 (see section 37.12.4.3, CH Selection). 000: HSYNC 001: VSYNC 010: DE Other settings are prohibited.

37.12.3.4 OLDInCHCR – Open LDI CH control register

This register controls Open LDI function.

Access: This register can be read/written in 32-bit units.

Address: <OLDIn_base> + 10_H

Initial value: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	–	–	CH4SEL[1:0]		–	–	CH3SEL[1:0]		–	–	CH2SEL[1:0]		–	–	CH1SEL[1:0]	
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W

Table 37.97 OLDInCHCR register contents

Bit position	Bit name	Function
31 to 14	–	These bits are always read as 0. When written, write the initial value.
13 to 12	CH4SEL[1:0]	CH4 Select Selects data to be output to CH4 (see section 37.12.4.3, CH Selection). 00: CH4 01: CH1 10: CH2 11: CH3
11 to 10	–	These bits are always read as 0. When written, write the initial value.
9 to 8	CH3SEL[1:0]	CH2 Select Selects data to be output to CH3 (see section 37.12.4.3, CH Selection). 00: CH3 01: CH4 10: CH1 11: CH2
7 to 6	–	These bits are always read as 0. When written, write the initial value.
5 to 4	CH2SEL[1:0]	CH2 Select Selects data to be output to CH2 (see section 37.12.4.3, CH Selection). 00: CH2 01: CH3 10: CH4 11: CH1
3 to 2	–	These bits are always read as 0. When written, write the initial value.
1 to 0	CH1SEL[1:0]	CH1 Select Selects data to be output to CH1 (see section 37.12.4.3, CH Selection). 00: CH1 01: CH2 10: CH3 11: CH4

37.12.3.5 OLDInSKEWCTR – Open LDI skew control register

This register controls Open LDI function.

Access: This register can be read/written in 32-bit units.

Address: <OLDIn_base> + 70_H

Initial value: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	–	SKEWC[2:0]			SKEW4[2:0]			SKEW3[2:0]			SKEW2[2:0]			SKEW1[2:0]		
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 37.98 OLDInSKEWCTR register contents

Bit position	Bit name	Function
31 to 15	–	These bits are always read as 0. When written, write the initial value.
14 to 12	SKEWC[2:0]	Setting Clock Skew. 011 : +150 ps 010 : +100 ps 001 : +50 ps x00 : 0 ps 101 : - 50 ps 110 : - 100 ps 111 : - 150 ps
11 to 9	SKEW2[2:0]	Setting CH4 Skew.
8 to 6	SKEW3[2:0]	Setting CH3 Skew.
5 to 3	SKEW2[2:0]	Setting CH2 Skew.
2 to 0	SKEW1[2:0]	Setting CH1 Skew.

37.12.4 Operation

The Open LDI module converts the RGB signals output from a Video Data Controller E (VDCE) Unit to the Open LDI format and outputs those signals. Since the output data format can be selected by register settings, this module can convert to any of the Open LDI formats. The output data format is determined by (1) the mode selection, (2) the Ctrl signal selection, and (3) the CH selection.

The registers concerned with the output data format must be set before the Open LDI module is started and must not be changed during module operation.

37.12.4.1 Mode Selection

The mode is selected by the LVMD[3:0] bits in the Open LDI control register 0. **Figure 37.28** shows the output data format.

Here, R0 to R7, G0 to G7, and B0 to B7 are the RGB signals and Ctrl0 to Ctrl3 are control signals (such as HSYNC and VSYNC). The Ctrl signals can be set by the CTR control register. CH1, CH2, CH3, and CH4 are buffers that hold data temporarily. The default is for the CH1, CH2, CH3, and CH4 data to be output directly without change to OLDIn_CH1_P/N, OLDIn_CH2_P/N, OLDI0_CH3_P/N, and OLDIn_CH4_P/N. The CH assignment can be switched with the CH control register settings.

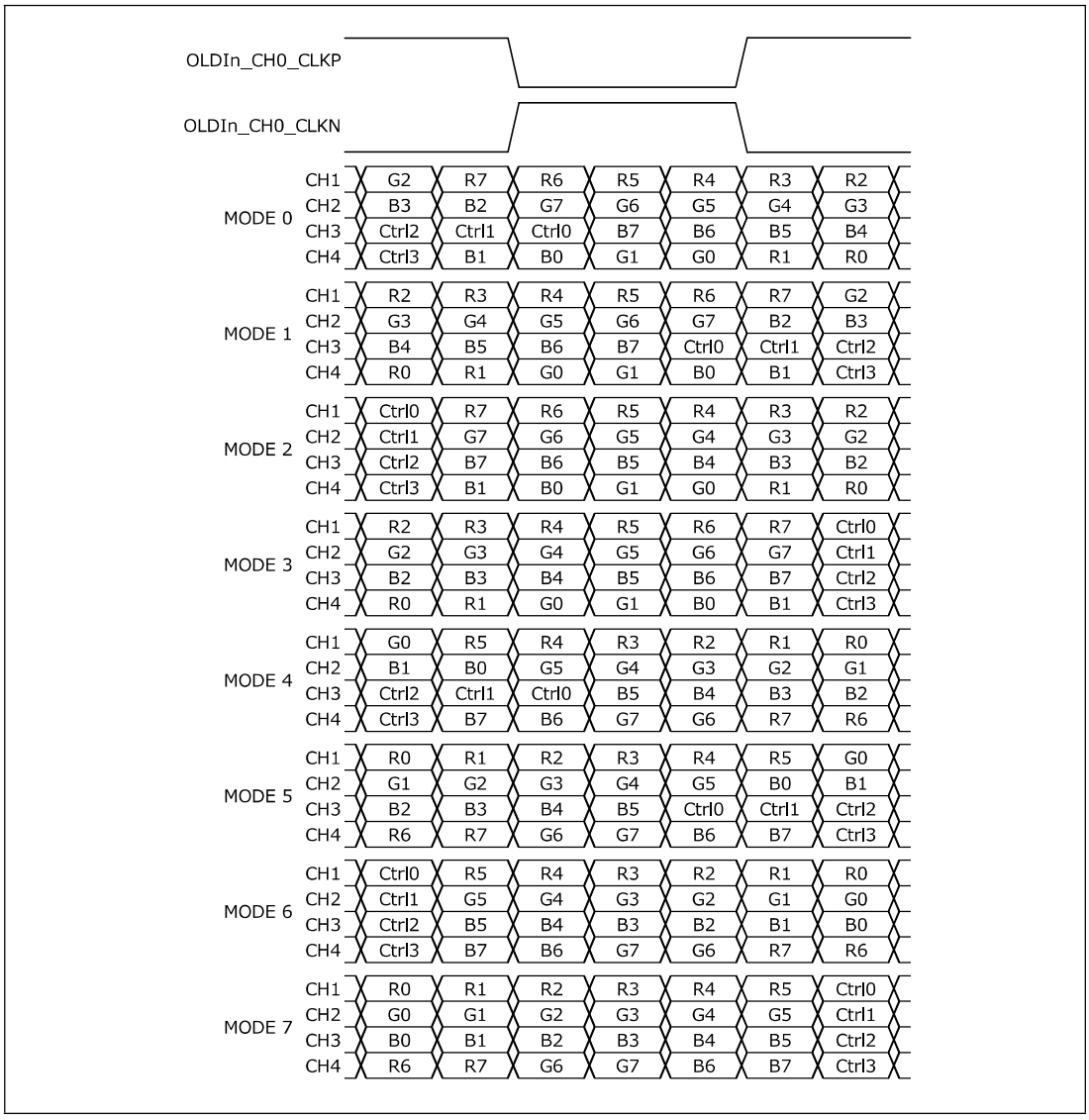


Figure 37.28 Output Data Format

37.12.4.2 Ctrl Signal Selection

The two stage settings shown below are required to select the Ctrl signals.

(1) Open LDI Port Selection

The Open LDI ports used to output the Ctrl signals are selected with the CTR control register (OLDInCTRCR). This results in the signals input to the Open LDI ports being output as Ctrl signals.

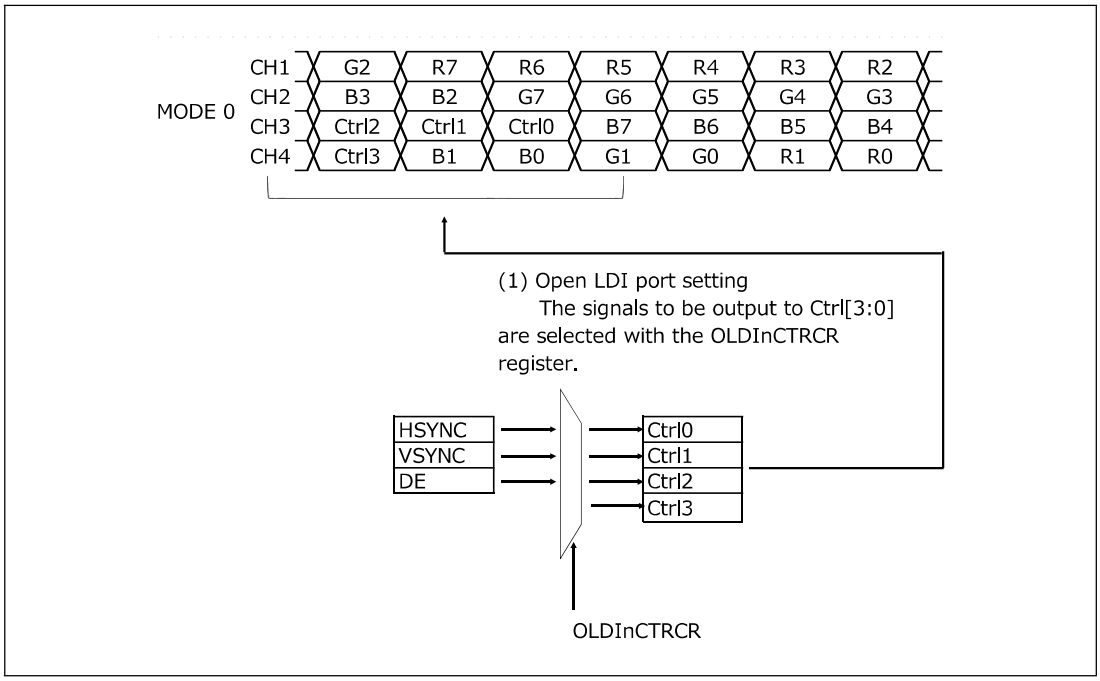


Figure 37.29 Ctrl Signal Selection

Set the desired Ctrl signal characteristics with the VDCE registers.

For example, to reverse the polarity of the HSYNC signal, the polarity must be set with VDCE register settings. (The Open LDI module only converts the signals output from VDCE to the Open LDI format.)

An example of Ctrl signal settings is shown below.

1. Ctrl0 = HSYNC, Ctrl1 = VSYNC, and Ctrl2 = DE

Set the VDCE registers so that HSYNC, VSYNC, and DE are output.

Then set the CTR control register (OLDInCTRCR) so that CTR0SEL[2:0] = 000, CTR1SEL[2:0] = 000, and CTR2SEL[2:0] = 000.

37.12.4.3 CH Selection

The Open LDI module stores the RGB signal data in CH1, CH2, CH3, and CH4 according to the mode selection and Ctrl signal selection registers.

The CH1, CH2, CH3, and CH4 data is then stored in the CH1, CH2, CH3, and CH4 FIFOs according to the CH selection register setting. The data stored in the CH1, CH2, CH3, and CH4 FIFOs is output from the external pins after passing through the LVDS buffers.

37.12.5 Usage Notes

The mode is selected by the LVMD[3:0] bits in the Open LDI control register 0. Figure 37.28, Output Data Format shows the output data format.

Here, R0 to R7, G0 to G7, and B0 to B7 are the RGB signals and Ctrl0 to Ctrl3 are control signals (such as HSYNC and VSYNC). The Ctrl signals can be set by the CTR control register. CH1, CH2, CH3, and CH4 are buffers that hold data temporarily. The default is for the CH1, CH2, CH3, and CH4 data to be output directly without change to OLDIn_CH1_P/N, OLDIn_CH2_P/N, OLDIn_CH3_P/N, and OLDIn_CH4_P/N. The CH assignment can be switched with the CH control register settings.

Sample Configuration flow of Open LDI

(1) Initialization

(1) Initialize Open LDI ports to port mode and input mode:
- PMCn_m = 0
- PMn_m = 1

(2) Set OLDInCR0 = 0:
- If OpenLDI was previously active, disable the output and stop the operation.

(3) Set RSDSCFG:
- VODDR output disable. (RSDSCFG.VODDR_OEN = 0)
- Open LDI output enable. (RSDSCFG.OLDI_OEN = 1)

(4) Set OLDInCR1:
- Select output off mode.
  (CLKSTBY/CH1STBY/CH2STBY/CH3STBY/CH4STBY[1:0] = 01)

(5) Set OLDInCTRCR:
- Configure the placement of synchronisation signals in data stream.

(6) Set OLDInCHCR:
- Select the data channel to be output on each of the data output pin pairs.

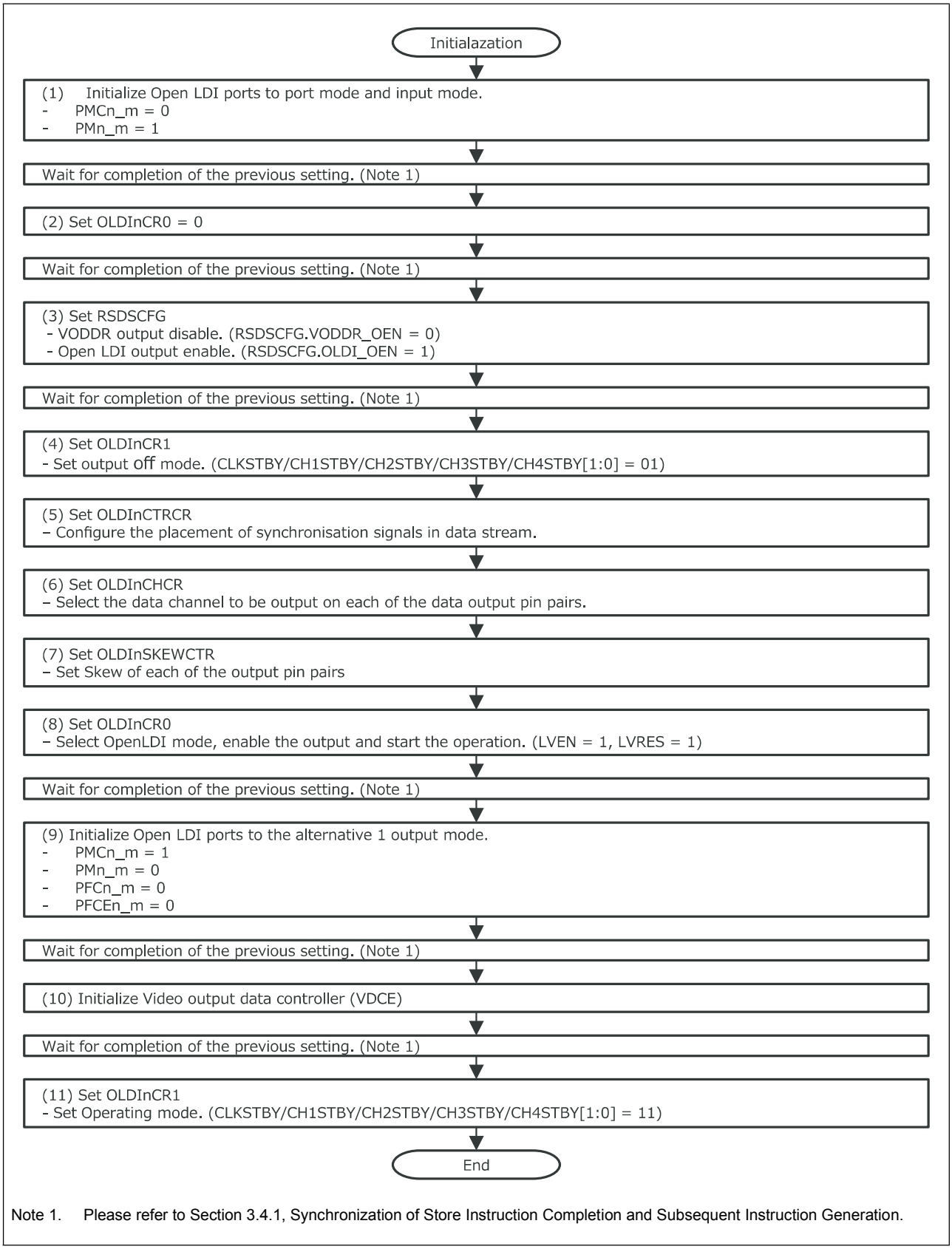
(7) Set OLDInSKEWCTR:
- Set Skew of each of the output pin pairs

(8) Set OLDInCR0:
- Select OpenLDI mode, enable the output and start the operation. (LVEN = 1, LVRES = 1)

(9) Initialize Open LDI ports to the alternative 1 output mode :
- PMCn_m = 1
- PMn_m = 0
- PFCn_m = 0
- PFCEn_m = 0

(10) Initialize Video data controller (VDCE)

(11) Set OLDI0CR1 :
- Select operating mode. (CLKSTBY/CH1STBY/CH2STBY/CH3STBY/CH4STBY[1:0] = 11)



Note 1. Please refer to Section 3.4.1, Synchronization of Store Instruction Completion and Subsequent Instruction Generation.

Figure 37.30 Open LDI initialization flow

(2) Shutdown

(12) Set OLDInCR1 :

– Select output off mode.
(CLKSTBY/CH1STBY/CH2STBY/CH3STBY/CH4STBY[1:0] = 01)

(13) De-initialize Open LDI ports to port mode :

– PMn_m = 1, PMCn_m = 0 or Pn_m = 0, PMn_m = 0, PMCn_m = 0 (Note 1)

NOTE

Ports get set to GPIO input (Hi-Z) with PMn_m = 1, PMCn_m = 0 or to GPIO output Low with Pn_m = 0, PMn_m = 0, PMCn_m = 0.

(3) 3. Recovery

(14) Re-initialize Open LDI ports to the alternative 1 output mode :

– PMCn_m = 1

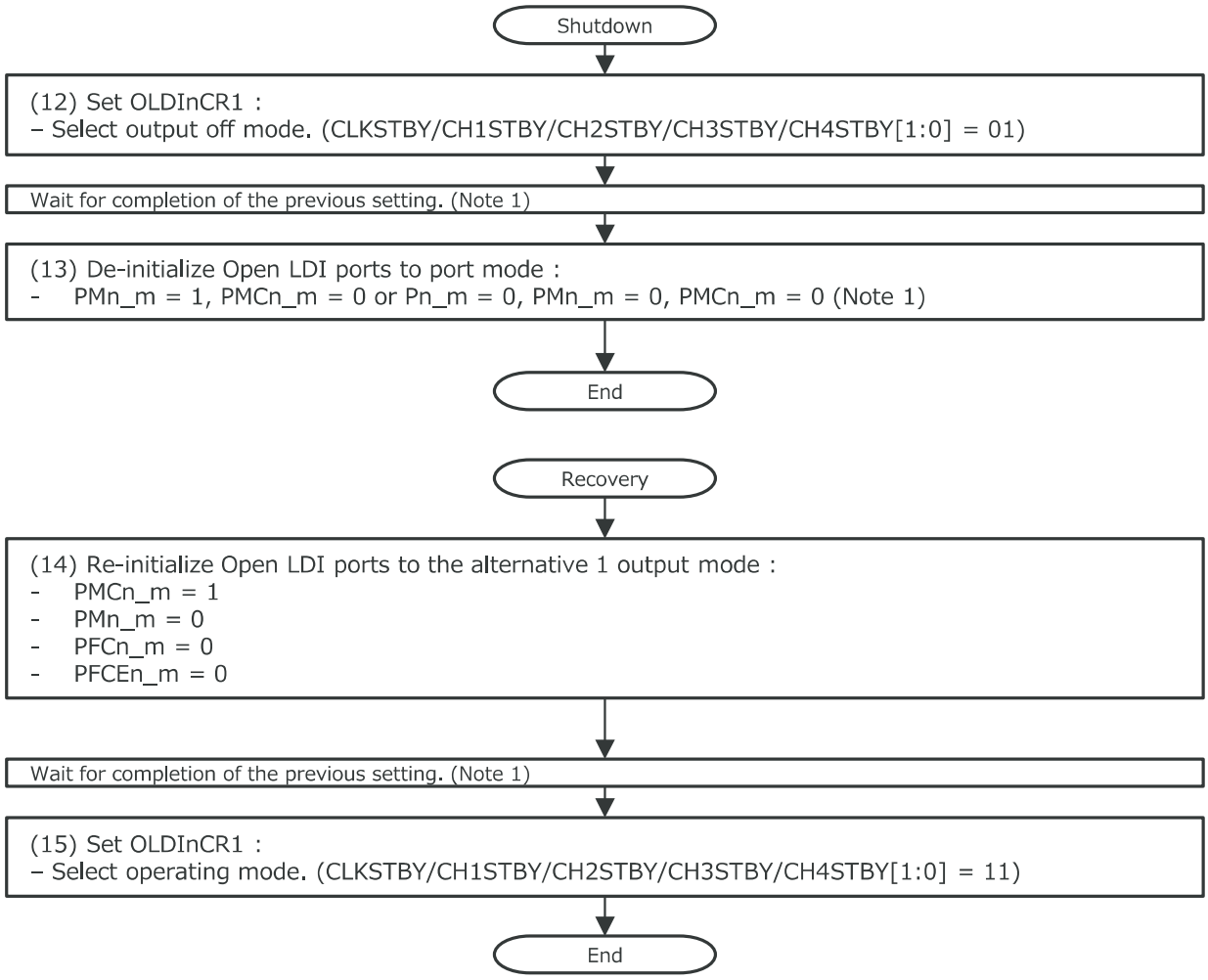
– PMn_m = 0

– PFCn_m = 0

– PFCEn_m = 0

(15) Set OLDInCR1:

– Select operating mode.
(CLKSTBY/CH1STBY/CH2STBY/CH3STBY/CH4STBY[1:0] = 11)



Note 1. Please refer to Section 3.4.1, Synchronization of Store Instruction Completion and Subsequent Instruction Generation.

Figure 37.31 Open LDI shutdown and recovery flow

NOTE

When OpenLDI is used (RSDSCFG.VODDR_OEN is set to 0 and RSDSCFG.OLDI_OEN is set to 1), P45_0 to P45_9 can not be used for other functions than Open LDI.

37.12.6 Limitation

1. Once Open LDI operation is started (OLDInCR0.LVRES = 1 and OLDInCR0.LVEN = 1), do not modify the Open LDI registers/bits as below.
   - OLDInCR0.LVRES = 0 (Stopped)
   - OLDInCR0.LVEN = 0 (Output off)
   - OLDInCR1.CLKSTBY/CH1STBY/CH2STBY/CH3STBY/CH4STBY[1:0] = 00 (Standby mode)
2. Do not disable C_ISO_VDCE0CLK or C_ISO_VDCE1CLK, while Open LDI operation clock (OLDICLK) is running.
   - CKSC_IVDCE0VOS_CTL.VDCE0VOSCSID[2:0] = 000B to disable C_ISO_VDCE0CLK (When using Open LDI Interface with video channel 0)
   - CKSC_IVDCE1VOS_CTL.VDCE1VOSCSID[2:0] = 000B to disable C_ISO_VDCE1CLK (When using Open LDI Interface with video channel 1)

When stopping the operation clock (OLDICLK), it is necessary to stop OLDICLK and IVOEXS0_OUTCLK at the same timing by using CKSC_IDOTCLK0S_CTL (When using Open LDI Interface with video channel 0) or CKSC_IDOTCLK1S_CTL (When using Open LDI Interface with video channel 1).

37.13 Video Output DDR format converter (VODDR) (D1M1A only)

37.13.1 Overview of VODDR

37.13.1.1 Units

This microcontroller has the following number of units of the VODDR.

Table 37.99 Number of Units

Product Name	D1L1	D1L2(H)	D1M1(H) D1M1-V2	D1M1A	D1M2(H)
Units	–	–	–	1	–
Names	–	–	–	VODDR0	–

37.13.1.2 Register addresses

The VODDR register addresses are given as address offsets from the individual base addresses <VODDR_base>.

These base addresses are listed in the following table.

Table 37.100 Register Base Address

Base Address Name	Base Address
<VODDR0_base>	F200 1000_H

37.13.1.3 Clock supply

All VODDR Interfaces provide following clock inputs.

Table 37.101 Clock Supply

Unit Name	Unit Clock Name	Supply Clock Name
VODDR0	APB bus clock PCLK	Clock Controller C_ISO_PCLK
	Operation clock PLLCLK	Clock Controller C_ISO_VODDR_SYSCLK

37.13.1.4 Reset sources

The VODDR and their registers are initialized by the following reset signal:

Table 37.102 Reset sources

Unit Name	Reset Source
VODDR0	• Reset Controller SYSRES

37.13.1.5 External Input/Output Signals

The external input/output signals of VODDR are listed below.

Table 37.103 I/O signals connections

Unit Signal Name	Outline	Alternative Port Pin Signal*1
VODDR0:		
VODDR_LCDOUT[23:0]	Video output data	Video output mode selection VODDR_LCDOUT[23:0]
VODDR_TCON0(VS)	Video output synchronization signal (VSYNC)	Video output mode selection VODDR_TCON0(VS)
VODDR_TCON2(HS)	Video output synchronization signal (HSYNC)	Video output mode selection VODDR_TCON2(HS)
VODDR_TCON3(DE)	Video output synchronization signal (DE)	Video output mode selection VODDR_TCON3(DE)
VODDR_OUT0_CLK	Video output Ch0 pixel clock	Video output mode selection VODDR_OUT0_CLK
VODDR_OUT1_CLK	Video output Ch1 pixel clock	Video output mode selection VODDR_OUT1_CLK

Note 1. The video output signals can be selected from several kinds of format. Refer to Section 37.9.3, D1M1A Video output configuration.

37.13.2 Function Overview

The VODDR module converts two video output channels (VDCE0, VDCE1) to a signal set of one channel video output (LCDOUT[23:0], VSYNC, HSYNC, DE). Two video output pixel clocks are generated by dividing the PLLCLK.

This module has the following features.

- Video output format conversion
- Pixel clock generation
  - Maximum PLLCLK: max. 480MHz
  - Maximum pixel clock: max. 30 MHz. (VO0_PCLK, VO1_PCLK)
- Timing adjustment function of each channel
  - Data output timing adjustment
  - Clock phase adjustment

37.13.3 Function Description

The following figure provides a functional block diagram of the VODDR module.

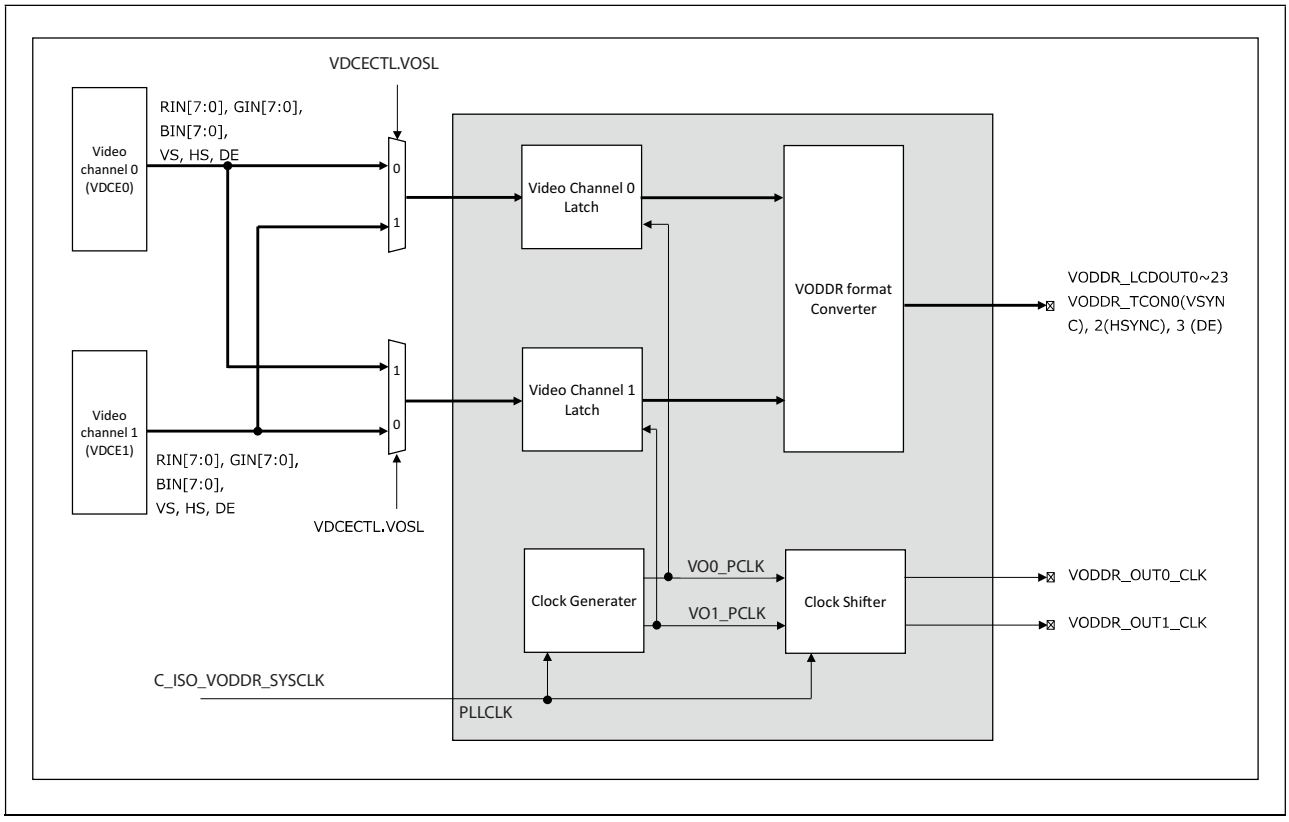


Figure 37.32 VODDR block diagram

37.13.3.1 Video output format conversion

The following figure shows examples of video output format conversion in VODDR module.

VODDR module mix two video output data streams to single video output signal set (LCDOUT23_0, TCON0:VS, TCON2:HS, TCON3:DE).

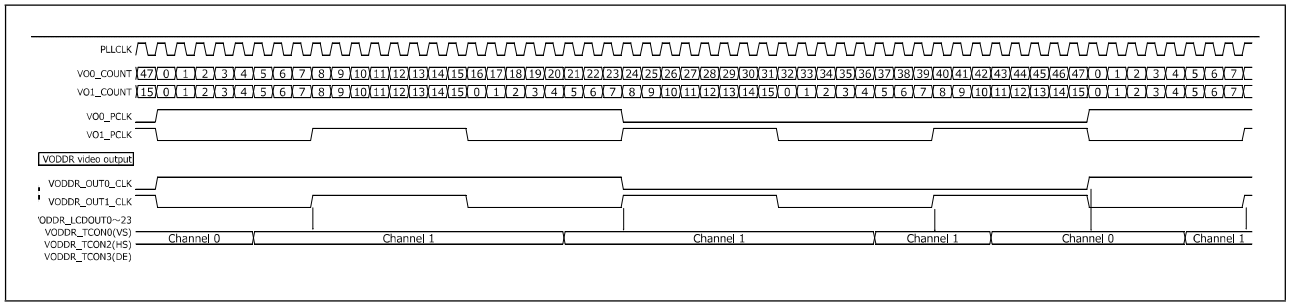


Figure 37.33 VODDR format conversion

37.13.3.2 Pixel clock generation

The following figure shows examples of pixel clock generation in VODDR module.

The frequency of the generated channel 0 pixel clock (VO0_PCLK) is defined by CLK_DIV0[7:0] and the frequency of PLLCLK as $f_{VO0\_PCLK} = f_{PLLCLK} / CLK\_DIV0[7:0]$. Also, the frequency of the generated channel 1 pixel clock (VO1_PCLK) is defined by CLK_DIV1[7:0] and the frequency of PLLCLK as $f_{VO1\_PCLK} = f_{PLLCLK} / CLK\_DIV1[7:0]$.

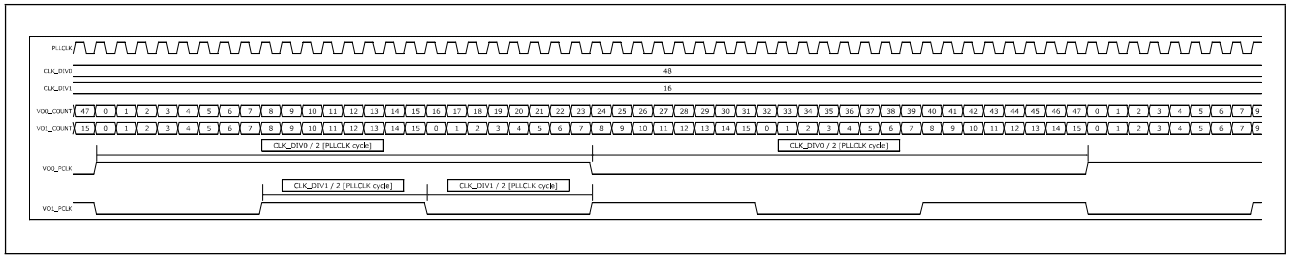


Figure 37.34 VODDR pixel clock generation

CLK_DIV0[7:0] and CLK_DIV1[7:0] must satisfy the followings.

1. The frequency of VO1_PCLK is larger than or equal to the frequency of VO0_PCLK
   - $f_{VO1\_PCLK} \geq f_{VO0\_PCLK}$
2. The frequency of VO0_PCLK and VO1_PCLK is less than or equal to maximum pixel clock frequency.
   - $CLK\_DIV0[7:0] \geq f_{PLLCLK} / f_{max.\ pixel\ clock}$: 30 MHz $\geq f_{VO0\_PCLK}$
   - $CLK\_DIV1[7:0] \geq f_{PLLCLK} / f_{max.\ pixel\ clock}$: 30 MHz $\geq f_{VO1\_PCLK}$
3. The values of CLK_DIV0[7:0] and CLK_DIV1[7:0] is even value.
   - CLK_DIV0[7:0] % 2 = 0
   - CLK_DIV1[7:0] % 2 = 0
4. VO1_PCLK frequency is an integer multiple of the VO0_PCLK frequency.
   - $f_{VO0\_PCLK} : f_{VO1\_PCLK} = 1 : 1, 1 : 2, 1 : 3, \ldots$

37.13.3.3 Timing adjustment function

(1) Data output timing adjustment

The following figure shows examples of video data output timing.

The video data output timing is defined by OUT_TIM0[2:0] and OUT_TIM1[2:0].

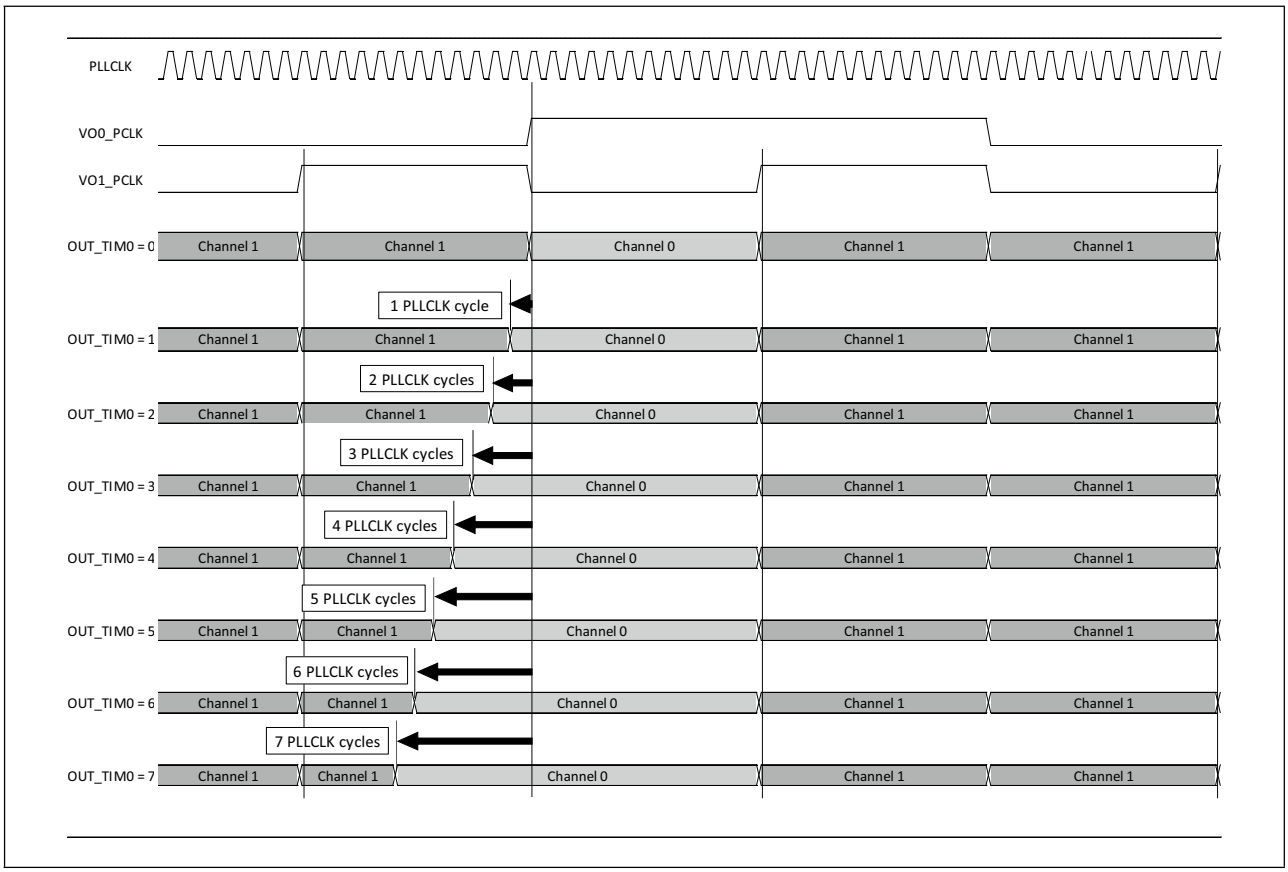


Figure 37.35 VODDR data output timing adjustment

(2) Clock phase adjustment

The following figure shows examples of pixel clock phase adjustment in VODDR module.

The VODDR_OUT0_CLK is shifted to CONFIG0[2:0] cycles behind. Also, the VODDR_OUT1_CLK is shifted to CONFIG1[2:0] cycles behind. The data output timing adjustment is performed the rise edge of VO0_PCLK or VO1_PCLK as a base point. Thus, the clock phase adjustment does not affect video data output timing.

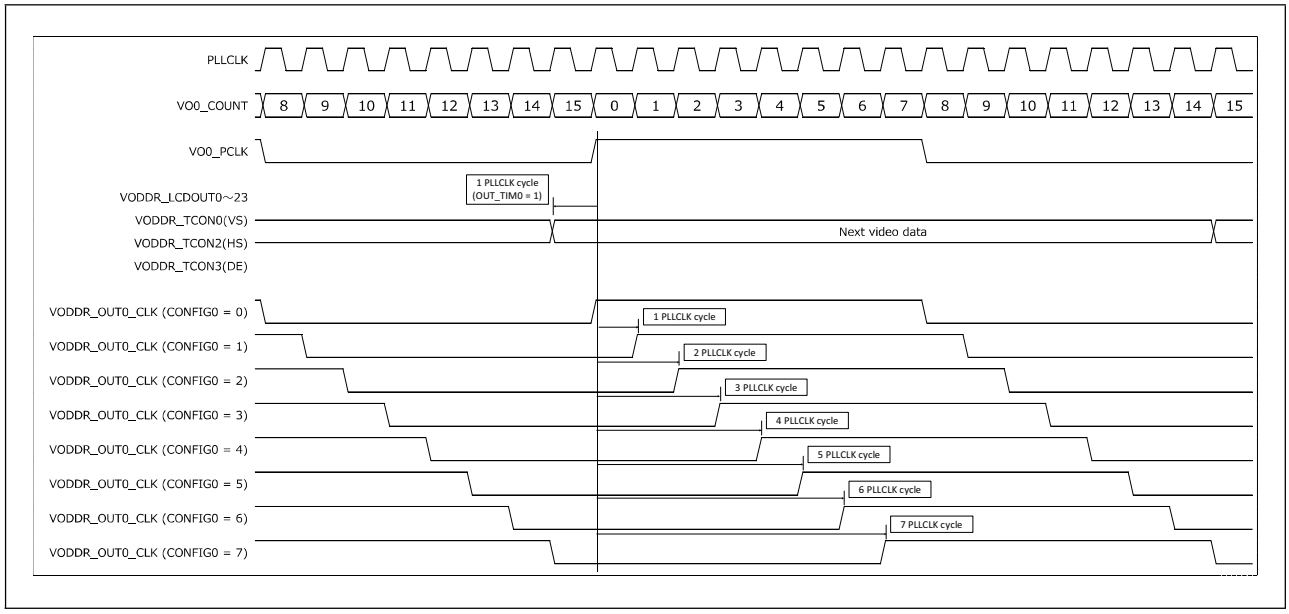


Figure 37.36 VODDR clock phase adjustment

37.13.4 Register Description

The table below lists the register configuration.

Register Access Size

All registers can be accessed in 32-bit units.

Correct device operation is not guaranteed if any access size other than 32-bit is used to access these registers.

<VODDR_base>

The base address <VODDR_base> of the VODDR is defined in Section 37.13.1.2, Register addresses.

Table 37.104 Register Configuration

Module Name	Register name	Shortcut	Address
VODDR0	System control register	VODDR0SYSCNT	<VODDR_base> + 00_H
	Clock divider control register	VODDR0CLKDIV	<VODDR_base> + 04_H
	Timing control 1 register	VODDR0TIMCNT1	<VODDR_base> + $0C_H$

37.13.4.1 VODDR0SYSCNT – VODDR System control register

This register controls VODDR function.

Access: This register can be read/written in 32-bit units.

Address: <VODDR_base> + 00_H

Initial value: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VO1_ENABLE
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK_START
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Table 37.105 VODDR0SYSCNT register contents

Bit Position	Bit Name	Function
31 to 17	-	These bits are always read as 0. When written, write the initial value.
16	VO1_ENABLE	Video channel 1 enable When this bit is 1 (enable), video channel 1 is output. 0: Disable Video channel 1. (Only video channel 0 is output) 1: Enable Video channel 1. (Both of video channel 0 and video channel 1 is output)
15 to 1	-	These bits are always read as 0. When written, write the initial value.
0	CLK_START	Clock generation enable 0: Disable clock generation 1: Enable clock generation Before setting this bit to 1, it must be set to following registers. VODDR0CLKDIV, VODDR0TIMCNT1

37.13.4.2 VODDR0CLKDIV – VODDR Clock divider control register

This register controls VODDR function.

Access: This register can be read/written in 32-bit units.

Address: <VODDR_base> + 04_H

Initial value: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	CLK_DIV1[7:0]							
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	CLK_DIV0[7:0]							
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 37.106 VODDR0CLKDIV register contents

Bit Position	Bit Name	Function
31 to 24	-	These bits are always read as 0. When written, write the initial value.
23 to 16	CLK_DIV1[7:0]	Clock divider setting for video channel 1 This register must be set even value. 0: Setting prohibited. 1: Setting prohibited. 2: $f_{(VO1\_PCLK)} = f_{(PLLCLK)} / 2$ 3: Setting prohibited. 4: $f_{(VO1\_PCLK)} = f_{(PLLCLK)} / 4$ ... 254: $f_{(VO1\_PCLK)} = f_{(PLLCLK)} / 254$ 255: Setting prohibited.
15 to 8	-	These bits are always read as 0. When written, write the initial value.
7 to 0	CLK_DIV0[7:0]	Clock divider setting for video channel 0 This register must be set even value. 0: Setting prohibited. 1: Setting prohibited. 2: $f_{(VO0\_PCLK)} = f_{(PLLCLK)} / 2$ 3: Setting prohibited. 4: $f_{(VO0\_PCLK)} = f_{(PLLCLK)} / 4$ ... 254: $f_{(VO0\_PCLK)} = f_{(PLLCLK)} / 254$ 255: Setting prohibited.

NOTES

1. CLK_DIV0[7:0] and CLK_DIV1[7:0] must satisfy the following condition:
   $f_{(VO0\_PCLK)} \leq f_{(VO1\_PCLK)}$
2. While VODDR0SYSCNT.CLK_START = 1, the setting of this register is prohibited.

37.13.4.3 VODDR0TIMCNT1 – VODDR timing control 1 register

This register controls VODDR function.

Access: This register can be read/written in 32-bit units.

Address: <VODDR_base> + $0C_H$

Initial value: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	OUT_TIM1[2:0]			—	—	—	—	—	OUT_TIM0[2:0]		
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W	R	R	R	R	R	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	CONFIG1[2:0]			—	—	—	—	—	CONFIG0[2:0]		
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W	R	R	R	R	R	R/W	R/W	R/W

Table 37.107 VODDR0TIMCNT1 register contents

Bit Position	Bit Name	Function
31 to 27	-	These bits are always read as 0. When written, write the initial value.
26 to 24	OUT_TIM1[2:0]	Setting data output timing adjustment for video channel 1 Video data of channel 1 are output OUT_TIM1[2:0] cycles prior to rising edge of VO1_PCLK. 0: 0 PLLCLK cycle 1: 1 PLLCLK cycles … 7: 7 PLLCLK cycles
23 to 19	-	These bits are always read as 0. When written, write the initial value.
18 to 16	OUT_TIM0[2:0]	Setting data output timing adjustment for video channel 0 Video data of channel 0 are output OUT_TIM0[2:0] cycles prior to rising edge of VO0_PCLK. 0: 0 PLLCLK cycle 1: 1 PLLCLK cycles … 7: 7 PLLCLK cycles
15 to 11	-	These bits are always read as 0. When written, write the initial value.
10 to 8	CONFIG1[2:0]	Setting clock phase adjustment for video channel 1 The VODDR_OUT1_CLK is output shifted to CONFIG1[2:0] cycles behind. 0: 0 PLLCLK cycle 1: 1 PLLCLK cycle 2: 2 PLLCLK cycles … 7: 7 PLLCLK cycles
7 to 3	-	These bits are always read as 0. When written, write the initial value.
2 to 0	CONFIG0[2:0]	Setting clock phase adjustment for video channel 0 The VODDR_OUT0_CLK is output shifted to CONFIG0[2:0] cycles behind. 0: 0 PLLCLK cycle 1: 1 PLLCLK cycle 2: 2 PLLCLK cycles … 7: 7 PLLCLK cycles

NOTE

While VODDR0SYSCNT.CLK_START = 1, the setting of this register is prohibited.

37.13.5 Operation

(1) Start sequence for VODDR.

The following sequence describes the start sequence for VODDR.

1. Select VODDR format by RSDSCFG.VODDR_OEN = 1B
2. Set C_ISO_VODDR_SYSCLK by configuration of CKSC_IDOTCLK0S_CTL
3. Set up following registers:
   - VODDR0CLKDIV.CLK_DIV0[7:0]
   - VODDR0CLKDIV.CLK_DIV1[7:0]
   - VODDR0TIMCNT1.CONFIG0[2:0]
   - VODDR0TIMCNT1.CONFIG1[2:0]
   - VODDR0TIMCNT1.OUT_TIM0[2:0]
   - VODDR0TIMCNT1.OUT_TIM1[2:0]
4. Enable clock generation by VODDR0SYSCNT = 0001 0001H
5. Setting Video Data Controller E (VDCE)

(2) Re-start sequence for VODDR

The following sequence describes the re-start sequence for VODDR.

1. Disable clock generation by VODDR0SYSCNT = 0000 0000H
2. Set up following registers:
   - VODDR0CLKDIV.CLK_DIV0[7:0]
   - VODDR0CLKDIV.CLK_DIV1[7:0]
   - VODDR0TIMCNT1.CONFIG0[2:0]
   - VODDR0TIMCNT1.CONFIG1[2:0]
   - VODDR0TIMCNT1.OUT_TIM0[2:0]
   - VODDR0TIMCNT1.OUT_TIM1[2:0]
3. Enable clock generation by VODDR0SYSCNT = 0001 0001_H

37.13.5.1 Setting Example for WVGA + WQVGA

Table 37.108 and **Table 37.109** are setting example for using WVGA (800 x 480 @ 60Hz) + WQVGA (480 x 272 @ 60Hz) displays.

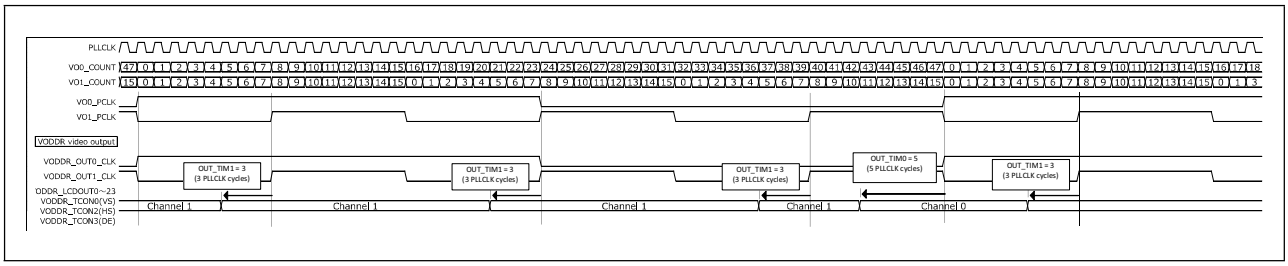


Figure 37.37 VODDR WVGA + WQVGA (60Hz) Video Output

Table 37.108 Video parameters for WVGA + WQVGA (60 Hz) display

Item		Settings
PLLCLK	PLLCLK frequency	480 MHz
Channel 0	Video image size	WQVGA
	Frame rate	60 Hz
	Vertical valid period	272 lines
	Horizontal valid period	480 pixels
	Pixel clock frequency	10 MHz
Channel 1	Video image size	WVGA
	Frame rate	60 Hz
	Vertical valid period	480 lines
	Horizontal valid period	800 pixels
	Pixel clock frequency	30 MHz

Table 37.109 VODDR Register Setting Example for WVGA + WQVGA (60 Hz) display

Register Name	Bit Name	Settings	Remarks
VODDR0CLKDIV	CLK_DIV0[7:0]	48	$f_{(VO0\_PCLK)} = f_{(PLLCLK)} / 48 = 10$ MHz
	CLK_DIV1[7:0]	16	$f_{(VO1\_PCLK)} = f_{(PLLCLK)} / 16 = 30$ MHz
VODDR0TIMCNT1	CONFIG0[2:0]	0	VODDR_OUT0_CLK is shifted 0 cycles for VO0_PCLK
	COMFIG1[2:0]	0	VODDR_OUT1_CLK is shifted 0 cycles for VO1_PCLK
	OUT_TIM0[2:0]	5	Video data of channel 0 are output before 5 cycles prior to rising edge of VO0_PCLK
	OUT_TIM1[2:0]	3	Video data of channel 1 are output before 3 cycles prior to rising edge of VO1_PCLK

Table 37.110 and **Table 37.111** are setting example for using WVGA (800 x 480 @ 60Hz) + WQVGA (480 x 272 @ 50Hz) displays.

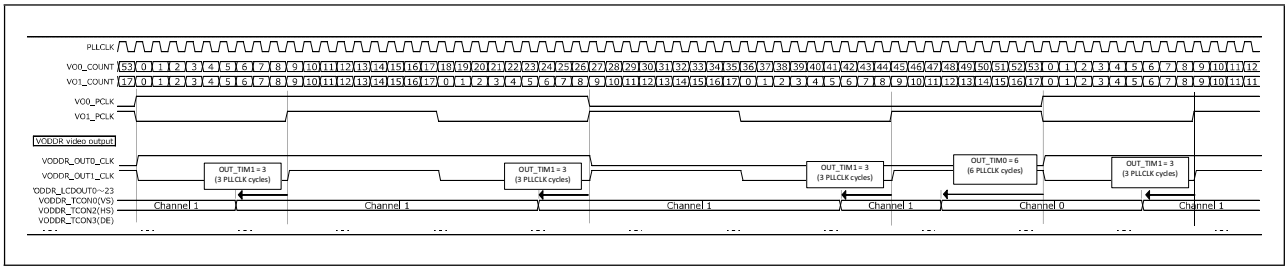


Figure 37.38 VODDR WVGA + WQVGA (50 Hz) Video Output

Table 37.110 Video parameters for WVGA + WQVGA (50 Hz) display

Item		Settings
PLLCLK	PLLCLK frequency	480MHz
Channel 0	Video image size	WQVGA
	Frame rate	50 Hz
	Vertical valid period	272 lines
	Horizontal valid period	480 pixels
	Pixel clock frequency	8.9 MHz
Channel 1	Video image size	WVGA
	Frame rate	60 Hz
	Vertical valid period	480 lines
	Horizontal valid period	800 pixels
	Pixel clock frequency	26.7 MHz

Table 37.111 VODDR Register Setting Example for WVGA + WQVGA (50Hz) display

Register Name	Bit Name	Settings	Remarks
VODDR0CLKDIV	CLK_DIV0[7:0]	54	$f_{(VO0\_PCLK)} = f_{(PLLCLK)} / 54 = 8.9$ MHz
	CLK_DIV1[7:0]	18	$f_{(VO1\_PCLK)} = f_{(PLLCLK)} / 18 = 26.7$ MHz
VODDR0TIMCNT1	CONFIG0[2:0]	0	VODDR_OUT0_CLK is shifted 0 cycles for VO0_PCLK
	COMFIG1[2:0]	0	VODDR_OUT1_CLK is shifted 0 cycles for VO1_PCLK
	OUT_TIM0[2:0]	6	Video data of channel 0 are output before 6 cycles prior to rising edge of VO0_PCLK
	OUT_TIM1[2:0]	3	Video data of channel 1 are output before 3 cycles prior to rising edge of VO1_PCLK

37.13.5.2 Setting Example for WQVGA + WQVGA

Table 37.112 and **Table 37.113** are setting example for using WQVGA (480 x 272 @ 60Hz) + WQVGA (480 x 272 @ 60Hz) displays.

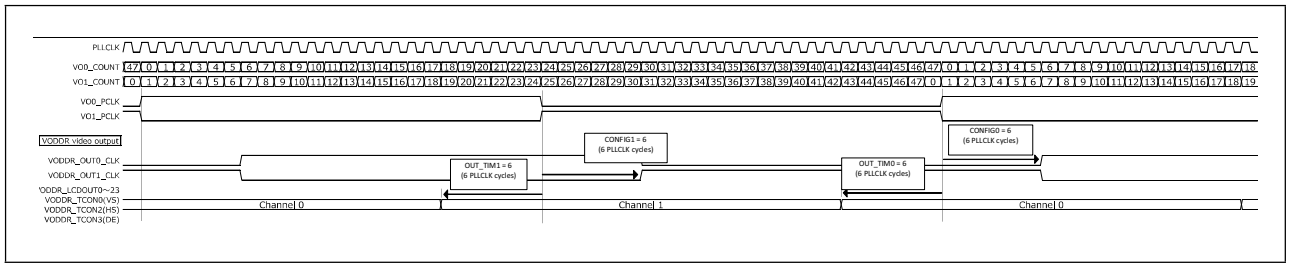


Figure 37.39 VODDR WQVGA + WQVGA Video Output

Table 37.112 Video parameters for WQVGA + WQVGA display

Item		Settings
PLLCLK	PLLCLK frequency	480 MHz
Channel 0	Video image size	WQVGA
	Frame rate	60 Hz
	Vertical valid period	272 lines
	Horizontal valid period	480 pixels
	Pixel clock frequency	10 MHz
Channel 1	Video image size	WQVGA
	Frame rate	60 Hz
	Vertical valid period	272 lines
	Horizontal valid period	480 pixels
	Pixel clock frequency	10 MHz

Table 37.113 VODDR Register Setting Example for WQVGA + WQVGA display

Register Name	Bit Name	Settings	Remarks
VODDR0CLKDIV	CLK_DIV0[7:0]	48	$f_{(VO0\_PCLK)} = f_{(PLLCLK)} / 48 = 10$ MHz
	CLK_DIV1[7:0]	48	$f_{(VO1\_PCLK)} = f_{(PLLCLK)} / 48 = 10$ MHz
VODDR0TIMCNT1	CONFIG0[2:0]	6	VODDR_OUT0_CLK is shifted 6 cycles for VO0_PCLK
	CONFIG1[2:0]	6	VODDR_OUT1_CLK is shifted 6 cycles for VO1_PCLK
	OUT_TIM0[2:0]	6	Video data of channel 0 are output before 6 cycles prior to rising edge of VO0_PCLK
	OUT_TIM1[2:0]	6	Video data of channel 1 are output before 6 cycles prior to rising edge of VO1_PCLK