Rockchip
PX3 SE
Technical Reference Manual
Part2

Revision 1.0
Aug. 2017

Revision History

Date	Revision	Description
2017-8-8	1.0	Initial Release

Table of Content

Figure Index

Table Index

Warranty Disclaimer

Rockchip Electronics Co.,Ltd makes no warranty, representation or guarantee (expressed, implied, statutory, or otherwise) by or with respect to anything in this document, and shall not be liable for any implied warranties of non-infringement, merchantability or fitness for a particular purpose or for any indirect, special or consequential damages.

Information furnished is believed to be accurate and reliable. However, Rockchip Electronics Co.,Ltd assumes no responsibility for the consequences of use of such information or for any infringement of patents or other rights of third parties that may result from its use.

Rockchip Electronics Co.,Ltd's products are not designed, intended, or authorized for using as components in systems intended for surgical implant into the body, or other applications intended to support or sustain life, or for any other application in which the failure of the Rockchip Electronics Co.,Ltd's product could create a situation where personal injury or death may occur, should buyer purchase or use Rockchip Electronics Co.,Ltd's products for any such unintended or unauthorized application, buyers shall indemnify and hold Rockchip Electronics Co.,Ltd and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs, damages, expenses, and reasonable attorney fees arising out of, either directly or indirectly, any claim of personal injury or death that may be associated with such unintended or unauthorized use, even if such claim alleges that Rockchip Electronics Co.,Ltd was negligent regarding the design or manufacture of the part.

Copyright and Patent Right

Information in this document is provided solely to enable system and software implementers to use Rockchip Electronics Co.,Ltd's products. There are no expressed or implied copyright licenses granted hereunder to design or fabricate any integrated circuits or integrated circuits based on the information in this document.

Rockchip Electronics Co.,Ltd does not convey any license under its patent rights nor the rights of others.
All copyright and patent rights referenced in this document belong to their respective owners and shall be subject to corresponding copyright and patent licensing requirements.

Trademarks

Rockchip and Rockchip™ logo and the name of Rockchip Electronics Co.,Ltd's products are trademarks of Rockchip Electronics Co.,Ltd. and are exclusively owned by Rockchip Electronics Co.,Ltd. References to other companies and their products use trademarks owned by the respective companies and are for reference purpose only.

Confidentiality

The information contained herein (including any attachments) is confidential. The recipient hereby acknowledges the confidentiality of this document, and except for the specific purpose, this document shall not be disclosed to any third party.

Reverse engineering or disassembly is prohibited.

ROCKCHIP ELECTRONICS CO.,LTD. RESERVES THE RIGHT TO MAKE CHANGES IN ITS PRODUCTS OR PRODUCT SPECIFICATIONS WITH THE INTENT TO IMPROVE FUNCTION OR DESIGN AT ANY TIME AND WITHOUT NOTICE AND IS NOT REQUIRED TO UNDATE THIS DOCUMENTATION TO REFLECT SUCH CHANGES.

Copyright © 2017 Rockchip Electronics Co., Ltd.

Chapter 1 Video Output Processor (VOP)

1.1 Overview

Video Output Processor is a video process engine and a display interface from memory frame buffer to display device(HDMI,LCDpanel,MIPI,LVDS, or CVBS). VOP is connected to an AHB bus through an AHB slave and AXI bus through an AXI master. The register setting is configured through the AHB slave interface and the display frame data is read through the AXI master interface.

1.1.1 Features

- Display interface
  - Parallel RGB LCD Interface: 24-bit(RGB888), 18-bit(RGB666), 15-bit(RGB565)
  - TV Interface
    - RGB2YCbCr, 8bit
    - TV encoder 10bit out for DAC
    - RGB888+1080i for HDMI
    - Parallel RGB HDMI Interface: 24-bit(RGB888 YCbCr444)
  - Max output resolution
    - 1920x1080 for HDMI
    - 480i/576i for CVBS
    - Support displaying the same source on HDMI and PAD simultaneously(not support HDMI+vertical PAD)
- Display process
  - Background layer
    - programmable 24-bit color
  - Win0 layer
    - RGB888, ARGB888, RGB565, YCbCr422, YCbCr420, YCbCr444
    - 1/8 to 8 scaling-down and scaling-up engine
    - Support virtual display
    - 256 level alpha blending (pre-multiplied alpha support)
    - Transparency color key
    - De-flicker support for interlace output
    - Direct path support
    - YCbCr2RGB(rec601-mpeg/ rec601-jpeg/rec709)
    - RGB2YCbCr(BT601/BT709)
  - Win1 layer
    - RGB888, ARGB888, RGB565
    - Support virtual display
    - 256 level alpha blending (pre-multiplied alpha support)
    - Transparency color key
    - Direct path support
    - RGB2YCbCr(BT601/BT709)
  - Hardware cursor:
    - 8BPP(ARGB888 LUT)
    - Support two size: 32x32 and 64x64
    - 256 level alpha blending
    - Support hwc over panel at right and below side
  - Scaler:
    - Display interface
      - output for LVDS/RGB
        max output resolution: 1280x800
        support display input directly bypass
      - Asynchronous output pixel clock (PLL required)
      - output dclk from PLL integer or fraction output
    - Image scale
      - Max input resolution: 1920x1080p
      - Min input resolution: 720x480p
      - Max output resolution: 1280x800

- Min output resolution: 800x600
- support 1:1 bypass using line buffer
- scale down
  Arbitrary non-integer scaling ratio
  Max 1/4 scaling ratio
- scale up
  Arbitrary non-integer scaling ratio
  Max 4 scaling ratio

- Others
  - Win0 layer and Win1 layer overlay exchangeable
  - BCSH(Brightness,Contrast,Saturation,Hue adjustment)
  - BCSH:YCbCr2RGB(rec601-mpeg/ rec601-jpeg/rec709)
  - BCSH:RGB2YCbCr(BT601/BT709)
  - Support Gamma adjust for PAD
  - Support dither down allegro RGB888to666 RGB888to565 & dither down frc (configurable) RGB888to666
  - Blank and black display
  - Standby mode
  - Support MMU
  - Support QoS request for higher bus priority for win1/HWC
  - Support NOC hurry for higher bus prioirity for win0
  - Support DMA stop mode

1.2 Block Diagram

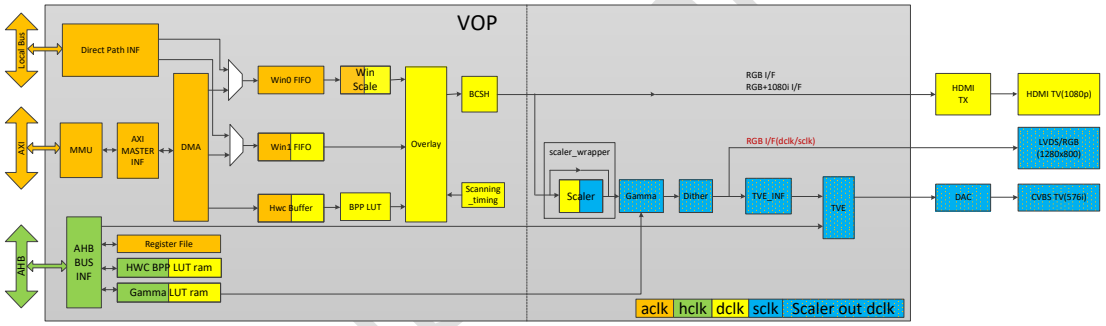


Fig. 18-1 VOP Block Diagram

1.3 Function description

1.3.1 Data Format

VOP master read the frame data from the frame buffer in the system memory. There are total 7 formats supported in three layers.

- Win0: RGB888, ARGB888, RGB565, YCbCr422_SP, YCbCr420_SP, YCbCr444_SP
- Win1: RGB888, ARGB888, RGB565
- Hwc: 8bpp

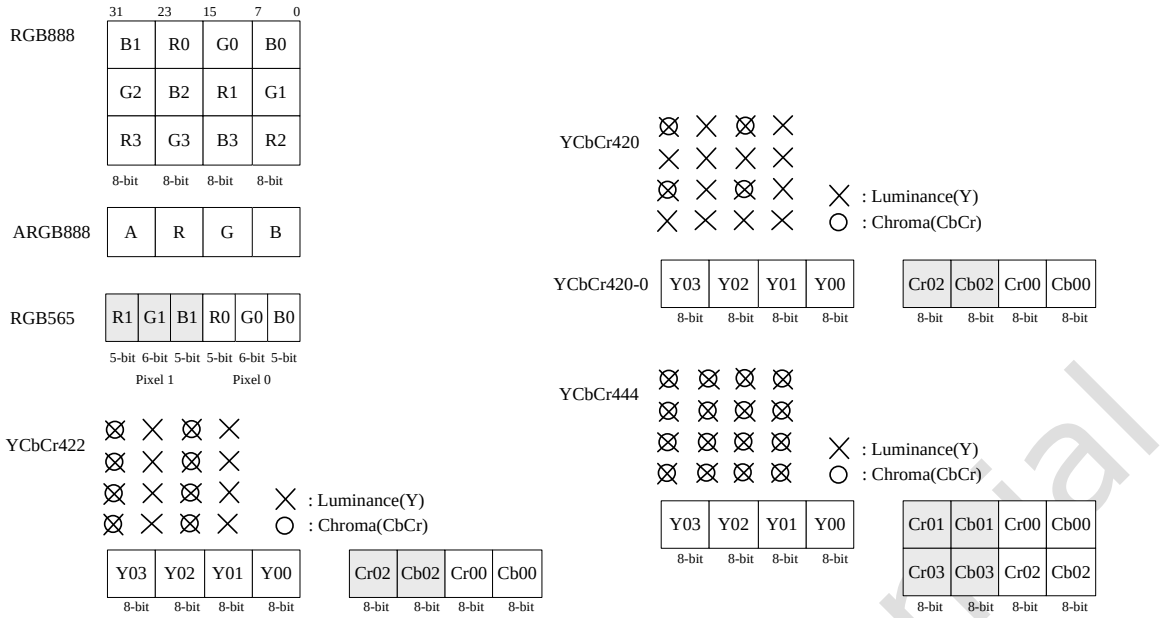
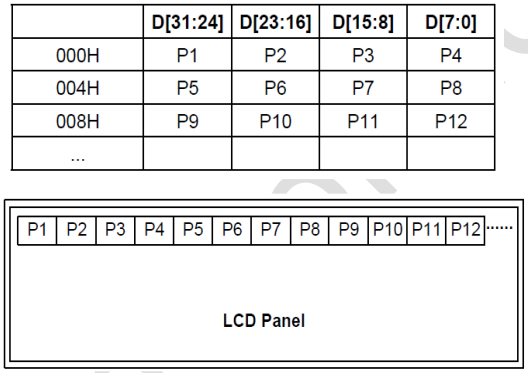


Fig. 18-2 VOP Frame Buffer Data Format



	D[31:24]	D[23:16]	D[15:8]	D[7:0]
000H	P1	P2	P3	P4
004H	P5	P6	P7	P8
008H	P9	P10	P11	P12
...				

Fig. 18-3 VOP Win1 Palette (8bpp)

Data SWAP function

There are several swap options for different frame data formats. The register is VOP_SYS_CTRL[21:15].

All the data swap types are in the following table.

Table 18-1 VOP Data Swap of Win0 and Win1

Data-swap	RB swap	Alpha swap	Y-M8 swap	CbCr swap
Win0	yes	yes	yes	yes
Win1	yes	yes	No	No

1.3.2 Data path

There are two data input path for VOP to get display layers' pixel data. One is internal DMA; the other is direction path interface.

1.Internal DMA

Internal DMA can fetch the pixel data through AXI bus from system memory (DDR) for all the display layers. Data fetching is driven by display output requirement.

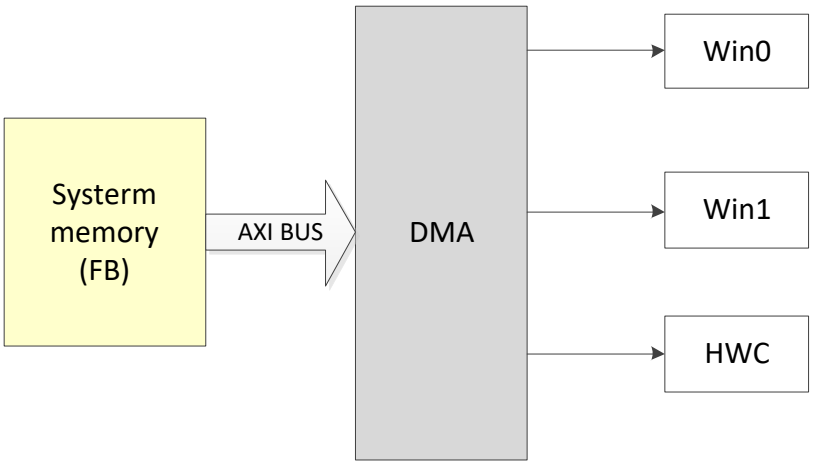


Fig. 18-4 VOP Internal DMA

2. Direct Path Interface

Direct path interface (DPI) is used for direct image display of external image processing IP. There is a local bus between VOP and external image processing IP for the data transfer. DPI is connected to WIN0/WIN1 but can only be configured for One layer use (Win0 or Win1) in each frame.

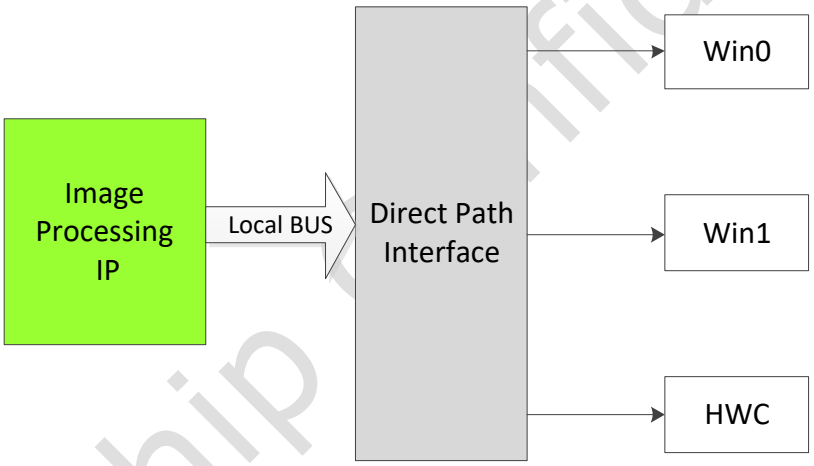


Fig. 18-5 VOP Direct Path Interface

1.3.3 Virtual display

Virtual display is supported in Win0 and Win1. The active image is part of the virtual (original) image in frame buffer memory. The virtual width is indicated by setting WIN0/WIN1_VIR_STRIDE for different data format.

The virtual stride should bemultiples of word (32-bit). That means dummy bytes in the end of virtual line if the real pixels are not 32-bit aligned.

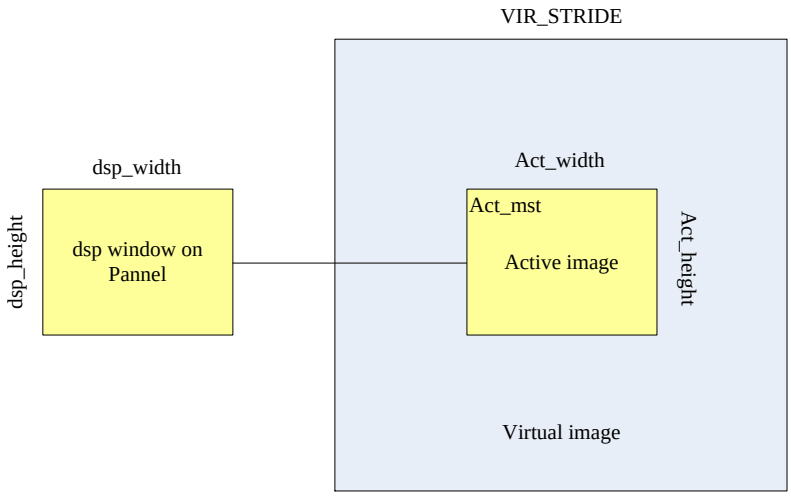


Fig. 18-6 VOP Virtual Display Mode

Scaling

The scaling operation is the imageresizing processof data transfer from the frame buffer memory toLCD panel or TV set.

Horizontal and vertical scaling factor should be set according the window scaling ratio.

1. Scaling factor

Because the Chroma data may have different sampling rate with Luma data in the memory format of YCbCr422/YCbCr420. The scaling factor of Win0 has two couples of factor registers:

VOP_WIN0_SCL_FACTOR_Y/VOP_WIN0_SCL_FACTOR_CBR

Software calculates the scaling factor value using thefollowing equations:

$$y\_rgb\_vertical\_factor = (\frac{VOP\_WIN0\_ACT\_INFO[31:16]}{VOP\_WIN0\_DSP\_INFO[31:16]}) \times 2^{12}$$

$$y\_rgb\_horizontal\_factor = (\frac{VOP\_WIN0\_ACT\_INFO[15:0]}{VOP\_WIN0\_DSP\_INFO[15:0]}) \times 2^{12}$$

$$yuv422\_yuv444\_Cbr\_vertical\_factor = (\frac{VOP\_WIN0\_ACT\_INFO[31:16]}{VOP\_WIN0\_DSP\_INFO[31:16]}) \times 2^{12}$$

$$yuv420\_Cbr\_vertical\_factor = (\frac{VOP\_WIN0\_ACT\_INFO[31:16]/2}{VOP\_WIN0\_DSP\_INFO[31:16]}) \times 2^{12}$$

$$yuv444\_Cbr\_horizontal\_factor = (\frac{VOP\_WIN0\_ACT\_INFO[15:0]}{VOP\_WIN0\_DSP\_INFO[15:0]}) \times 2^{12}$$

$$yuv422\_yuv420\_Cbr\_horizontal\_factor = (\frac{VOP\_WIN0\_ACT\_INFO[15:0]/2}{VOP\_WIN0\_DSP\_INFO[15:0]}) \times 2^{12}$$

2. Scaling start point offset

The x and y start point of the generated pixels can be adjusted, the offset value is in the range of 0 to 0.99.

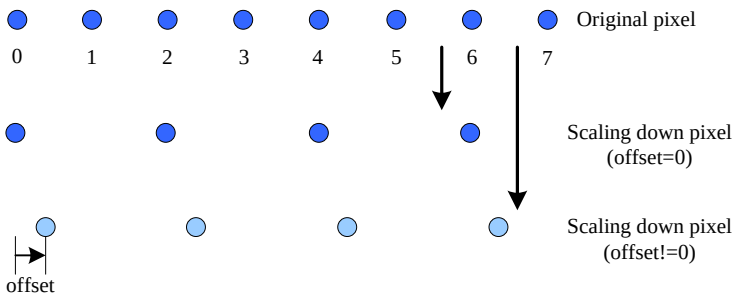


Fig. 18-7 VOP Scaling Down Offset

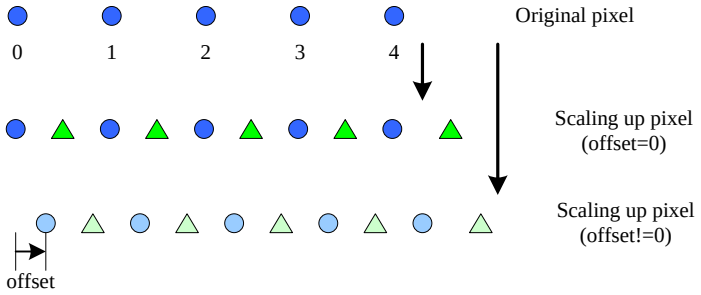


Fig. 18-8 VOP Scaling Up Offset

Table 18-2 VOP Scaling Start Point Offset Registers

scaling down/up start point offset	Offset variable	Register
Win0 YRGB vertical scaling offset	Win0_YRGB_vscl_offset	Win0_SCL_OFFSET [32:24]
Win0 YRGB horizontal scaling offset	Win0_YRGB_hscl_offset	Win0_SCL_OFFSET [23:16]
Win0 Cbr vertical scaling offset	Win0_CBR_vscl_offset	Win0_SCL_OFFSET [15:8]
Win0 Cbr horizontal scaling offset	Win0_CBR_hscl_offset	Win0_SCL_OFFSET [7:0]

1.3.4 De-flicker (Interlace vertical filtering)

It is necessary to display a non-interlaced video signal on an interlaced display (such as TV set). Thus some formof"non-interlaced-to-interlaced conversion" may be required.
The easiest approach is to throw away every other active scan line in each non-interlaced frame. Although the cost is minimal, there are problems with this approach. If there is a sharp vertical transition of color or intensity.It will flicker at one-half the refresh rate.
A better solution is to use two lines of non-interlaced data to generation one line of interlace data. Fast vertical transitions are smoothed out over several interlace lines.
The vertical filtering of two non-interlaced lines can be done by enabling the vertical scaling offset dynamic change in different field (even/odd). The dynamic change value of scaling offset is half of the scaling factor. You should enable the scaling down vertical offset in scaling down mode; enable the scaling up vertical offset in scaling up mode, or one of it in no-scaling mode.

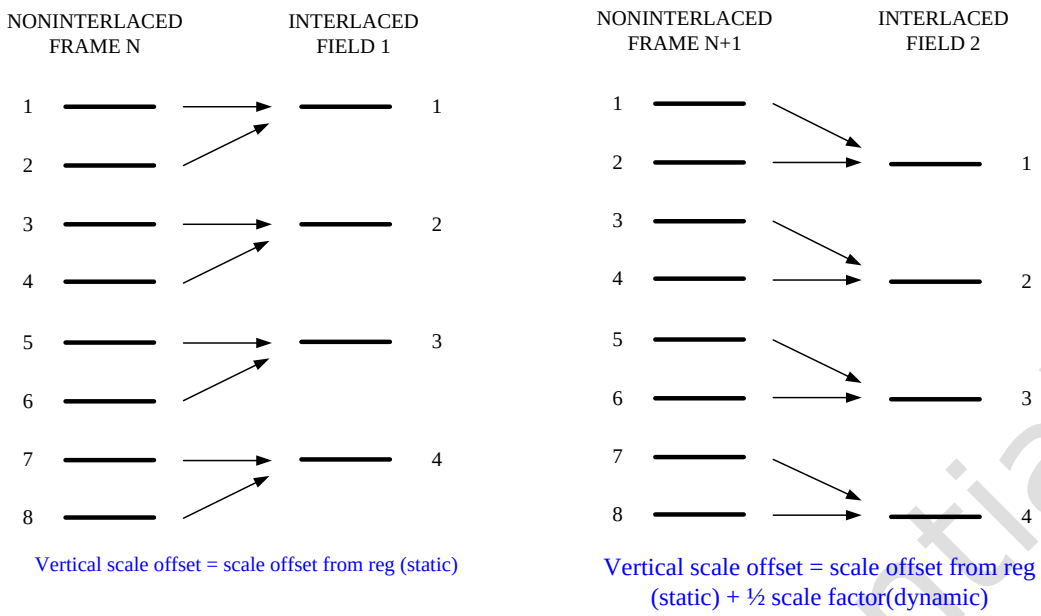


Fig. 18-9 VOP Interlace Vertical Filtering

1.3.5 Overlay

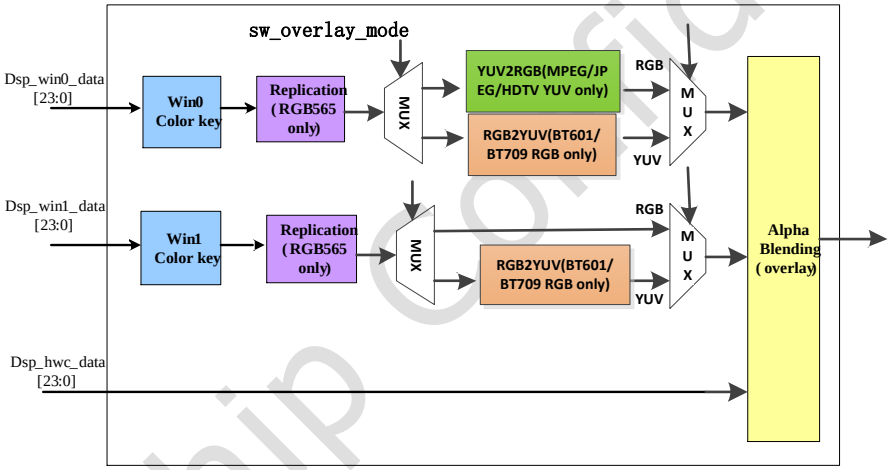


Fig. 18-10 VOP Overlay Block Diagram

1. Overlay display

There are totally 4 layers for overlay display: Background, Win0, Win1 and Hwc.
The background is a programmable solid color layer, which is always the bottom of the display screen.
Hwc is a 32x32 or 64x64 8BPP color palette layer, which is the top layer of the display screen. dsp_hwc_data[23:0] come from the HWC_LUT, which input is a 8BPP data format and output is 24 bits dsp_hwc_data.
The two middle layers are Win0 and Win1. Win1 is on the top of Win0 in default setting, setting VOP_DSP_CTRL0[8] to '1' can let Win0 be on the top of Win1.

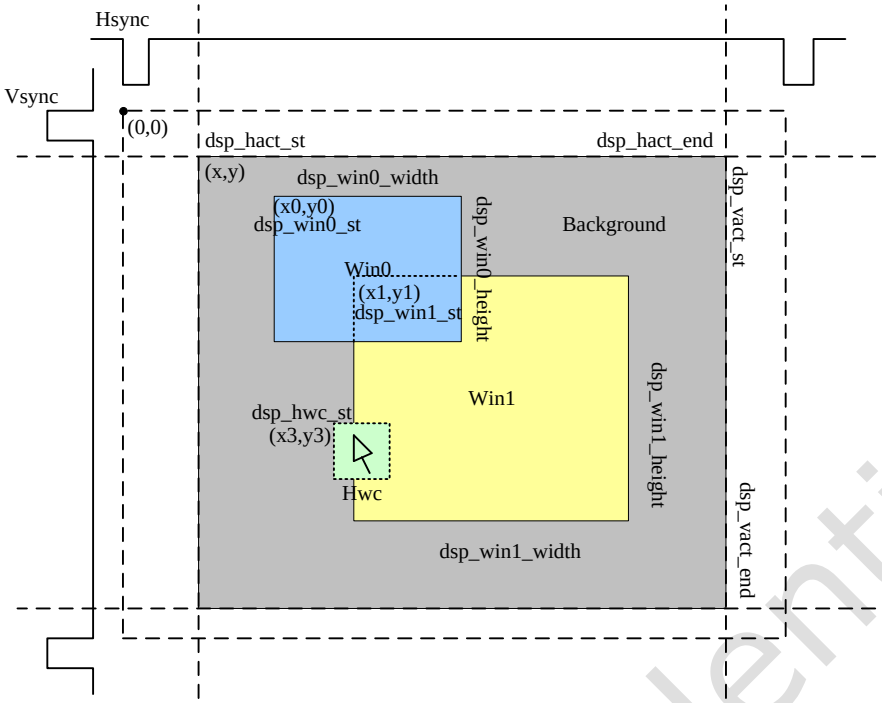


Fig. 18-11 VOP Overlay Display

2. Transparency color key

There are specific registers(VOP_WIN0_COLOR_KEY,VOP_WIN1_COLOR_KEY) for Win0 and Win1 layer to configure the colorkey value.The two transparency color keycan be active at the same time.

The pixel color value is compared to the transparency color key before final display. The transparency color key value defines the pixel data considered asthe transparent pixel. The pixel values with the source color key value are pixels not visibleon the screen, and the under layer pixel values or solid background color are visible.

Transparency color key is done after the scaling module . So transparency color key can only be used in non-scaling mode.

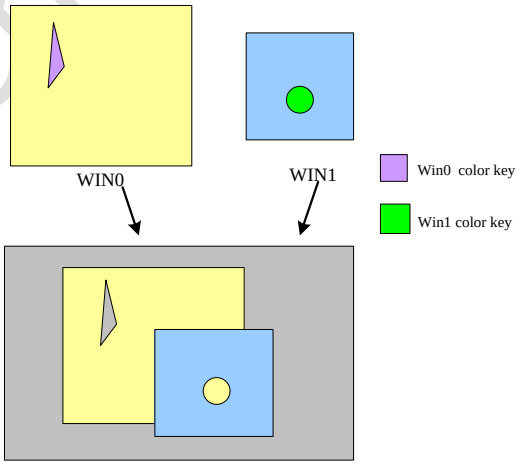


Fig. 18-12 VOP Transparency Color Key

3. Alpha Blending

There are three alpha values for blending between four overlay layers: alpha_win0[7:0], alpha_win1[7:0], alpha_hwc[7:0].

Two blending modes are supported. One is per-pixel (ARGB) mode; the other is user-specified mode. In ARGB mode, the alpha value is in the ARGB data (Win0 and Win1 normal mode only). In user-specified mode, the alpha value comes from the register

(VOP_ALPHA_CTRL[27: 4]).
Pre-multiplied alpha are supported for per-pixel alpha in Win0 and Win 1, for Pre-multiplied alpha, the SRC data has already been multiplied with alpha value.
For win1 layer, it supports alpha scale, and it use "win1_aa_pre_mul" to pre_multiply RGB by alpha in hardware if RGB is not pre_multiplied by software. This is used before scaling alpha channel.

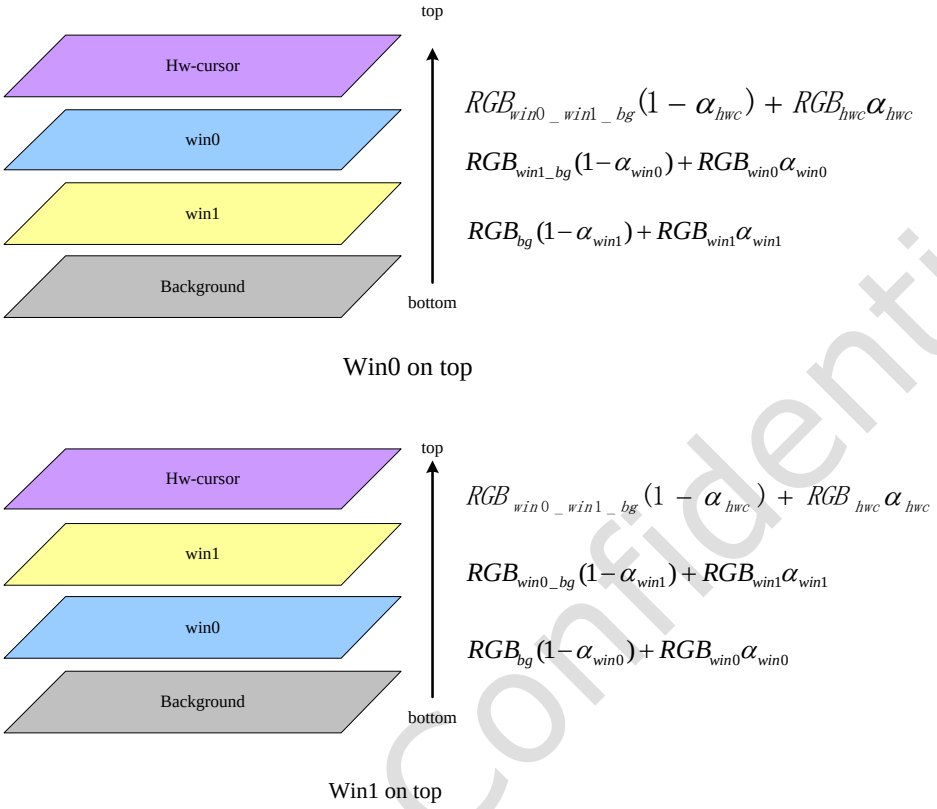


Fig. 18-13 VOP Alpha blending

4.RGB OVERLAY and YCbCr OVERLAY
VOP has a signal named sw_overlay_mode at DSP_CTRL0[31] to decide overlay in RGB or YUV color space.
RGB OVERLAY
All layers overlay in RGB color space. If win0 is YUV420/422/444, it can be onverted to RGB888 by YUV2RGB use MPEG,JPEG and HDTV formula.
YCbCr OVERLAY
All layers overlay in YCbCr color space. It is designed for HDMI/TVE output. When win0/1 input picture format is RGB, it can be converted to YUV444 using BT601 or BT709 formula.
SinceHWC supports BPP8, we can directly use AYUV444 LUT instead of ARGB888 LUT.
The output format of overlay module is YUV. If BCSH is enable, RGB2YUV shoule be disabled, because BCSH works at YCbCr color space. If TVE is enable, YUV2RGB should be disabled in order to make sure thar the input format of TVE is YUV444 and sw_uv_offset_en should be enabled.

1.3.6 Scaler

Scaler is used to support dual panel display for HDMI + PAD.

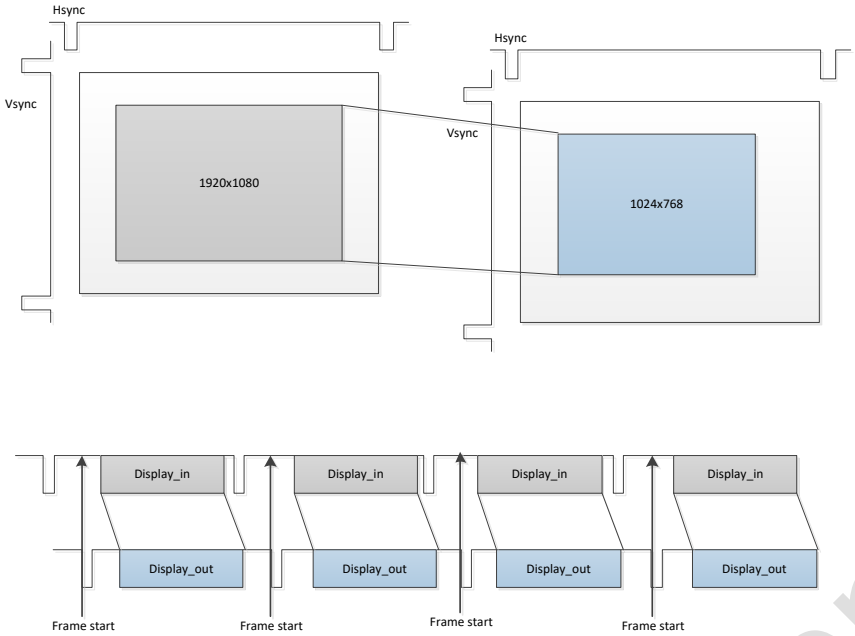


Fig. 18-14 Scaler Frame Sync

The horizontal and vertical scale ratio ofscaler is 0.25~4.
It allows border(black display on the pannel edge) for PAD output,as the figure showed below.

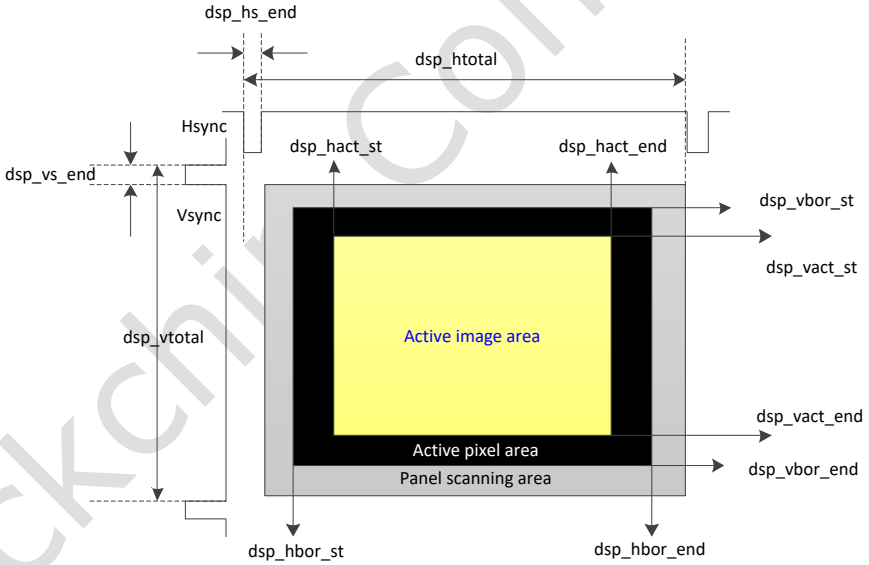


Fig. 18-15 Scaler Output Timing setting

1.3.7 Gamma Correction

Gamma Correction is necessary because most monitors don't have a linear relationship between the voltage and the brightness, which results in your scene looking like it has toomuch contrast andthelight falling off fromthe source outward, happens too quickly. The result can also be problematic if you are going into a composition program.
You can correct this by "Gamma Correction", which allows you to display the images and textures on your computer in an accurate manner.
Your screen is not linear, in that it displays the brightness unevenly.Asa result, the image looksto be morehigh contrast than it should, you end up addingmore lightsor turning up the intensity,or you don't use the lighting in a realistic way thatmatches well with live action scenes. Italso creates problems for you if youusecompositing software.
It consumes 256x8bit LUT for each channel.You can write gamma correction LUT through register" DSP_LUT_ADDR" one by one after set dsp_lut_en = 0.

1.3.8 Replication and dither down

1 Replication(dither up)

If the interface data bus is wider than the pixelformat size, by programming the pixel components replication active/inactive, the MSB is replicated tothe LSB of the interface data bus or the LSB is filled with 0s.

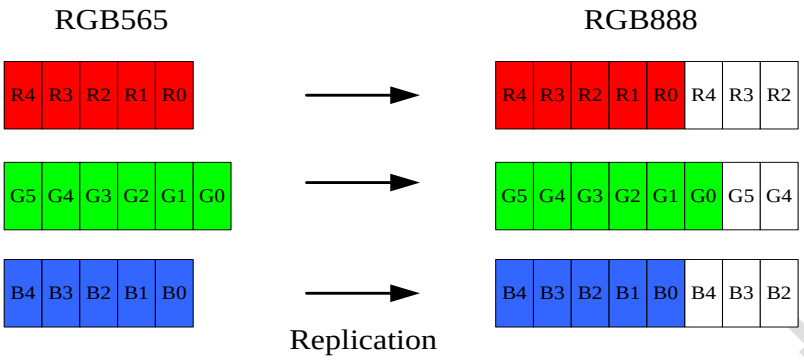


Fig. 18-16 VOP Replication

Dithering is used to improve the quality of display the pixel data in a lower color depth on the LCD panel. The Dithering algorithm is based on the (x,y) pixel position, the value of removed bits and frame counter.

2 Dither down

Here we support three kinds of dithering arithmetic. RGB888 to RGB666 has two ways, one is allegro, the other is FRC. RGB888 to RGB565 has only one arithmetic, which is allegro. When Dithering is not enabled, the MSBs of the pixel color components are output on the interface data bus if the interface data bus is smaller than the pixel format size.

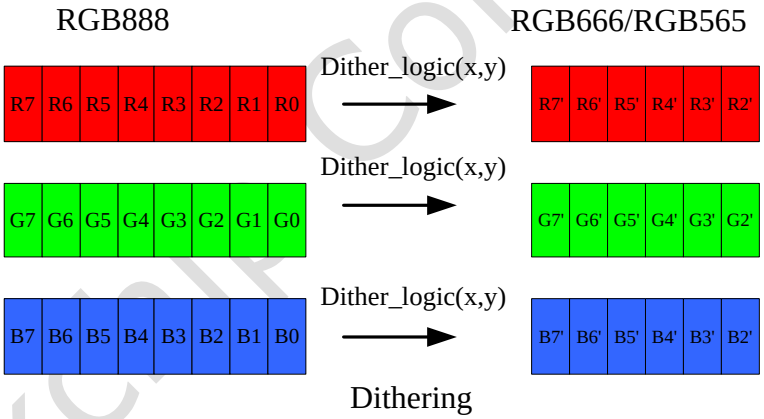


Fig. 18-17 vop dithering

The pattern of dither frc was configured by vopregfile vop_base+0x0e0~0x0f4.
The following figure is the recommended pattern picture in vop, you can config different value of reg 0x1e0~0x1f4，to change the pattern picture.(default value is all 0s here, can give recommended values when initial)

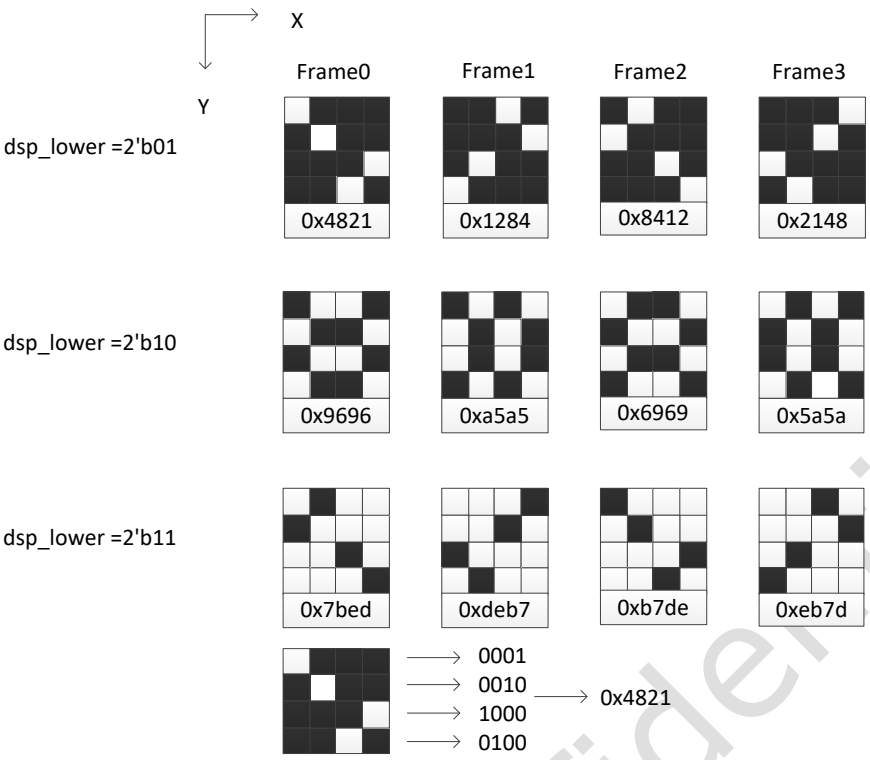


Fig. 18-18 frc pattern diagram

Recommended pattern:
0x0e0 : 0x12844821
0x0e4 : 0x21488412
0x0e8 : 0x55aaaa55
0x0ec : 0x55aaaa55
0x0f0 : 0xdeb77bed
0x0f4 : 0xed7bb7de

1.3.9 TV ENCODE

The TV ENCODE core feature :

- Single 10-bit output to on-chip VDAC
- PAL/NTSC encoding
- Programableluma filter coefficient
- Programableluma/chroma delay
- Programable brightness/contrast

There is a special register to config In TV mode.
tve_mode -> dsp_ctrl0[25]
0 : NTSC mode
1 : PAL mode

Other registers reference to resigister description.
There list NTSC and PAL typical register config as follow:

```
ntsc_config[][2] ={
    {0x00 , 0x000a0000} ,
    {0x04 , 0x00C07a81} ,
    {0x08 , 0x169800FC} ,
    {0x0C , 0x96B40000} ,
    {0x10 , 0x21F07BD7} ,
    {0x14 , 0x02ff0000} ,
    {0x18 , 0xF40202fd} ,
    {0x1C , 0xF332d919} ,
    {0x34 , 0x001500D6} ,
```

```
    {0x38 , 0x0100888C} ,
    {0x3C , 0x00000000} ,
    {0x50 , 0x00000000} ,
    {0x68 , 0x00000000} ,
    {0x78 , 0x0052543C} ,
    {0x8C , 0x00000002} ,
    {0x90 , 0x00008300}};

pal_config[][2] ={
    {0x00 , 0x010a0000} ,
    {0x04 , 0x00C28381} ,
    {0x08 , 0x2694011D} ,
    {0x0C , 0xB6C00880} ,
    {0x10 , 0x2A098ACB} ,
    {0x14 , 0x02ff0000} ,
    {0x18 , 0xF40202fd} ,
    {0x1C , 0xF332d919} ,
    {0x34 , 0x001500F6} ,
    {0x38 , 0x4100088A} ,
    {0x3C , 0x00000000} ,
    {0x50 , 0x00000000} ,
    {0x68 , 0x00000000} ,
    {0x78 , 0x002e553C} ,
    {0x8C , 0x00000022} ,
    {0x90 , 0x00008900}};
```

When working in TV mode ,the dclk is 27Mhz,the sw_core_clk_sel should enable.

1.3.10 VDAC

If vop working in pal/ntscmode ,Tve_dac_dclk_en should be configured to 1, the vdac clk invert can be configured at the same time.
Tve_dac_dclk_en ->vop_base_addr + 0x20[20];
Tve_dac_dclk_inv->vop_base_addr + 0x20[21];

There is a 10 bit on-chip VDAC for the TVE ENCODE.
You should configure the VDAC digital signals. GRF_TVE_CON is the VDAC global register, the high 16bit is the write enable signals map to low 16bits.

VDAC GRF signals as follow:

GRF register	function name	function
GRF_TVE_CON[0]	enextref	Enable control pin to allow internally generated bandgap to be probed in the analog pin 'vbg'. Under normal operation and envbg=1 1'b1: Internal bandgap is passed to vbg pin; 1'b0: vbg pins is at high impedance.
GRF_TVE_CON[1]	envbg	Enable control pin to power up internal bandgap. 1'b1: Internal bandgap is ON; 1'b0: Internal bandgap is OFF.
GRF_TVE_CON[2]	endac	Enable control pin to power up the VDAC

GRF_TVE_CON[3]	ensc	Sense comparator enable for cable connection detection of DAC
GRF_TVE_CON[6:4]	enctr[2:0]	Enable control pins for analog biasing Test. In normal mode set enctr[2:0]=3'b000
GRF_TVE_CON[12:7]	dacgc[5:0]	Gain setting digital input word for VDAC

The detail description of VDAC reference to VDAC chapter.

1.4 Register Description

1.4.1 Internal Address Mapping

Slave address can be divided into different length for different usage, which is shown as follows.

1.4.2 Registers Summary

Name	Offset	Size	Reset Value	Description
VOP_SYS_CTRL	0x0000	W	0x00000000	System control register
VOP_DSP_CTRL0	0x0004	W	0x00000000	Display control register0
VOP_DSP_CTRL1	0x0008	W	0x01000000	Display control register1
VOP_INT_SCALER	0x000c	W	0x00000000	interrupt register about scaler
VOP_INT_STATUS	0x0010	W	0x00000000	Interrupt status register
VOP_ALPHA_CTRL	0x0014	W	0x00000000	Blending control register
VOP_WIN0_COLOR_KEY	0x0018	W	0x00000000	Win0 color key register
VOP_WIN1_COLOR_KEY	0x001c	W	0x00000000	Win1 color key register
VOP_WIN0_YRGB_MST0	0x0020	W	0x00000000	Win0 YRGB memory start address 0
VOP_WIN0_CBR_MST0	0x0024	W	0x00000000	Win0 Cbr memory start address 0
VOP_WIN1_VIR	0x0028	W	0x00000000	Win1 virtual stride
VOP_AXI_BUS_CTRL	0x002c	W	0x15500000	Axi Bus interface control register
VOP_WIN0_VIR	0x0030	W	0x01400140	Win0 virtual stride
VOP_WIN0_ACT_INFO	0x0034	W	0x00ef013f	Win0 active window width/height
VOP_WIN0_DSP_INFO	0x0038	W	0x00ef013f	Win0 display width/height on panel
VOP_WIN0_DSP_ST	0x003c	W	0x000a000a	Win0 display start point on panel
VOP_WIN0_SCL_FACTOR_YRGB	0x0040	W	0x10001000	Win0 YRGB scaling factor
VOP_WIN0_SCL_FACTOR_CBR	0x0044	W	0x10001000	Win0 CBR scaling factor
VOP_WIN0_SCL_OFFSET	0x0048	W	0x00000000	Win0 scaling start point offset
VOP_WIN1_MST	0x004c	W	0x00000000	Win1 memory start address
VOP_WIN1_DSP_INFO	0x0050	W	0x00ef013f	Win1 display width/height on panel
VOP_WIN1_DSP_ST	0x0054	W	0x000a000a	Win1 display start point on panel
VOP_HWC_MST	0x0058	W	0x00000000	Hwc memory start address
VOP_HWC_DSP_ST	0x005c	W	0x000a000a	Hwc display start point on panel

Name	Offset	Size	Reset Value	Description
VOP_DSP_HTOTAL_HS_END	0x006c	W	0x014a000a	Panel scanning horizontal width and hsync pulse end point
VOP_DSP_HACT_ST_END	0x0070	W	0x000a014a	Panel active horizontal scanning start point and end point
VOP_DSP_VTOTAL_VS_END	0x0074	W	0x00fa000a	Panel scanning vertical height and vsync pulse end point
VOP_DSP_VACT_ST_END	0x0078	W	0x000a00fa	Panel active vertical scanning start point and end point
VOP_DSP_VS_ST_END_F1	0x007c	W	0x00000000	Vertical scanning start point and vsync pulse end point of even filed in interlace mode
VOP_DSP_VACT_ST_END_F1	0x0080	W	0x00000000	Vertical scanning active start point and end point of even filed in interlace mode
VOP_GATHER_TRANSFER	0x0084	W	0x00008220	AXI read transfer gather setting
VOP_REG_CFG_DONE	0x0090	W	0x00000000	Register config done flag
VOP_VERSION	0x0094	W	0x00000000	version for vop
VOP_SCALER_CTRL	0x00a0	W	0x00000000	scaler ctrl register
VOP_SCALER_FACTOR	0x00a4	W	0x00000000	scaler module scaling factor
VOP_SCALER_FRAME_ST	0x00a8	W	0x00000000	scaler output scanning start setting register
VOP_SCALER_DSP_HOR_TIMING	0x00ac	W	0x00000000	scaler output scanning horizontal timing
VOP_SCALER_DSP_HACT_TIMING	0x00b0	W	0x00000000	scaler output horizontal active window
VOP_SCALER_DSP_VER_TIMING	0x00b4	W	0x00000000	scaler output scanning vertical timing
VOP_SCALER_DSP_VACT_TIMING	0x00b8	W	0x00000000	scaler output vertical active window
VOP_SCALER_DSP_HBOR_TIMING	0x00bc	W	0x00000000	scaler output horizontal border window
VOP_SCALER_DSP_VBOR_TIMING	0x00c0	W	0x00000000	scaler output vertical border window
VOP_BCSH_CTRL	0x00d0	W	0x00000000	Brightness/Contrast enhancement/Saturation/Hue contrl
VOP_BCSH_COL_BAR	0x00d4	W	0x00000000	Colorbar YUV value
VOP_BCSH_BCS	0x00d8	W	0x00000000	Brightness/Contrast enhancement/Saturation
VOP_BCSH_H	0x00dc	W	0x00000000	Hue
VOP_FRC_LOWER01_0	0x00e0	W	0x00000000	Frc algorithm configuration register 1
VOP_FRC_LOWER01_1	0x00e4	W	0x00000000	Frc algorithm configuration register 2

Name	Offset	Size	Reset Value	Description
VOP_FRC_LOWER10_0	0x00e8	W	0x00000000	Frc algorithm configuration register 3
VOP_FRC_LOWER10_1	0x00ec	W	0x00000000	Frc algorithm configuration register 4
VOP_FRC_LOWER11_0	0x00f0	W	0x00000000	Frc algorithm configuration register 5
VOP_FRC_LOWER11_1	0x00f4	W	0x00000000	Frc algorithm configuration register 6
VOP_TV_CTRL	0x0200	W	0x00000000	tv mode control register
VOP_TV_SYNC_TIMING	0x0204	W	0x00000000	sync timing ctrl register
VOP_TV_ACT_TIMING	0x0208	W	0x00000000	act timing ctrl register
VOP_TV_ADJ_TIMING	0x020c	W	0x00000000	adjust timing register
VOP_TV_FRQ_SC	0x0210	W	0x00000000	Sub-carrier Frequency
VOP_TV_FILTER0	0x0214	W	0x00000000	Filter 0
VOP_TV_FILTER1	0x0218	W	0x00000000	Filter 1
VOP_TV_FILTER2	0x021c	W	0x00000000	Filter 2
VOP_TV_ACT_ST	0x0234	W	0x00000000	
VOP_TV_ROUTING	0x0238	W	0x00000000	Routing
VOP_TV_SYNC_ADJUST	0x0250	W	0x00000000	Sync Adjust
VOP_TV_STATUS	0x0254	W	0x00000000	TV Status register
VOP_TV_RST	0x0268	W	0x00000000	tv reset Control
VOP_TV_SATURATION	0x0278	W	0x00000000	Colour Burst and Saturation
VOP_TV_BANDWIDTH_CTRL	0x028c	W	0x00000000	Chroma bandwidth
VOP_TV_BRIGHTNESS_CONTRAST	0x0290	W	0x00000000	Brightness and Contrast
VOP_MMU_DTE_ADDR	0x0300	W	0x00000000	MMU current page Table address
VOP_MMU_STATUS	0x0304	W	0x00000000	MMU status register
VOP_MMU_COMMAND	0x0308	W	0x00000000	MMU command register
VOP_MMU_PAGE_FAULT_ADDR	0x030c	W	0x00000000	MMU logical address of last page fault
VOP_MMU_ZAP_ONE_LINE	0x0310	W	0x00000000	MMU Zap cache line register
VOP_MMU_INT_RAWSTAT	0x0314	W	0x00000000	MMU raw interrupt status register
VOP_MMU_INT_CLEAR	0x0318	W	0x00000000	MMU raw interrupt status register
VOP_MMU_INT_MASK	0x031c	W	0x00000000	MMU raw interrupt status register
VOP_MMU_INT_STATUS	0x0320	W	0x00000000	MMU raw interrupt status register
VOP_MMU_AUTO_GATING	0x0324	W	0x00000000	mmu auto gating
VOP_HWC_LUT_ADDR	0x0800	W	0x00000000	Access entry for HWC LUT memory(size is word only)
VOP_DSP_LUT_ADDR	0x0c00	W	0x00000000	Access entry for DSP LUT memory(size is word only)

*Notes:Size:**B**- Byte (8 bits) access, **HW**- Half WORD (16 bits) access, **W**-WORD (32 bits) access*

1.4.3 Detail Register Description

VOP_SYS_CTRL
Address: Operational Base + offset (0x0000)
System control register

Bit	Attr	Reset Value	Description
31	RW	0x0	auto_gating_en VOP layer axi-clk auto gating enable 1'b0: disable 1'b1: enable
30	RW	0x0	vop_standby_en VOP standby mode Writing "1" to turn VOP into standby mode, All the layer would disable and the data transfer from frame buffer memory would stop at the end of current frame. The output would be blank. When writing "0" to this bit, standby mode would disable and the VOP go back to work immediately. 1'b0: disable 1'b1: enable * Black display is recommended before setting standby mode enable.
29	RW	0x0	vop_dma_stop VOP DMA stop mode 1'b0: disable 1'b1: enable * If DMA is working, the stop mode would not be active until current bus transfer is finished.
28	RW	0x0	dsp_lut_en Display LUT ram enable 1'b0: disable 1'b1: enable *This bit should be "0" when CPU updates the LUT, and should be "1" when Display LUT mode enable.
27	RO	0x0	reserved
26	RW	0x0	hwc_load_en Hardware cursor data reload enable 1'b0: disable 1'b1: enable *Setting this bit would reload the Hwc data. It would be auto cleared after reload finish.
25:24	RW	0x0	dma_burst_length DMA read Burst length 2'b00: burst16 (burst 15 in rgb888 pack mode) 2'b01: burst8 (burst 12 in rgb888 pack mode) 2'b10: burst4 (burst 6 in rgb888 pack mode)

Bit	Attr	Reset Value	Description
23	RW	0x0	win1_no_outstanding Win1 AXI master read outstanding 1'b0: enable 1'b1: disable
22	RW	0x0	win0_no_outstanding Win0 AXI master read outstanding 1'b0: enable 1'b1: disable
21	RO	0x0	reserved
20	RW	0x0	win1_alpha_swap Win1 RGB alpha swap 1'b0: ARGB 1'b1: RGBA
19	RW	0x0	win1_rb_swap Win1 RGB Red and Blue swap 1'b0: RGB 1'b1: BGR
18	RW	0x0	win0_uv_swap Win0 CbCr swap 1'b0: CrCb 1'b1: CbCr
17	RW	0x0	win0_y8_swap Win0 Y middle 8-bit swap 1'b0: Y3Y2Y1Y0 1'b1: Y3Y1Y2Y0
16	RW	0x0	win0_alpha_swap Win0 RGB alpha swap 1'b0: ARGB 1'b1: RGBA
15	RW	0x0	win0_rb_swap Win0 RGB Red and Blue swap 1'b0: RGB 1'b1: BGR
14:13	RO	0x0	reserved
12	RW	0x0	direct_path_layer_sel direct_path_layer_select 1'b0 : select win0 1'b1 : select win1
11	RW	0x0	direct_path_en iep direct path enable signal 1'b0: disable iep direct path 1'b1: enable iep direct path

Bit	Attr	Reset Value	Description
10	RW	0x0	hwc_size Hwc size select 1'b0: 32x32 1'b1: 64x64
9	RW	0x0	hwc_lut_en 1'b0: disable 1'b1: enable *This bit should be "0" when CPU updates the LUT, and should be "1" when hwc LUT mode enable.
8:6	RW	0x0	win1_data_fmt Win1 source Format 3'b000 : ARGB888 3'b001 : RGB888 3'b010 : RGB565 others: reserved
5:3	RW	0x0	win0_data_fmt Win0 source data Format 3'b000 : ARGB888 3'b001 : RGB888 3'b010 : RGB565 3'b100 : YCbCr420 3'b101 : YCbCr422 3'b110 : YCbCr444 others: reserved
2	RW	0x0	hwc_en Hwc enable bit 1'b0: Hwc layer disable 1'b1: Hwc layer enable
1	RW	0x0	win1_en Win1 enable bit 1'b0: Win1 layer disable 1'b1: Win1 layer enable
0	RW	0x0	win0_en Win0 enable bit 1'b0: Win0 layer disable 1'b1: Win0 layer enable

VOP_DSP_CTRL0
Address: Operational Base + offset (0x0004)
Display control register0

Bit	Attr	Reset Value	Description
31	RW	0x0	sw_overlay_mode 1'b0: overlay in rgb domain 1'b1: overlay in yuv domain

Bit	Attr	Reset Value	Description
30	RW	0x0	alpha_mode_sel1 alpha mode select 1 1'b0 : alpha value no change 1'b1 : alpha = alpha + alpha[7]
29	RW	0x0	alpha_mode_sel0 alpha mode select 0 1'b0 : Non-premultiplied alpha 1'b1 : Premultiplied alpha
28	RW	0x0	hwc_alpha_mode hwc alpha mode select 1'b0: user-defined alpha 1'b1: per-pixel alpha
27	RW	0x0	dither_down_sel dither down mode select 1'b0: allegro 1'b1: FRC
26	RW	0x0	sw_uv_offset_en 1'b0:uv no offset 1'b1:uv value minus 128
25	RW	0x0	tve_mode tve mode select 1'b0 : NTSC mode 1'b1 : PAL mode
24	RO	0x0	reserved
23	RW	0x0	yuv_clip YCrCb clip 1'b0: disable, YCbCr no clip 1'b1: enable, YCbCr clip before YCbCr2RGB *Y clip: 16~235, CBCR clip: 16~239
22	RO	0x0	reserved
21:20	RW	0x0	win0_csc_mode Win0 YUV2RGB Color space conversion: 2'b00: mpeg 2'b01: jpeg 2'b10: hd 2'b11: reserved reused by win0 r2y color space conversion: 2'bX0: BT601 2'bX1: BT709
19	RW	0x0	win1_alpha_mode Win1 alpha mode 1'b0: user-defined alpha 1'b1: per-pixel alpha

Bit	Attr	Reset Value	Description
18	RW	0x0	win0_alpha_mode Win0 alpha mode 1'b0: user-defined alpha 1'b1: per-pixel alpha
17	RW	0x0	win0_cbr_deflick Win0 Cbr deflick mode 1'b0: disable 1'b1: enable
16	RW	0x0	win0_yrgb_deflick Win0 YRGB deflick mode 1'b0: disable 1'b1: enable
15	RW	0x0	win1_interlace_read Win1 interlace read mode 1'b0: disable 1'b1: enable
14	RW	0x0	win0_interlace_read Win0 interlace read mode 1'b0: disable 1'b1: enable
13	RW	0x0	interlace_field_pol Interlace field polarity 1'b0: normal 1'b1: invert
12	RW	0x0	dsp_interlace Interlace display enable 1'b0: disable 1'b1: enable *This mode is related to the TVE output, the display timing of odd field must be set correctly. (vop_dsp_vs_st_end_f1/vop_dsp_vact_end_f1)
11	RW	0x0	dither_down Dither-down enable 1'b0: disable 1'b1: enable
10	RW	0x0	dither_down_mode Dither-down mode 1'b0: RGB888 to RGB565 1'b1: RGB888 to RGB666
9	RW	0x0	dither_up dither up RGB565 to RGB888 enable 1'b0: disable 1'b1: enable

Bit	Attr	Reset Value	Description
8	RW	0x0	dsp_win0_top Win0 and Win1 position swap 1'b0: win1 on the top of win0 1'b1: win0 on the top of win1
7	RW	0x0	dsp_dclk_pol DCLK invert enable 1'b0: normal 1'b1: invert
6	RW	0x0	dsp_den_pol DEN polarity 1'b0: positive 1'b1: negative
5	RW	0x0	dsp_vsync_pol VSYNC polarity 1'b0: negative 1'b1: positive
4	RW	0x0	dsp_hsync_pol HSYNC polarity 1'b0: negative 1'b1: positive
3:0	RW	0x0	dsp_out_mode Display output format 4'b0000: Parallel 24-bit RGB888 output R[7:0],G[7:0],B[7:0] 4'b0001: Parallel 18-bit RGB666 output 6'b0,R[5:0],G[5:0],B[5:0] 4'b0010: Parallel 16-bit RGB565 output 8'b0,R[4:0],G[5:0],B[4:0] Others: Reserved.

VOP_DSP_CTRL1
Address: Operational Base + offset (0x0008)
Display control register1

Bit	Attr	Reset Value	Description
31	RW	0x0	dsp_out_zero DEN、HSYNC、VSYNC output software ctrl 1'b0：normal output 1'b1：all output '0' means:hsync,vsync,den =000
30:29	RO	0x0	reserved
28	RW	0x0	dsp_rg_swap Display output red and green swap enable 1'b0: RGB 1'b1: GRB

Bit	Attr	Reset Value	Description
27	RW	0x0	dsp_rb_swap Display output red and blue swap enable 1'b0: RGB 1'b1: BGR
26	RW	0x0	dsp_bg_swap Display output blue and green swap enable 1'b0: RGB 1'b1: RBG
25	RW	0x0	dsp_black_en Black display mode When this bit enable, the pixel data output is all black (0x000000)
24	RW	0x1	dsp_blank_en Blank display mode When this bit enable, the hsync/vsync/den output is blank. means:hsync,vsync,den =110
23:16	RW	0x00	dsp_bg_red Background Red color Background Red color
15:8	RW	0x00	dsp_bg_green Background Green color Background Green color
7:0	RW	0x00	dsp_bg_blue Background Blue color Background Blue color

VOP_INT_SCALER
Address: Operational Base + offset (0x000c)
interrupt register about scaler

Bit	Attr	Reset Value	Description
31:7	RO	0x0	reserved
6	RW	0x0	HDMI_den_pol DEN polarity 1'b0: positive 1'b1: negative
5	RW	0x0	HDMI_vsync_pol VSYNC polarity 1'b0: negative 1'b1: positive
4	RW	0x0	HDMI_hsync_pol HSYNC polarity 1'b0: negative 1'b1: positive

Bit	Attr	Reset Value	Description
3	RW	0x0	fs_addr_mask_en frame start mask bit 1'b0: disable 1'b1: enable
2	RO	0x0	scaler_empty_intr_status scaler empty interrupt status
1	W1C	0x0	scaler_empty_intr_clr scaler empty interrupt clear (Auto clear)
0	RW	0x0	scaler_empty_intr_en scaler empty interrupt enable 1'b0: disable 1'b1: enable

VOP_INT_STATUS
Address: Operational Base + offset (0x0010)
Interrupt status register

Bit	Attr	Reset Value	Description
31	RO	0x0	line_flag_raw_status Line flag raw Interrupt status
30	RO	0x0	fs_raw_status Frame start raw interrupt status
29	RO	0x0	win1_empty_intr_status win1 data empty interrupt status
28	RO	0x0	win0_empty_intr_status win0 data empty interrupt status
27	W1C	0x0	win1_empty_intr_clr win1 data empty interrupt clear(auto clear)
26	W1C	0x0	win0_empty_intr_clr win0 data empty interrupt clear(auto clear)
25	RW	0x0	win1_empty_intr_en win1 data empty interrupt enable signal 1'b0:disable 1'b1:enable
24	RW	0x0	win0_empty_intr_en win0 data empty interrupt enable signal 1'b0:disable 1'b1:enable
23:12	RW	0x000	dsp_line_frag_num Line number of the Line flag interrupt The display line number when the flag interrupt occur, the range is (0~ DSP_VTOTAL-1).
11	W1C	0x0	bus_error_intr_clr Bus error Interrupt clear (Auto clear)
10	W1C	0x0	line_frag_intr_clr Line flag Interrupt clear (Auto clear)

Bit	Attr	Reset Value	Description
9	W1C	0x0	fs_intr_clr Frame start interrupt clear (Auto clear)
8	W1C	0x0	dsp_hold_valid_intr_clr display hold valid interrupt clear (Auto clear)
7	RW	0x0	bus_error_intr_en Bus error interrupt enable 1'b0: disable 1'b1: enable
6	RW	0x0	line_frag_intr_en Line flag Interrupt enable 1'b0: disable 1'b1: enable
5	RW	0x0	fs_intr_en Frame start interrupt enable 1'b0: disable 1'b1: enable
4	RW	0x0	dsp_hold_valid_intr_en display hold valid interrupt enable 1'b0: disable 1'b1: enable
3	RO	0x0	bus_error_intr Bus error Interrupt status
2	RO	0x0	line_frag_intr Line flag Interrupt status
1	RO	0x0	fs_intr Frame start interrupt status
0	RO	0x0	dsp_hold_valid_intr display hold valid interrupt status

VOP_ALPHA_CTRL
Address: Operational Base + offset (0x0014)
Blending control register

Bit	Attr	Reset Value	Description
31:28	RO	0x0	reserved
27:20	RW	0x00	hwc_alpha_value HWC alpha blending value
19:12	RW	0x00	win1_alpha_value Win1 alpha blending value
11:4	RW	0x00	win0_alpha_value Win0 alpha blending value
3	RO	0x0	reserved
2	RW	0x0	hwc_alpha_en HWC alpha blending enable 1'b0: disable 1'b1: enable

Bit	Attr	Reset Value	Description
1	RW	0x0	win1_alpha_en Win1 alpha blending enable 1'b0: disable 1'b1: enable
0	RW	0x0	win0_alpha_en Win0 alpha blending enable 1'b0: disable 1'b1: enable

VOP_WIN0_COLOR_KEY
Address: Operational Base + offset (0x0018)
Win0 color key register

Bit	Attr	Reset Value	Description
31:25	RO	0x0	reserved
24	RW	0x0	win0_key_en Win0 transparency color key enable 1'b0: disable 1'b1: enable
23:0	RW	0x000000	win0_key_color Win0 key color

VOP_WIN1_COLOR_KEY
Address: Operational Base + offset (0x001c)
Win1 color key register

Bit	Attr	Reset Value	Description
31:25	RO	0x0	reserved
24	RW	0x0	win1_key_en Win1 transparency color key enable 1'b0: disable 1'b1: enable
23:0	RW	0x000000	win1_key_color Win1 key color

VOP_WIN0_YRGB_MST0
Address: Operational Base + offset (0x0020)
Win0 YRGB memory start address 0

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	win0_yrgb0_mst win0 YRGB frame buffer memory start address 0

VOP_WIN0_CBR_MST0
Address: Operational Base + offset (0x0024)
Win0 Cbr memory start address 0

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	win0_cbr0_mst win0 CBR frame buffer memory start address 0

VOP_WIN1_VIR
Address: Operational Base + offset (0x0028)
Win1 virtual stride

Bit	Attr	Reset Value	Description
31:13	RO	0x0	reserved
12:0	RW	0x0000	win1_vir_stride Win1 virtual stride Number of words of Win1 Virtual width ARGB888 : win1_vir_width RGB888 : (win1_vir_width*3/4) + (win1_vir_width%3) RGB565 : ceil(win1_vir_width/2)

VOP_AXI_BUS_CTRL
Address: Operational Base + offset (0x002c)
Axi Bus interface control register

Bit	Attr	Reset Value	Description
31	RW	0x0	sw_io_pad_clk_sel IO pad clk select bit 1'b0: normal dclk out 1'b1: gating dclk out
30	RW	0x0	sw_core_clk_sel 1'b0: select dclk sclk 1'b1: select dclk_div sclk_div
29	RW	0x0	mipi_dclk_inv mipi dclk invert select bit 1'b0:mipi dclk inv disable 1'b1:mipi dclk inv enable
28	RW	0x1	mipi_dclk_en mipi dclk enable bit 1'b0:mipi dclk disable 1'b1:mipi dclk enable
27	RW	0x0	lvds_dclk_inv lvds dclk invert select bit 1'b0:lvds dclk inv disable 1'b1:lvds dclk inv enable
26	RW	0x1	lvds_dclk_en lvds dclk enable bit 1'b0:lvds dclk disable 1'b1:lvds dclk enable

Bit	Attr	Reset Value	Description
25	RW	0x0	rgb_dclk_inv rgb dclk invert select bit 1'b0:rgb dclk inv disable 1'b1:rgb dclk ivn enable
24	RW	0x1	rgb_dclk_en rgb dclk enable bit 1'b0:rgb dclk disable 1'b1:rgb dclk enable
23	RW	0x0	HDMI_dclk_inv HDMI dclk invert select bit 1'b0:HDMI dclk inv disable 1'b1:HDMI dclk inv enable
22	RW	0x1	HDMI_dclk_en HDMI dclk enable bit 1'b0:HDMI dclk disable 1'b1:HDMI dclk enable
21	RW	0x0	tve_dac_dclk_inv tve dclk invert select bit 1'b0:tve dac dclk inv disable 1'b1:tve dac dclk inv enable
20	RW	0x1	tve_dac_dclk_en tve dac dclk enable bit 1'b0:tve dac dclk disable 1'b1:tve dac dclk enable
19	RW	0x0	sw_HDMI_clk_i_sel HDMI io clk select bit 1'b0: select dclk to HDMI io 1'b1: select dclk_core to HDMI io
18:17	RO	0x0	reserved
16:12	RW	0x00	sw_axi_max_outstand_num Max number of AXI read outstanding
11	RW	0x0	sw_axi_max_outstand_en AXI read oustanding limited enable bit 1'b0: disable 1'b1: enable
10	RW	0x0	sw_vop_mmu_en Vop MMU enable bit 1'b0: disable 1'b1: enable
9:6	RW	0x0	sw_noc_hurry_threshold Noc hurry threshold
5:4	RW	0x0	sw_noc_hurry_value Noc hurry level

Bit	Attr	Reset Value	Description
3	RW	0x0	sw_noc_hurry_en Noc hurry mode enable bit 1'b0: disable 1'b1: enable
2:1	RW	0x0	sw_noc_qos_value Noc QoS level
0	RW	0x0	sw_noc_qos_en Noc QoS mode enable bit 1'b0: disable 1'b1: enable

VOP_WIN0_VIR
Address: Operational Base + offset (0x0030)
Win0 virtual stride

Bit	Attr	Reset Value	Description
31:29	RO	0x0	reserved
28:16	RW	0x0140	win0_cbr_vir_stride Number of words of Win0 cbr Virtual width UV420 : ceil(win0_vir_width/4) UV422 : ceil(win0_vir_width/4) UV444 : ceil(win0_vir_width/2)
15:13	RO	0x0	reserved
12:0	RW	0x0140	win0_vir_stride Win0 Virtual stride Number of words of Win0 Virtual width ARGB888 : win0_vir_width RGB888 : (win0_vir_width*3/4) + (win0_vir_width%3) RGB565 : ceil(win0_vir_width/2) YUV : ceil(win0_vir_width/4)

VOP_WIN0_ACT_INFO
Address: Operational Base + offset (0x0034)
Win0 active window width/height

Bit	Attr	Reset Value	Description
31:29	RO	0x0	reserved
28:16	RW	0x00ef	win0_act_height Win0 active(original) window height win_act_height = (win0 vertical size -1)
15:13	RO	0x0	reserved
12:0	RW	0x013f	win0_act_width Win0 active(original) window width win_act_width = (win0 horizontial size -1)

VOP_WIN0_DSP_INFO
Address: Operational Base + offset (0x0038)

Win0 display width/height on panel

Bit	Attr	Reset Value	Description
31:27	RO	0x0	reserved
26:16	RW	0x0ef	dsp_win0_height Win0 display window height win0_dsp_height = (win0 vertical size -1)
15:11	RO	0x0	reserved
10:0	RW	0x13f	dsp_win0_width Win0 display window width win0_dsp_width = (win0 horizontial size -1)

VOP_WIN0_DSP_ST
Address: Operational Base + offset (0x003c)
Win0 display start point on panel

Bit	Attr	Reset Value	Description
31:28	RO	0x0	reserved
27:16	RW	0x00a	dsp_win0_yst Win0 vertical start point(y) of the Panel scanning
15:12	RO	0x0	reserved
11:0	RW	0x00a	dsp_win0_xst Win0 horizontal start point(x) of the Panel scanning

VOP_WIN0_SCL_FACTOR_YRGB
Address: Operational Base + offset (0x0040)
Win0 YRGB scaling factor

Bit	Attr	Reset Value	Description
31:16	RW	0x1000	win0_vs_factor_yrgb Win0 YRGB vertical scaling factor: factor=((VOP_WIN0_ACT_INFO[31:16])/(VOP_WIN0_DSP_INFO[31:16]))*2^12
15:0	RW	0x1000	win0_hs_factor_yrgb Win0 YRGB horizontal scaling factor: factor=((VOP_WIN0_ACT_INFO[15:0])/(VOP_WIN0_DSP_INFO[15:0]))*2^12

VOP_WIN0_SCL_FACTOR_CBR
Address: Operational Base + offset (0x0044)
Win0 CBR scaling factor

Bit	Attr	Reset Value	Description
31:16	RW	0x1000	win0_vs_factor_cbr Win0 CBR vertical scaling factor: YCbCr420: factor=((VOP_WIN0_ACT_INFO[31:16]/ 2)/(VOP_WIN0_DSP_INFO[31:16]))*2^12 YCbCr422,YCbCr444: factor=((VOP_WIN0_ACT_INFO[31:16])/(VOP_WIN0_DSP_INFO[31:16]))*2^12
15:0	RW	0x1000	win0_hs_factor_cbr Win0 CBR horizontal scaling factor: YCbCr422,YCbCr420: factor=((VOP_WIN0_ACT_INFO[15:0]/2)/(VOP_WIN0_DSP_INFO [15:0]))*2^12 YCbCr444: factor=((VOP_WIN0_ACT_INFO[15:0])/(VOP_WIN0_DSP_INFO[1 5:0]))*2^12

VOP_WIN0_SCL_OFFSET
Address: Operational Base + offset (0x0048)
Win0 scaling start point offset

Bit	Attr	Reset Value	Description
31:24	RW	0x00	win0_vs_offset_cbr Cbr Vertical scaling start point offset (0x00~0xff)/0x100 = 0~0.99
23:16	RW	0x00	win0_vs_offset_yrgb Y Vertical scaling start point offset (0x00~0xff)/0x100 = 0~0.99
15:8	RW	0x00	win0_hs_offset_cbr Cbr Horizontal scaling start point offset (0x00~0xff)/0x100 = 0~0.99
7:0	RW	0x00	win0_hs_offset_yrgb Y Horizontal scaling start point offset (0x00~0xff)/0x100 = 0~0.99

VOP_WIN1_MST
Address: Operational Base + offset (0x004c)
Win1 memory start address

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	win1_mst Win1 frame buffer memory start address

VOP_WIN1_DSP_INFO
Address: Operational Base + offset (0x0050)
Win1 display width/height on panel

Bit	Attr	Reset Value	Description
31:27	RO	0x0	reserved
26:16	RW	0x0ef	dsp_win1_height Win1 display window height win1_dsp_height = (win1 dsp vertical size -1)
15:11	RO	0x0	reserved
10:0	RW	0x13f	dsp_win1_width Win1 display window width win1_dsp_width = (win1 dsp horizontial size -1)

VOP_WIN1_DSP_ST
Address: Operational Base + offset (0x0054)
Win1 display start point on panel

Bit	Attr	Reset Value	Description
31:28	RO	0x0	reserved
27:16	RW	0x00a	dsp_win1_yst Win1 vertical start point(y) of the Panel scanning
15:12	RO	0x0	reserved
11:0	RW	0x00a	dsp_win1_xst Win1 horizontal start point(x) of the Panel scanning

VOP_HWC_MST
Address: Operational Base + offset (0x0058)
Hwc memory start address

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	hwc_mst HWC data memory start address

VOP_HWC_DSP_ST
Address: Operational Base + offset (0x005c)
Hwc display start point on panel

Bit	Attr	Reset Value	Description
31:28	RO	0x0	reserved
27:16	RW	0x00a	dsp_hwc_yst HWC vertical start point(y) of the Panel scanning
15:12	RO	0x0	reserved
11:0	RW	0x00a	dsp_hwc_xst HWC horizontal start point(x) of the Panel scanning

VOP_DSP_HTOTAL_HS_END
Address: Operational Base + offset (0x006c)
Panel scanning horizontal width and hsync pulse end point

Bit	Attr	Reset Value	Description
31:28	RO	0x0	reserved

Bit	Attr	Reset Value	Description
27:16	RW	0x14a	dsp_htotal Panel display scanning horizontal period
15:12	RO	0x0	reserved
11:0	RW	0x00a	dsp_hs_end Panel display scanning hsync pulse width

VOP_DSP_HACT_ST_END
Address: Operational Base + offset (0x0070)
Panel active horizontal scanning start point and end point

Bit	Attr	Reset Value	Description
31:28	RO	0x0	reserved
27:16	RW	0x00a	dsp_hact_st Panel display scanning horizontal active start point
15:12	RO	0x0	reserved
11:0	RW	0x14a	dsp_hact_end Panel display scanning horizontal active end point

VOP_DSP_VTOTAL_VS_END
Address: Operational Base + offset (0x0074)
Panel scanning vertical height and vsync pulse end point

Bit	Attr	Reset Value	Description
31:28	RO	0x0	reserved
27:16	RW	0x0fa	dsp_vtotal Panel display scanning vertical period.
15:12	RO	0x0	reserved
11:0	RW	0x00a	dsp_vs_end Panel display scanning vsync pulse width

VOP_DSP_VACT_ST_END
Address: Operational Base + offset (0x0078)
Panel active vertical scanning start point and end point

Bit	Attr	Reset Value	Description
31:28	RO	0x0	reserved
27:16	RW	0x00a	dsp_vact_st Panel display scanning vertical active start point
15:12	RO	0x0	reserved
11:0	RW	0x0fa	dsp_vact_end Panel display scanning vertical active end point

VOP_DSP_VS_ST_END_F1
Address: Operational Base + offset (0x007c)
Vertical scanning start point and vsync pulse end point of even filed in interlace mode

Bit	Attr	Reset Value	Description
31:28	RO	0x0	reserved

Bit	Attr	Reset Value	Description
27:16	RW	0x000	dsp_vs_st_f1 Panel display scanning vertical vsync start point of 2nd field (interlace display mode)
15:12	RO	0x0	reserved
11:0	RW	0x000	dsp_vs_end_f1 Panel display scanning vertical vsync end point of 2nd field(interlace display mode)

VOP_DSP_VACT_ST_END_F1
Address: Operational Base + offset (0x0080)
Vertical scanning active start point and end point of even filed in interlace mode

Bit	Attr	Reset Value	Description
31:28	RO	0x0	reserved
27:16	RW	0x000	dsp_vact_st_f1 Panel display scanning vertical active start point of 2nd field (interlace display mode)
15:12	RO	0x0	reserved
11:0	RW	0x000	dsp_vact_end_f1 Panel display scanning vertical active end point of 2nd field (interlace display mode)

VOP_GATHER_TRANSFER
Address: Operational Base + offset (0x0084)
AXI read transfer gather setting

Bit	Attr	Reset Value	Description
31:16	RO	0x0	reserved
15:12	RW	0x8	win1_axi_gather_num
11	RO	0x0	reserved
10:8	RW	0x2	win0_cbr_axi_gather_num
7:4	RW	0x2	win0_yrgb_axi_gather_num
3	RO	0x0	reserved
2	RW	0x0	win1_axi_gather_en win1 dma channel AXI read transfer gather 1'b0: disable 1'b1: enable
1	RW	0x0	win0_cbr_axi_gather_en win0 cb/cr dma channel AXI read transfer gather 1'b0: disable 1'b1: enable
0	WO	0x0	win0_yrgb_axi_gather_en win0 yrgb dma channel AXI read transfer gather 1'b0: disable 1'b1: enable

VOP_REG_CFG_DONE
Address: Operational Base + offset (0x0090)
Register config done flag

Bit	Attr	Reset Value	Description
31:1	RO	0x0	reserved
0	WO	0x0	reg_load_en vop register config done flag In the first setting of the register, the new value was saved into the mirror register. When all the register config finish, writing this register to enable the copyright of the mirror register to real register. Then register would be updated at the start of every frame.

VOP_VERSION
Address: Operational Base + offset (0x0094)
version for vop

Bit	Attr	Reset Value	Description
31:24	RO	0x00	major IP major vertion used for IP structure
23:16	RO	0x00	minor minor vertion big feature change under same structure
15:0	RO	0x0000	build rtl current svn number

VOP_SCALER_CTRL
Address: Operational Base + offset (0x00a0)
scaler ctrl register

Bit	Attr	Reset Value	Description
31:16	RO	0x0	reserved
15:8	RW	0x00	scaler_in_vsync_vst 0~255,scaler input line num after vsync negedge
7:6	RW	0x0	scaler_in_vsync_mode scl_in_vsync_mode 2'b00:scaler input vsync negedge 2'b01:scaler input trigger for output 2'b10:configurable scaler input frame start 2'b11:reserved
5	RW	0x0	scaler_out_sel 1'b0: scaler output disable 1'b1: scaler output enable
4	RW	0x0	scaler_out_zero DEN、HSYNC、VSYNC output software ctrl 1'b0：normal output 1'b1：all output '0' means:hsync,vsync,den =000

Bit	Attr	Reset Value	Description
3	RW	0x0	scaler_den_inv 1'b0: positive 1'b1: negtive
2	RW	0x0	scaler_sync_inv hsync vsync polarity 1'b0: negative 1'b1: positive
1	RO	0x0	reserved
0	RW	0x0	scaler_en scaler enable bit 1'b0: scaler disable 1'b1: scaler enable

VOP_SCALER_FACTOR
Address: Operational Base + offset (0x00a4)
scaler module scaling factor

Bit	Attr	Reset Value	Description
31:30	RO	0x0	reserved
29:16	RW	0x0000	scaler_v_factor (1)scaling up: (Hin_act-1)/(Hout_act-1)*2^12; (2)scaling down(bilinear): (Hin_act-1)/(Hout_act-1)*2^12;
15:14	RO	0x0	reserved
13:0	RW	0x0000	scaler_h_factor (1)scaling up: (Hin_act-1)/(Hout_act-1)*2^12; (2)scaling down(bilinear): (Hin_act-1)/(Hout_act-1)*2^12;

VOP_SCALER_FRAME_ST
Address: Operational Base + offset (0x00a8)
scaler output scanning start setting register

Bit	Attr	Reset Value	Description
31:28	RO	0x0	reserved
27:16	RW	0x000	scaler_dsp_frame_vst scaler dsp frame vertical start point
15:12	RO	0x0	reserved
11:0	RW	0x000	scaler_dsp_frame_hst scaler dsp frame horizontal start point

VOP_SCALER_DSP_HOR_TIMING
Address: Operational Base + offset (0x00ac)
scaler output scanning horizontal timing

Bit	Attr	Reset Value	Description
31:24	RO	0x0	reserved
23:16	RW	0x00	scaler_dsp_hs_end scaler dsp hs end
15:12	RO	0x0	reserved

Bit	Attr	Reset Value	Description
11:0	RW	0x000	scaler_dsp_htotal scaler dsp htotal

VOP_SCALER_DSP_HACT_TIMING
Address: Operational Base + offset (0x00b0)
scaler output horizontal active window

Bit	Attr	Reset Value	Description
31:26	RO	0x0	reserved
25:16	RW	0x000	scaler_dsp_hact_st scaler dsp horizontal active start point
15:12	RO	0x0	reserved
11:0	RW	0x000	scaler_dsp_hact_end scaler dsp horizontal active end point

VOP_SCALER_DSP_VER_TIMING
Address: Operational Base + offset (0x00b4)
scaler output scanning vertical timing

Bit	Attr	Reset Value	Description
31:24	RO	0x0	reserved
23:16	RW	0x00	scaler_dsp_vs_end scaler dsp vs end
15:12	RO	0x0	reserved
11:0	RW	0x000	scaler_dsp_vtotal scaler dsp vtotal

VOP_SCALER_DSP_VACT_TIMING
Address: Operational Base + offset (0x00b8)
scaler output vertical active window

Bit	Attr	Reset Value	Description
31:24	RO	0x0	reserved
23:16	RW	0x00	scaler_dsp_vact_st scaler dsp vertical active start point
15:12	RO	0x0	reserved
11:0	RW	0x000	scaler_dsp_vact_end scaler dsp vertical active end point

VOP_SCALER_DSP_HBOR_TIMING
Address: Operational Base + offset (0x00bc)
scaler output horizontal border window

Bit	Attr	Reset Value	Description
31:26	RO	0x0	reserved
25:16	RW	0x000	scaler_dsp_hbor_st scaler dsp horizontal border start point
15:12	RO	0x0	reserved

Bit	Attr	Reset Value	Description
11:0	RW	0x000	scaler_dsp_hbor_end scaler dsp horizontal border end point

VOP_SCALER_DSP_VBOR_TIMING
Address: Operational Base + offset (0x00c0)
scaler output vertical border window

Bit	Attr	Reset Value	Description
31:24	RO	0x0	reserved
23:16	RW	0x00	scaler_dsp_vbor_st scaler dsp vertical border start point
15:12	RO	0x0	reserved
11:0	RW	0x000	scaler_dsp_vbor_end scaler dsp vertical border end point

VOP_BCSH_CTRL
Address: Operational Base + offset (0x00d0)
Brightness/Contrast enhancement/Saturation/Hue contrl

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7	RW	0x0	sw_bcsh_r2y_en 1'b0:bypass 1'b1:enable
6	RW	0x0	sw_bcsh_y2r_en 1'b0:bypass 1'b1:enable
5:4	RW	0x0	sw_bcsh_y2r_csc_mode Color space conversion: 2'b00: mpeg 2'b01: jpeg 2'b10: hd 2'b11: Bypass
3:2	RW	0x0	video_mode 2'b00 : black 2'b01 : blue 2'b10 : color bar 2'b11 : normal video
1	RW	0x0	sw_bcsh_r2y_csc_mode Color space conversion: 1'b0: BT601 1'b1: BT709
0	RW	0x0	bcsh_en 1'b0 : bcsh bypass 1'b1 : bcsh enable

VOP_BCSH_COL_BAR
Address: Operational Base + offset (0x00d4)
Colorbar YUV value

Bit	Attr	Reset Value	Description
31:24	RO	0x0	reserved
23:16	RW	0x00	color_bar_v
15:8	RW	0x00	color_bar_u
7:0	RW	0x00	color_bar_y

VOP_BCSH_BCS
Address: Operational Base + offset (0x00d8)
Brightness/Contrast enhancement/Saturation

Bit	Attr	Reset Value	Description
31:25	RO	0x0	reserved
24:16	RW	0x000	sat_con Saturation*Contrast*256 : 0,1.992*1.992
15:8	RW	0x00	contrast Contrast*256 : 0,1.992
7:6	RO	0x0	reserved
5:0	RW	0x00	brightness Brightness : -128,127

VOP_BCSH_H
Address: Operational Base + offset (0x00dc)
Hue

Bit	Attr	Reset Value	Description
31:16	RO	0x0	reserved
15:8	RW	0x00	cos_hue cos hue value
7:0	RW	0x00	sin_hue sin hue value

VOP_FRC_LOWER01_0
Address: Operational Base + offset (0x00e0)
Frc algorithm configuration register 1

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	lower01_frm1
15:0	RW	0x0000	lower01_frm0

VOP_FRC_LOWER01_1
Address: Operational Base + offset (0x00e4)
Frc algorithm configuration register 2

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	lower01_frm3
15:0	RW	0x0000	lower01_frm2

VOP_FRC_LOWER10_0
Address: Operational Base + offset (0x00e8)
Frc algorithm configuration register 3

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	lower10_frm1
15:0	RW	0x0000	lower10_frm0

VOP_FRC_LOWER10_1
Address: Operational Base + offset (0x00ec)
Frc algorithm configuration register 4

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	lower10_frm3
15:0	RW	0x0000	lower10_frm2

VOP_FRC_LOWER11_0
Address: Operational Base + offset (0x00f0)
Frc algorithm configuration register 5

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	lower11_frm1
15:0	RW	0x0000	lower11_frm0

VOP_FRC_LOWER11_1
Address: Operational Base + offset (0x00f4)
Frc algorithm configuration register 6

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	lower11_frm3
15:0	RW	0x0000	lower11_frm2

VOP_TV_CTRL
Address: Operational Base + offset (0x0200)
tv mode control register

Bit	Attr	Reset Value	Description
31:25	RO	0x0	reserved
24	RW	0x0	cvbs_mode 1'b0: Encoding is NTSC 1'b1: Encoding is PAL
23:20	RO	0x0	reserved
19:18	RW	0x0	clk_upstream_en 2'b00: Reserved 2'b01: Upstream enable is 1 * pix_clk (HDTV) 2'b10: Upstream enable is 1/2* pix_clk (SDTV) 2'b11: Reserved

Bit	Attr	Reset Value	Description
17:16	RW	0x0	timing_en 2'b00: Reserved 2'b01: Timing enable is 1 * pix_clk (HDTV) 2'b10: Timing enable is 1 * pix_clk (SDTV) 2'b11: Reserved
15:11	RO	0x0	reserved
10:9	RW	0x0	filter_gain 2'b00: Luma Filter gain is 1.0 (default value) 2'b01: Luma Filter gain is 0.5 2'b10: Luma Filter gain is 2.0 2'b11: Reserved
8	RW	0x0	filter_upsample 1'b0: Luma Filter output is sampled at 13.5 MHz 1'b1: Luma Filter output is up-sampled to 27 MHz
7:3	RO	0x0	reserved
2:1	RW	0x0	csc_path_sel 2'b00: RGB input to RGB+YUV444 output 2'b01: YUV422 input to YUV444 output 2'b10: Reserved 2'b11: Bypass (YUV444 input to YUV444 output)
0	RO	0x0	reserved

VOP_TV_SYNC_TIMING
Address: Operational Base + offset (0x0204)
sync timing ctrl register

Bit	Attr	Reset Value	Description
31:24	RO	0x0	reserved
23:16	RW	0x00	h_fp Delay to the end of colour burst from H Sync Start , num of 27Mhz clk cycle
15:8	RW	0x00	h_bp Delay to the start of the colour burst from H Sync Start , num of 27Mhz clk cycle
7:0	RW	0x00	h_pw Width of horizontal sync from H Sync Start , num of 27Mhz clk cycle

VOP_TV_ACT_TIMING
Address: Operational Base + offset (0x0208)
act timing ctrl register

Bit	Attr	Reset Value	Description
31:30	RO	0x0	reserved

Bit	Attr	Reset Value	Description
29:28	RW	0x0	tv_syncs 4'b0000: Sync generator is stopped 4'b0001: 525/60 interlaced 4'b0010: 625/50 interlaced other : reserved
27:16	RW	0x000	h_act_end Delay to the end of active video from H Sync Start , num of 27Mhz clk cycle
15:13	RO	0x0	reserved
12	RW	0x0	sc_rst_en Subcarrier reset enable 1'b0: Subcarrier phase is not reset regularly 1'b1: Subcarrier phase is reset every four fields (NTSC) or eight fields (PAL)
11:0	RW	0x000	h_act_st Delay to the start of the active video from H Sync Start , num of 27Mhz clk cycle

VOP_TV_ADJ_TIMING
Address: Operational Base + offset (0x020c)
adjust timing register

Bit	Attr	Reset Value	Description
31:28	RW	0x0	luma_delay_1 Programmable delay for Luma relative to Chroma
27:16	RW	0x000	h_total Total number of output pixels per line. 525/60 = 1716 625/50 = 1728
15:8	RO	0x0	reserved
7:0	RW	0x00	sc_phase Phase offset applied if subcarrier phase is reset (1.4 degrees per lsb)

VOP_TV_FRQ_SC
Address: Operational Base + offset (0x0210)
Sub-carrier Frequency

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	dds_incr colour subcarrier counter . Recommended settings (for 27.00 MHz) are: NTSC: 0x21F07BD7 PAL : 0x2A098ACB

VOP_TV_FILTER0
Address: Operational Base + offset (0x0214)

Filter 0

Bit	Attr	Reset Value	Description
31:24	RW	0x00	coeff3 filter coefficients 3
23:16	RW	0x00	coeff2 filter coefficients 2
15:8	RW	0x00	coeff1 filter coefficients 1
7:0	RW	0x00	coeff0 filter coefficients 0

VOP_TV_FILTER1
Address: Operational Base + offset (0x0218)
Filter 1

Bit	Attr	Reset Value	Description
31:24	RW	0x00	coeff7 filter coefficients 7
23:16	RW	0x00	coeff6 filter coefficients 6
15:8	RW	0x00	coeff5 filter coefficients 5
7:0	RW	0x00	coeff4 filter coefficients 4

VOP_TV_FILTER2
Address: Operational Base + offset (0x021c)
Filter 2

Bit	Attr	Reset Value	Description
31:24	RW	0x00	coeff11 filter coefficients 11
23:16	RW	0x00	coeff10 filter coefficients 10
15:8	RW	0x00	coeff9 filter coefficients 9
7:0	RW	0x00	coeff8 filter coefficients 8

VOP_TV_ACT_ST
Address: Operational Base + offset (0x0234)

Bit	Attr	Reset Value	Description
31:27	RO	0x0	reserved
26:16	RW	0x000	v_offset Vertical picture start offset (independent of picture blanking)
15:12	RO	0x0	reserved

Bit	Attr	Reset Value	Description
11:0	RW	0x000	h_offset Horizontal picture start offset (independent of picture blanking)

VOP_TV_ROUTING
Address: Operational Base + offset (0x0238)
Routing

Bit	Attr	Reset Value	Description
31	RW	0x0	RB_swap 1'b0: No swap 1'b1: Swap Blue/Pb and Red/Pr channels on DACs, relative to the routing shown in 30:28
30:28	RW	0x0	dac_fmt DAC3 DAC2 DAC1 DAC0 3'b000: CVBS Blue Green Red 3'b001: CVBS1 CVBS2 Y C 3'b011: Y CVBS2 CVBS1 C 3'b100: CVBS Pb Y Pr 3'b101: C Pb Y Pr
27:24	RW	0x0	sense_on When ON, outputs a steady mid level on the selected DAC(s) to allow load sensing via DAC or external comparators (application dependent). May be applied singly or together DAC3 DAC2 DAC1 DAC0 4'b0000: OFF OFF OFF OFF 4'b0001: OFF OFF OFF ON 4'b0010: OFF OFF ON OFF 4'b0100: OFF ON OFF OFF 4'b1000: ON OFF OFF OFF 4'b1111: ON ON ON ON
23:20	RO	0x0	reserved
19:16	RW	0x0	setup_on When ON, adds 7.5IRE setup to the specified channel Luma Red/Pr Green Blue/Pb 4'b0000: OFF OFF OFF OFF 4'bXXX1: ON 4'bXX1X: ON 4'bX1XX: ON 4'b1XXX: ON

Bit	Attr	Reset Value	Description
15:12	RW	0x0	AGC_pluse_on When ON, adds AGC and EOF pulses to the specified channel. N.B. This field should not be used for disabling macrovision functionality. Y/C Red/Pr Green Blue/Pb 4'b0000: OFF OFF OFF OFF 4'bXXX1: ON 4'bXX1X: ON 4'bX1XX: ON 4'b1XXX: ON
11:8	RW	0x0	video_on When ON, adds appropriate video format to the specified channel Y/C Red/Pr Green Blue/Pb 4'b0000 : OFF OFF OFF OFF 4'bXXX1: ON 4'bXX1X: ON 4'bX1XX: ON 4'b1XXX: ON
7:4	RW	0x0	sync_on When ON, adds composite sync to the specified channel N.B. Macrovision pseudo-sync pulses (if enabled) will be added to each channel that has sync enabled. Luma Red/Pr Green Blue/Pb 4'b0000: OFF OFF OFF OFF 4'bXXX1: ON 4'bXX1X: ON 4'bX1XX: ON 4'b1XXX: ON
3	RW	0x0	YPP 1'b0: Component output is RGB 1'b1: Component output is YPbPr
2	RW	0x0	chroma_off 1'b0: Chroma is switched ON (normal colour display) 1'b1: Chroma is switched OFF (monochrome display)
1	RW	0x0	Picture_Sync_amplitudes1 1'b0 : Composite has picture/sync ratio of 714/286 (NTSC) 1'b1 : Composite has picture/sync ratio of 700/300 (PAL)
0	RO	0x0	reserved

VOP_TV_SYNC_ADJUST

Address: Operational Base + offset (0x0250)

Sync Adjust

Bit	Attr	Reset Value	Description
31:12	RO	0x0	reserved

Bit	Attr	Reset Value	Description
11:8	RW	0x0	line_adj Signed number of lines by which to change frame length. Affects every frame whilst a non-zero value.
7:0	RW	0x00	pix_adj Signed number of (27MHz) pixels by which to adjust frame length. Affects every frame whilst a non-zero value.

VOP_TV_STATUS
Address: Operational Base + offset (0x0254)
TV Status register

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7	RO	0x0	pixel_manager_vstt Pixel Manager Vertical Start State 1'b0: Vertical Start not active 1'b1: Vertical Start active
6:4	RO	0x0	pixel_manager_sta Pixel Manager Flow Controller State 3'b000: Idle 3'b001: Seeking 3'b010: Ready 3'b011: First pixel 3'b100: Active 3'b111: Other
3:2	RO	0x0	reserved
1	RO	0x0	sync_gen_VBI Sync Generator Vertical Blanking Interval 1'b0: In VBI 1'b1: Not in VBI
0	RO	0x0	sync_gen_FID Sync Generator Field Identity 1'b0: First Field 1'b1: Second Filed

VOP_TV_RST
Address: Operational Base + offset (0x0268)
tv reset Control

Bit	Attr	Reset Value	Description
31:2	RO	0x0	reserved
1	RW	0x0	disp_rst Software reset for all other TVE functions: 1'b0: Normal operation. 1'b1: Software reset (please do NOT reset the TVE every field or frame).

Bit	Attr	Reset Value	Description
0	RO	0x0	reserved

VOP_TV_SATURATION
Address: Operational Base + offset (0x0278)
Colour Burst and Saturation

Bit	Attr	Reset Value	Description
31:24	RO	0x0	reserved
23:16	RW	0x00	burst_size Colour burst amplitude
15:8	RW	0x00	V_weight Conversion factor for Cr to V
7:0	RW	0x00	U_weight Conversion factor for Cb to U

VOP_TV_BANDWIDTH_CTRL
Address: Operational Base + offset (0x028c)
Chroma bandwidth

Bit	Attr	Reset Value	Description
31:6	RO	0x0	reserved
5:4	RW	0x0	chroma_bandwidth 2'b00: Chroma bandpass filter is bypassed 2'b01: Chroma bandpass filter is centred on 3.58 MHz 2'b10: Chroma bandpass filter is centred on 4.43 MHz
3:0	RW	0x0	cdiff_bandwidth 4'b0000: Colour difference filters OFF (no colour) 4'b0001: Colour difference bandwidth is 0.6 MHz 4'b0010: Colour difference bandwidth is 1.3 MHz 4'b0011: Colour difference bandwidth is 2.0 MHz

VOP_TV_BRIGHTNESS_CONTRAST
Address: Operational Base + offset (0x0290)
Brightness and Contrast

Bit	Attr	Reset Value	Description
31:16	RO	0x0	reserved
15:8	RW	0x00	Contrast This value determines the gain of the luma channel. It does not affect the chroma gain.
7:0	RW	0x00	Brightness This value determines the black level on the luma channel during active video only. The setup (if applicable) is applied in addition to this value.

VOP_MMU_DTE_ADDR
Address: Operational Base + offset (0x0300)
MMU current page Table address

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	MMU_DTE_ADDR

VOP_MMU_STATUS
Address: Operational Base + offset (0x0304)
MMU status register

Bit	Attr	Reset Value	Description
31:11	RO	0x0	reserved
10:6	RO	0x00	PAGE_FAULT_BUS_ID Index of master reponsible for last page fault
5	RO	0x0	PAGE_FAULT_IS_WRITE The direction of access for last page fault: 0 = Read 1 = Write
4	RO	0x0	REPLAY_BUFFER_EMPTY The MMU replay buffer is empty
3	RO	0x0	MMU_IDLE The MMU is idle when accesses are being translated and there are no unfinished translated accesses.
2	RO	0x0	STAIL_ACTIVE MMU stall mode currently enabled. The mode is enabled by command
1	RO	0x0	PAGE_FAULT_ACTIVE MMU page fault mode currently enabled . The mode is enabled by command.
0	RO	0x0	PAGING_ENABLED Paging is enabled

VOP_MMU_COMMAND
Address: Operational Base + offset (0x0308)
MMU command register

Bit	Attr	Reset Value	Description
31:3	RO	0x0	reserved
2:0	WO	0x0	MMU_CMD MMU_CMD. This can be: 3'b000: MMU_ENABLE_PAGING 3'b001: MMU_DISABLE_PAGING 3'b010: MMU_ENABLE_STALL 3'b011: MMU_DISABLE_STALL 3'b100: MMU_ZAP_CACHE 3'b101: MMU_PAGE_FAULT_DONE 3'b110: MMU_FORCE_RESET

VOP_MMU_PAGE_FAULT_ADDR
Address: Operational Base + offset (0x030c)
MMU logical address of last page fault

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	PAGE_FAULT_ADDR address of last page fault

VOP_MMU_ZAP_ONE_LINE
Address: Operational Base + offset (0x0310)
MMU Zap cache line register

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	MMU_ZAP_ONE_LINE address to be invalidated from the page table cache

VOP_MMU_INT_RAWSTAT
Address: Operational Base + offset (0x0314)
MMU raw interrupt status register

Bit	Attr	Reset Value	Description
31:2	RO	0x0	reserved
1	RW	0x0	READ_BUS_ERROR read bus error
0	RW	0x0	PAGE_FAULT page fault

VOP_MMU_INT_CLEAR
Address: Operational Base + offset (0x0318)
MMU raw interrupt status register

Bit	Attr	Reset Value	Description
31:2	RO	0x0	reserved
1	WO	0x0	READ_BUS_ERROR read bus error
0	WO	0x0	PAGE_FAULT page fault

VOP_MMU_INT_MASK
Address: Operational Base + offset (0x031c)
MMU raw interrupt status register

Bit	Attr	Reset Value	Description
31:2	RO	0x0	reserved
1	RW	0x0	READ_BUS_ERROR read bus error
0	RW	0x0	PAGE_FAULT page fault

VOP_MMU_INT_STATUS
Address: Operational Base + offset (0x0320)
MMU raw interrupt status register

Bit	Attr	Reset Value	Description
31:2	RO	0x0	reserved
1	RO	0x0	READ_BUS_ERROR read bus error
0	RO	0x0	PAGE_FAULT page fault

VOP_MMU_AUTO_GATING
Address: Operational Base + offset (0x0324)
mmu auto gating

Bit	Attr	Reset Value	Description
31:1	RO	0x0	reserved
0	RW	0x0	mmu_auto_gating mmu auto gating when it is 1'b1, the mmu will auto gating it self

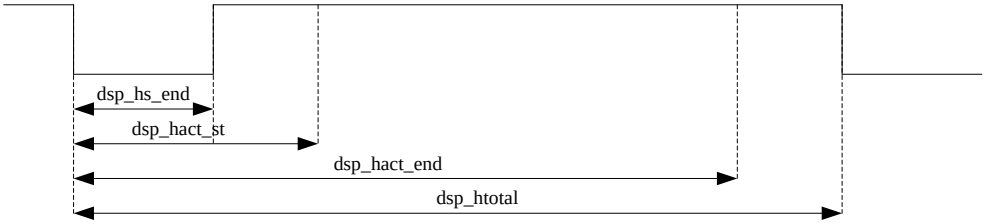
VOP_HWC_LUT_ADDR
Address: Operational Base + offset (0x0800)
Access entry for HWC LUT memory(size is word only)

Bit	Attr	Reset Value	Description
31:24	RO	0x0	reserved
23:0	RW	0x000000	dsp_lut_addr Access entry for DSP LUT memory(size is word only)

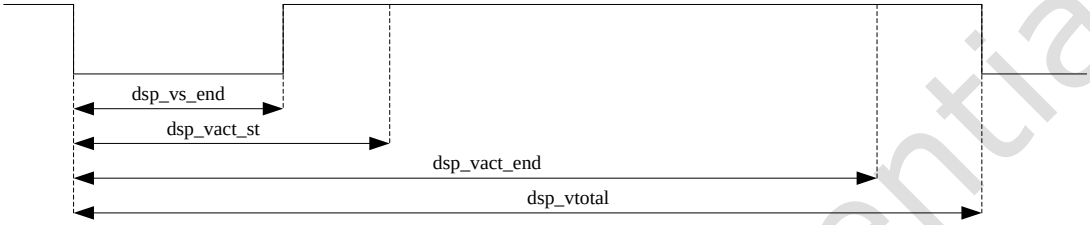
VOP_DSP_LUT_ADDR
Address: Operational Base + offset (0x0c00)
Access entry for DSP LUT memory(size is word only)

Bit	Attr	Reset Value	Description
31:24	RO	0x0	reserved
23:0	RW	0x000000	dsp_lut_addr Access entry for DSP LUT memory(size is word only)

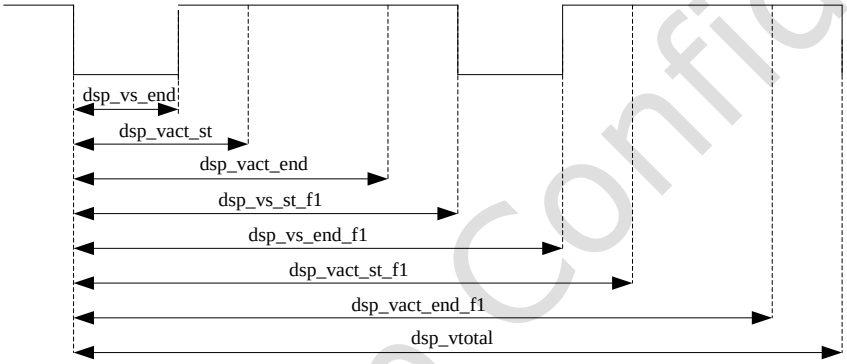
1.5 Timing Diagram



Horizontal timing



Vertical timing(Progressive mode)



Vertical timing(Interlace mode)

Fig. 18-19 VOP RGB interface timing setting

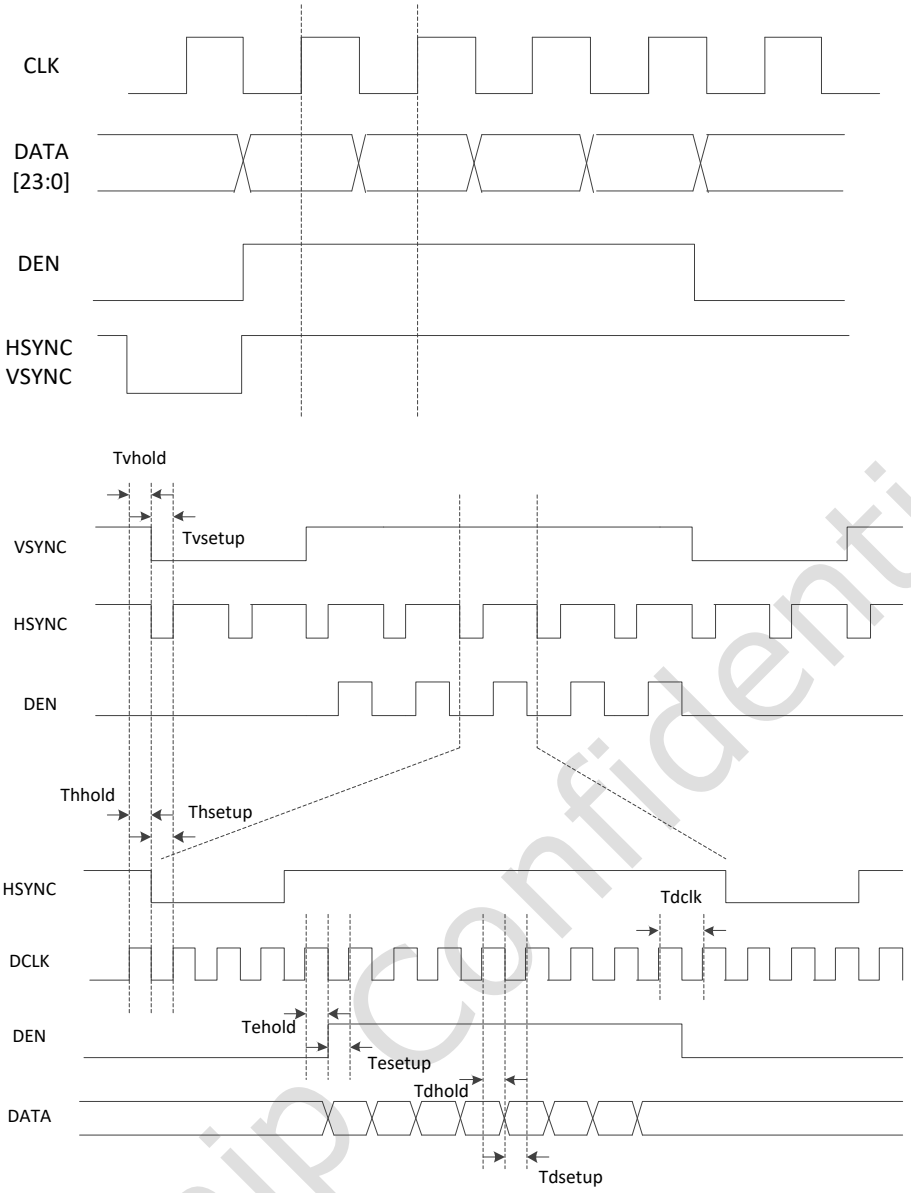


Fig. 18-20 VOP RGB Interface Timing(SDR)

Table 18-3 VOP RGB interface(SDR) signal timing constant
(VDD_core =0.9V to 1.1V, VDD_IO=3.0V to 3.6V , TA = -40℃ot 125℃)

Item	Symbol	Min	Typ	Max	Unit
Display clock period	Tdclk	2.2		6	ns
VSYNC setup to DCLK falling edge	Tvsetup	0.5		0.6	ns
VSYNC hold from DCLK falling edge	Tvhold	1.8		2.0	ns
HSYNC setup to DCLK falling edge	Thsetup	0.50		0.58	ns
HSYNC hold from DCLK falling edge	Thhold	1.89		2.13	ns
DEN setup to DCLK falling edge	Tesetup	0.48		0.56	ns
DEN hold from DCLK falling edge	Tehold	1.85		2.1	ns
DATA setup to DCLK falling edge	Tdsetup	0.47		0.51	ns
DATA hold from DCLK falling edge	Tdhold	4.79		4.26	ns

1.6 Interface Description

1.6.1 VOP Outputs

VOP supports RGB, LVDS, HDMI, MIPI,and TVE output. the LVDS HDMI MIPI output interface is same as RGB interface, TVE output is connected to a DAC(10bit) . VOP is suitable for different display mode by different usage, which is shown as follows.

Table 18-4 VOP Control Pins Definition

Display mode	RGB Parallel 24-bit	RGB Parallel 18-bit	RGB Parallel 16-bit
DCLK	DCLK	DCLK	DCLK
VSYNC	VSYNC	VSYNC	VSYNC
HSYNC	HSYNC	HSYNC	HSYNC
DEN	DEN	DEN	DEN
DATA	DATA[23:0]	DATA[17:0]	DATA[15:0]

1.7 Application Notes

1.7.1 DMA transfer mode

There are three DMA transfer modes for loading win0 or win1 frame data determined by following parameters(X=0,1):
dma_burst_length
winX_no_outstanding
winX_gather_en
winX_gather_thres

1 auto outstanding transfer mode(random transfer)
When winX_no_outstanding is 0, multi-bursts transfer command could be sent out to AXI master interface continuously if the internal memory has enough space to store new data. The continuous random burst number is in the range of 1 to 4, mainly depending on the empty level of internal memory, dma_burst_length, data format and active image width.

2 configured outstanding transfer mode(fixed transfer)
When winX_gather_en is 1, fixed-number of bursts transfer command should be sent out to AXI master interface continuously if the internal memory has enough space to store new data. The fixed-number is determined by winX_gather_thres. Since the internal memory size is limited, there is some restriction for the winX_gather_thres as follows.

Table 18-5 Gather configuration for all format

Gather Threshold	dma_burst_length =2'b00(burst16)	dma_burst_length =2'b01(burst8)	dma_burst_length =2'b10(burst4)
YCbCr420 YCbCr422 YCbCr444	0	0,1,2	0,1,2,3
ARGB888 RGB888 RGB565	0,1,2,3	0,1,2,3	0,1,2,3
8BPP	0,1,2,3	0,1,2,3	0,1,2,3

1.7.2 HWC data load

Hardware cursor data is refreshed when hwc_load_en is high, but not needed for every frame. And hwc_load_en will be high until hwc loading finished.
The HWC data size is 32x32 or 64x64, determined by hwc_size. The data in external memory should be 32-bit aligned or 64-bit aligned, and stored in VOP internal memory in 64-bit aligned as follows.

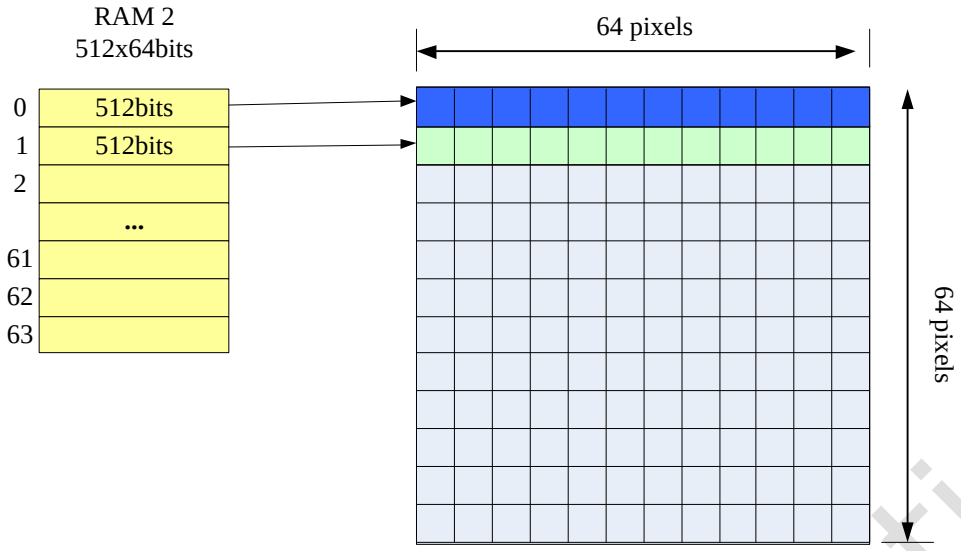


Fig. 18-21 hwc data format

1.7.3 IEP direct path

There are two data source for win0/win1: external memory and IEP internal memory. However, the IEP data is just active for one layer at one frame time when IEP data path(SYS_CTRL[11]) is enable, determined by direct_path_layer_sel (SYS_CTRL[12]) signal. Moreover, it is suitable for win0with scaling, but there is some limitation when picture scale down(because of there is not enough fifo to store data for scaling down) more than 4 times(horizontal multiple by vertical).
Direct path interface (DPI) can be used for VOP to display images from other image processing IPs, which also has DPI (slave).
There are programming flows for both DPI display on sequence and DPI display off sequence.

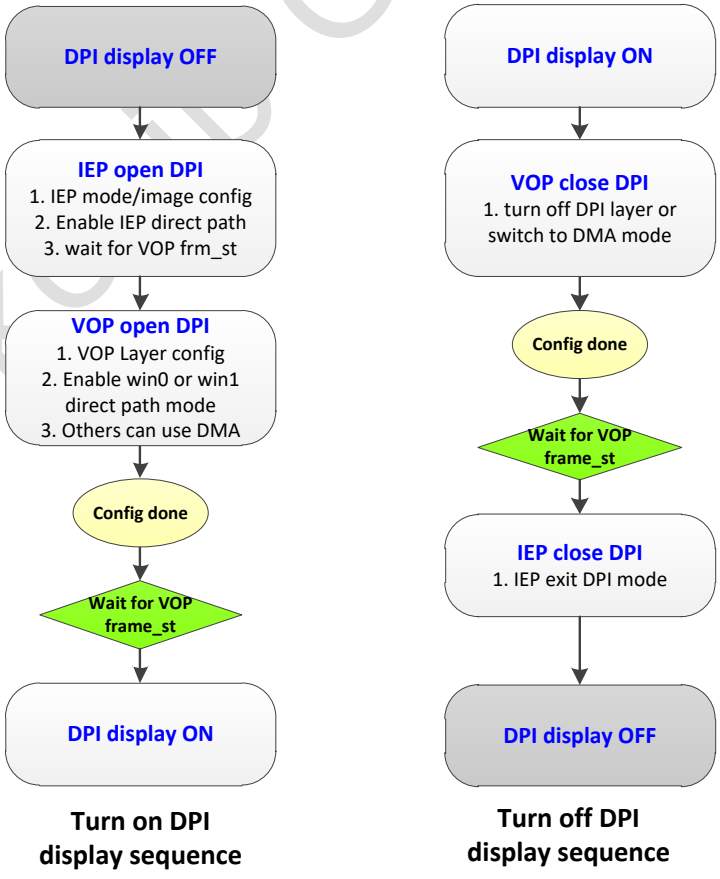


Fig. 18- 22 VOP DPI Programming Flow

1.Turn on DPI display
First, configure IEP into DPI mode. After doing image information and image processing mode configuration, IEP DPI mode can be turn on for display. IEP is in idle mode only if VOP's frame start input signal is valid.
Second, configure VOP for DPI display. Note that only one layer (Win0 or Win1) can use DPI in same frame. Other layers still can use internal DMA.
Finally, set VOP "config_done" to confirm all the new configuration and waiting frame sync to start DPI display actually.
2.Turn off DPI display
First, close VOP layer's DPI mode by turning off DPI layer or switching it to DMA mode. Then set VOP "config_done" to confirm new configuration.
Second, wait for VOP's frame synchronization to close DPI display in VOP.
Finally, turn off IEP's DPI mode.

1.7.4 GAMMA LUT
When dsp_lut_en is 0, the DSP LUT data should be refreshed by software. i.e, writing dsplut data to the internal memory with the start address DSP_LUT_MST. The memory size is 256x24, i.e, lower 24bits valid, and the writing data number is determined by software.

1.7.5 DMA control（QoS/Hurry/Outstanding）
If you want to get higher priority for VOP to access external memory when the frame data is urgent, a QoS and hurry request can be generated and sent out basing on the configured values:
sw_noc_qos_en: AXI_BUS_CTRL[0]
sw_noc_qos_value: AXI_BUS_CTRL[2:1]
sw_noc_hurry_en: AXI_BUS_CTRL[3]
sw_noc_hurry_value: AXI_BUS_CTRL[5:4]
sw_noc_hurry_threshold: AXI_BUS_CTRL[9:6]
sw_axi_max_outstand_en: AXI_BUS_CTRL[11]
sw_axi_max_outstand_num: AXI_BUS_CTRL[16:12]
QoS request for higher bus priority for win1
NOC hurry for higher bus prioirity for VIO when win0 needs higher priority.
Max Outstanding num is configurable,but limited at 31 when mmu enables or 32 when mmu disables.

1.7.6 Interrupt
VOP interrupt is comprised of 9 interrupt sources:
- frame start interrupt
- line flag interrupt
- bus error interrupt
- win0 empty interrupt
- win1 empty interrupt
- scaler empty interrupt
- irq_mmu
- irq_tve
- dsp hold intrrupt

Every interrupt has independent interrupt enable signal(VOP_INT_EN),interrupt clear signal(VOP_INT_CLR), interrupt status signal (VOP_INT_STATUS).
MMU's and TVE's interrupt enable and clear signal are set in their own group, and are combined with others in top module. The last interrupt to outside is just a combinated signal and high active.
There are 2 raw interruption:
- Line flag raw Interrupt status (INT_STATUS[31])
- Frame start raw interrupt status(INT_STATUS[30])

1.7.7 RGB display mode
RGB display mode is used for RGBpanel display. It is a continuous frames display mode.

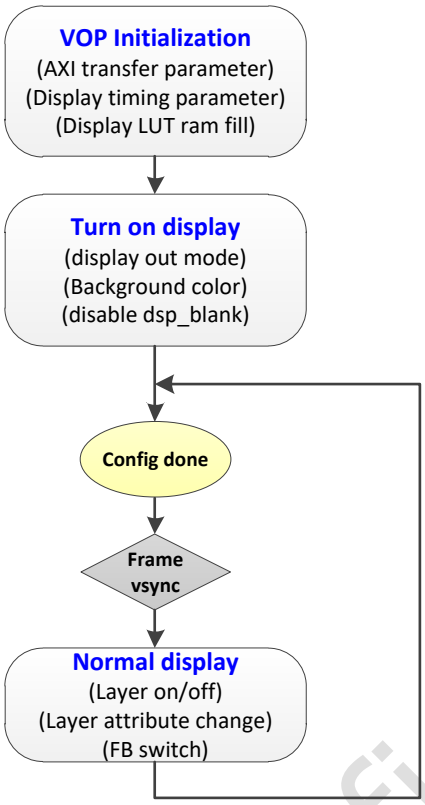


Fig. 18- 23 VOP RGB Mode Programming Flow

1.VOP initialization

VOP initialization should be done before turning display on.

First, AXI bus parameter (VOP_AXI_BUS_CTRL) should be set for DMA transfer.

Second, display panel/interface timing should be set for display output. The registers are: VOP_DSP_HTOTAL_HS_END/ VOP_DSP_HACT_ST_END/ VOP_DSP_VTOTAL_HS_END/ VOP_DSP_VACT_ST_END/ VOP_DSP_VS_ST_END_F1/ VOP_DSP_VACT_ST_END_F1

2.Background display

Before normal display, the background display could be turn on.

First, set display output mode (VOP_DSP_CTRL0/1) according to display device.

Second, disable dsp_blank mode, which would not be enable until frame synchronization.

Finally, writing '1' to "VOP_REG_CFG_DONE" register then all the frame-sync registers will be enable at the beginning of next frame.

3.Normal display

In normal display, all the display layers' attribute could be different according display scenario. So there is a programming loop in this mode.

First, configure all the display layers' attribute registers for the change of image format, location, size, scaling factor, alpha and overlay and so on. Those register would not be enable until frame synchronization.

Finally, write 1 to "VOP_REG_CFG_DONE" register then all the frame-sync registers will be enable at the beginning of next frame.

1.7.8 Immediately control register

There are two type registers in VOP , one is effective immediately,the other is effective by frame sync.Effective immediately registers list as follows,other registers are all effective by frame sync.

Table 18-6 effective immediately register table

register address	description
0x000	some ctrl function bits
0x004	some dsp ctrl function bits
0x008	background color register dsp control function bits

0x014	Alpha control register
0x018	win0 color key register
0x01c	win1 color key register
0x06c~0x080	dsp_timing register
0xe0~0xf4	Frc configuration bits

1.7.9 Output Polarity Control

There are five channels output ,list as follow:
Tve_dac_dclk_en ,tve_dac_dclk_inv;
HDMI_dclk_en,HDMI_dclk_inv;
Rgb_dclk_en,rgb_dclk_inv;
Lvds_dclk_en,lvds_dclk_inv;
Mipi_dclk_en,mipi_dclk_inv.

The xxx_dclk_en should be set to 1 when output select the xxx channel,and the other channel's xxx_dclk_en should be set to 0 to gate the output clk and data.
The rgb, lvds, mipi channel are mutual exclusion . They share one pair of output polarity control registers(DSP_CTRL0[6:4]).
HDMI channel polarity control registers maps to INT_SCALER[6:4].
When using RGB panel,the dclk should be tied to "0" or "1" in some scenarios.
In this case ,you should enable sw_io_pad_clk_sel ,to tie dclk to "0". IF enable rgb_dclk_inv at the same time,the dclk will tie to "1".

1.7.10 Scaler

Program sequence

(1)set PLL and CPLL as slow mode;
set CRU_MODE_CON = 0xff000000;
(2)config dclk and sclk
You can use CPLL or GPLL for sclk or dclk.
sclk needs PLL fraction mode to get precise clock.
sclk has been figured out as the fllowing table.
(3)set CPLL/GPLL as normal mode;
(4)set VOP including scaler configuration, refer to the following table.
Assert scaler_en after all scaler configuration registers are set.

Table 18-7 VOP Scaler configuration

HDMI				PAD			SCALER configuration				
TYPE	**dclk**	**width**	**height**	**sclk**	**width**	**height**	**in_vst**	**dsp_vst**	**dsp_hst**	**h_factor**	**v_factor**
1080P @60Hz	148.5	1920	1080	70.400002	1280	800	4	32	1059	0x1801	0x159b
				57.599998	1024	768	4	17	1946	0x1e03	0x1682
				50.400002	1024	600	4	4	1040	0x1e03	0x1cd2
				26.73	800	480	9	23	1586	0x266d	0x2402
				39.599998	800	600	4	4	1339	0x266d	0x1cd2

				39.599998	800	600	4	28	19	0x266d	0x1cd2
1080P @50Hz	148.5	1920	1080	58.666668	1280	800	1	30	1233	0x1801	0x159b
				48	1024	768	1	15	2297	0x1e03	0x1682
				42	1024	600	1	2	1210	0x1e03	0x1cd2
				22.275	800	480	9	23	1865	0x266d	0x2402
				33	800	600	4	4	1568	0x266d	0x1cd2
				33	800	600	4	27	2624	0x266d	0x1cd2
720P @60Hz	74.25	1280	720	70.400002	1280	800	1	19	143	0x1000	0xe65
				57.999998	1024	768	1	9	587	0x1401	0xeff
				50.400002	1024	600	1	0	684	0x1401	0x1334
				26.73	800	480	1	8	957	0x199c	0x1804
				39.599998	800	600	4	2	833	0x199c	0x1334
				39.599998	800	600	4	18	173	0x199c	0x1334
720P @50Hz	74.25	1280	720	58.666668	1280	800	1	19	120	0x1000	0xe65
				48	1024	768	1	9	652	0x1401	0xeff
				42	1024	600	1	0	769	0x1401	0x1334
				22.275	800	480	1	8	1096	0x199c	0x1804
				33	800	600	4	2	948	0x199c	0x1334

				33	800	600	4	18	156	0x199c	0x1334
576P @50Hz	27	720	480	61.111111	1280	800	1	39	774	0x8fc	0xb83
				50	1024	768	1	32	96	0xb3e	0xbfe
				43.75	1024	600	1	24	828	0xb3e	0xf5b
				23.203123	800	480	1	31	424	0xe68	0x1334
				34.375	800	600	4	27	26	0xe68	0xf5b
				34.375	800	600	4	39	441	0xe68	0xf5b
480P @60Hz	27	720	480	73.846153	1280	800	1	33	81	0x8fc	0x997
				60.419579	1024	768	1	25	521	0xb3e	0x9fd
				52.867134	1024	600	1	20	555	0xb3e	0xccb
				28.038462	800	480	1	26	77	0xe68	0x1000
				73.846153	800	600	1	19	606	0xe68	0xccb
				60.419579	800	600	1	30	92	0xe68	0xccb

*in_vst:scl_in_vsync_vst;
dsp_vst:scl_dsp_frame_vst;
dsp_hst:scl_dsp_frame_hst;

sclk Requirement

Sclk(the output pixel clock) is asynchronous with input pixel clock to support different display resolution. But the input/output frame frequency should be same (FSin=FSout)because there is no frame buffer in SCALER. So there is strict clock requirement for output pixel clock, which is generated from CPLL or GPLL.

Assume that Fout is the output pixel clock frequency and Fin is the input pixel clock frequency.

dsp_in_vtotal*dsp_in_htotal*Fin=dsp_vtotal*dsp_in_htotal*Fout;

The total output active frame period (include horizontal blank period, but without vertical blank period) must be close to the total input active frame period. The difference between them should be less than Xmax input active line period(include horizontal blank period).

Output frame scanning start point setting

To meet the requirement of active pixel start point, there are dsp_frm_hst/ dsp_frm_vst settings for frame scanning start point adjustment.
The timing delay of output active frame pixel start point behind input active frame pixel start point can be calculated as following equation.
Usually,we set Delta_T as follow:
ifv_scale_ratio<2,Delta_T=4* dsp_in_htotal*1/Fin; or Delta_T=5* dsp_in_htotal*1/Fin;
ifv_scale_ratio>=2,Delta_T=12* dsp_in_htotal*1/Fin;

T_frm_st = (T_BP_in + Delta_T - T_BP_out);
In where,
T_frm_st = (dsp_frm_vst*dsp_in_htotal+dsp_frm_hst)/Fin;
T_BP_in is the blank period of input frame;
T_BP_out is the blank period of output frame and.

if(T_frm_st<0),Delta_T should be adjusted as below:
T_frm_st=Tin –T_frm_st;
then we can calc out dsp_frm_vst and dsp_frm_hst as below:
dsp_frm_vst = floor((T_frm_st*Fin)/dsp_in_htotal);
dsp_frm_hst = (T_frm_st–dsp_frm_vst*dsp_in_htotal/Fin)*Fin;

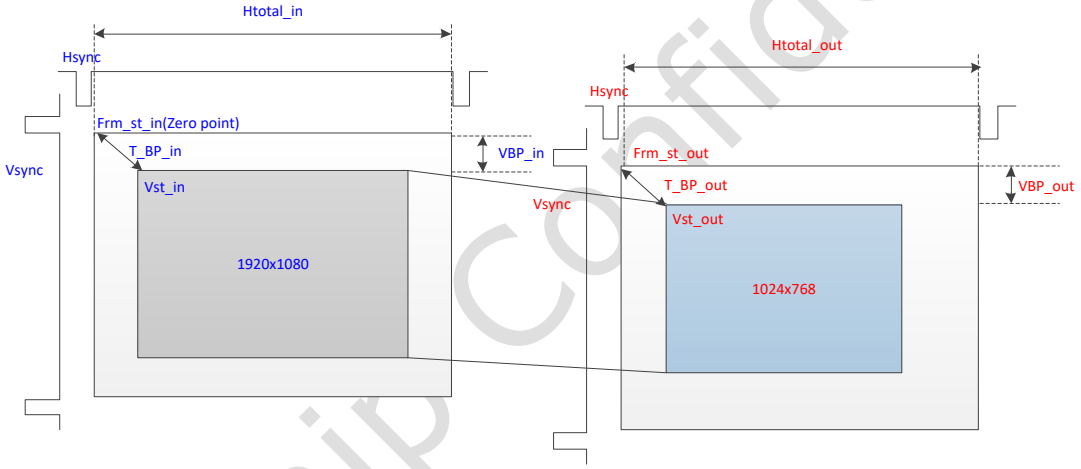


Fig. 18-24 Scaler frame scanning start point

Chapter 2 VPU_Combo

2.1 Overview

VPU_Combo is composed by the H.265 (HEVC) decoder and the H.264 encoder/decoder to realize the high quality video decoding. VPU_Combo is connected to the AHB bus through an AHB slave and the AXI bus through an AXI master. The register configuration is fed into the VPU_Combo through the AHB slave interface while the stream data is transacted between DDR and VPU_Combo through the AXI master interface.
To reduce the area, H.265 decoder and H.264 encoder/decoder not only share several pieces of the internal memories, but also share the bus master and slave interfaces. Therefore VPU_Combo has no any possibility to have the H.265 video decoding and H.264 video encoding/decoding to work simultaneously
In order to improve large data transaction performance, VPU embeds MMU (memory management unit) and supports the cacheable bus operation.
VPU_Combo supports the next-generation video coding standard HEVC (High Efficiency Video Coding, aka H.265) full-HD decoding up to 60fps. With HEVC standard, the data compression ratio can be doubled compared to H.264/MEPG-4 at the same video quality or alternatively to provide substantially improved video at the same bit rate.

2.1.1 Features

- Supports HEVC Main Profile up to Level 4.1 High Tier: 1920x1080@60 fps
  - MMU embedded
  - Supports frame timeout interrupt , frame finish interrupt and bitstream error interrupt
  - Supports RLC write mode, RLC mode and Normal Mode
- Supports decoding of the following standards
  - Error detection and concealment support for all video formats
  - Output data structure after decoder is YCbCr 4:2:0 semi-planar to have more efficient bus usage
  - H.264 up to HP level 4.1 : 1080p@60fps (1920x1088)
  - SVC: base layer only for Baseline/High Profile
  - MVC: Stereo High
  - MPEG-4: Simple Profile up to Level 6; Advanced Profile up to Level 5
  - MPEG-2: Main Profile up to High Level
  - MPEG-1: Main Profile up to High Level
  - H.263: Level 10-70 for Profile 0, and image size up to 720x576
  - JPEG: Baseline interleaved, and supports ROI (region of image) decode
  - RV: RV8, RV9, V10
  - VP7: up to version 3
  - VP8: version 2(webM)
  - WebP
  - For H.264, Image cropping not supported
- Built-in post processor in H.264 decoder supports:
  - Stand-alone mode: rotation, deinterlace, RGB conversion, scaling, dithering and alpha blending
  - Pipe-ling mode:, RGB conversion, scaling, dithering and alpha blending
- Supports encoding of the following standards:
  - H.264: up to HP level 4.1
  - JPEG: Baseline (DCT sequential)
- Built-in pre-processor in H.264 encoder supports:
  - Cropping, rotation, YCbCr conversion

2.2 Block Diagram

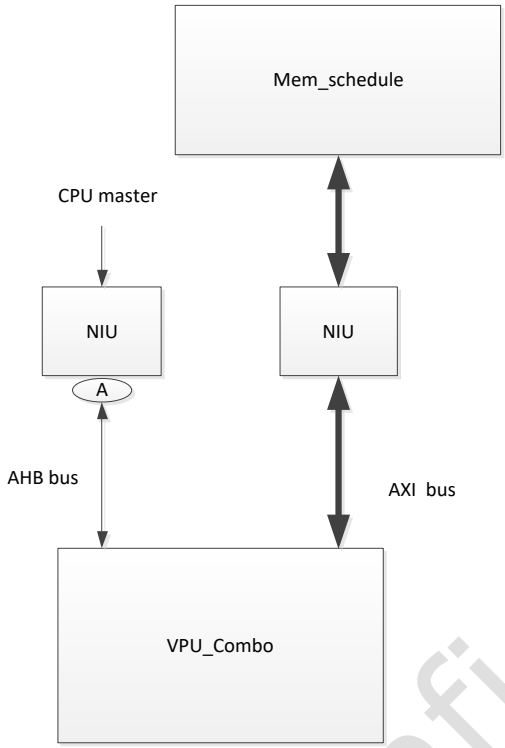


Fig. 2-1 VPU Combo in SOC

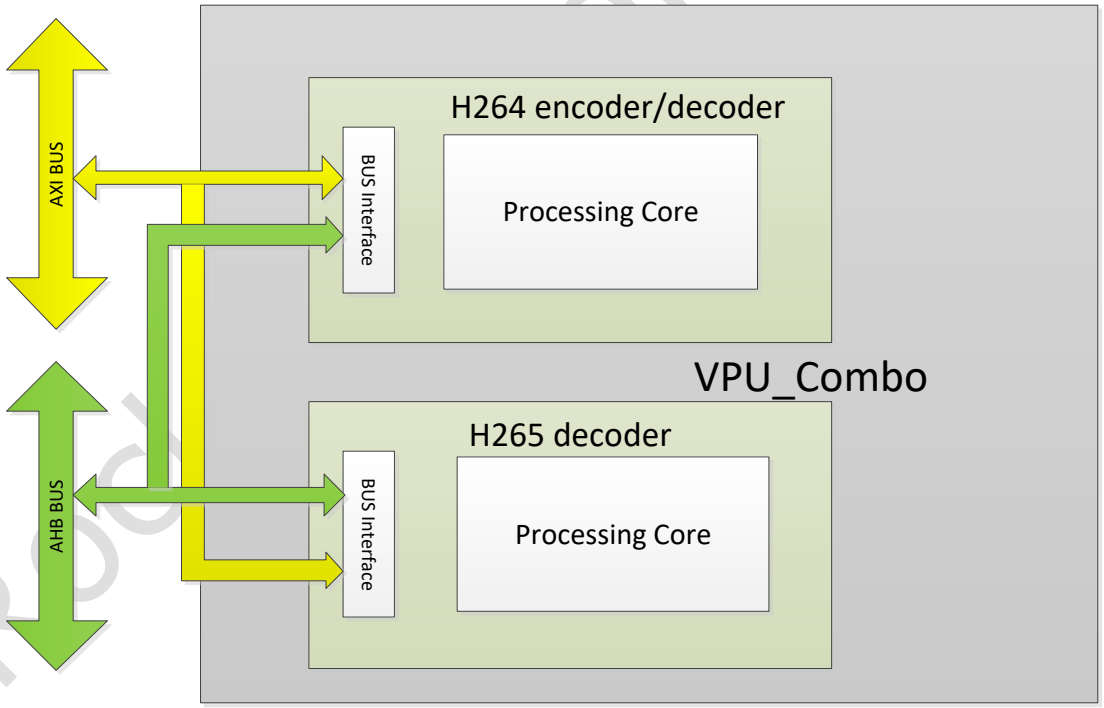


Fig. 2-2 VPU Combo Block Diagram

As shown in the figures above, CPU accesses to HEVC register bank through 32-bit AHB bus. Bitstream and compressed video data are fed into HEVC core though 128-bit AXI read channel, and after several steps of decoding process, decoded pictures are transferred to designated location in the DDR through 64-bit AXI write channel.

CPU accesses to H264 encoder/decoder register bank through 32-bit AHB bus. Video data are fed into H.264 core though 64-bit AXI read channel, and after several steps of decoding

process, process results are transferred to designated location in the DDR through 64-bit AXI write channel.

2.3 Function Description

2.3.1 HEVC Standard

High Efficiency Video Coding (HEVC) is a video compression standard, a successor to H.264/MPEG-4 AVC (Advanced Video Coding), that was jointly developed by the ISO/IEC Moving Picture Experts Group (MPEG) and ITU-T Video Coding Experts Group (VCEG) as ISO/IEC 23008-2 MPEG-H Part 2 and ITU-T H.265. MPEG and VCEG established a Joint Collaborative Team on Video Coding (JCT-VC) to develop the HEVC standard.
HEVC was designed to substantially improve coding efficiency compared to H.264/MPEG-4 AVC HP, i.e. to reduce bitrate requirements by half with comparable image quality, at the expense of increased computational complexity. HEVC was designed with the goal of allowing video content to have a data compression ratio of up to 1000:1. Depending on the application requirements HEVC encoders can trade off computational complexity, compression rate, robustness to errors, and encoding delay time.Two of the key features where HEVC was improved compared to H.264/MPEG-4 AVC was support for higher resolution video and improved parallel processing methods.

2.3.2 HEVC Coding Tools

1. Coding tree unit
HEVC replaces macroblocks, which were used with previous standards, with Coding Tree Units (CTUs) which can use a larger block structures of up to 64x64 pixels and can better sub-partition the picture into variable sized structures. HEVC initially divides the picture into CTUs which can be 64x64, 32x32, or 16x16 with a larger pixel block size usually increasing the coding efficiency.

2. Parallel processing tools

Tiles allow for the picture to be divided up into a grid of rectangular regions that can independently be decoded/encoded and the main purpose of tiles is to allow for parallel processing.Tiles can be independently decoded and can even allow for random access to specific regions of a picture in a video stream.

Wavefront parallel processing (WPP) is when a slice is divided into rows of CTUs in which the first row is decoded normally but each additional row requires that decisions be made in the previous row. WPP has the entropy encoder use information from the preceding row of CTUs and allows for a method of parallel processing that may allow for better compression than tiles.

Tiles and WPP are allowed but are optional.If tiles are present they must be at least 64 pixels high and 256 pixels wide with a level specific limit on the number of tiles allowed.

Slices can for the most part be decoded independently from each other with the main purpose of tiles being re-synchronization in case of data loss in the video stream. Slices can be defined as self-contained in that prediction is not made across slice boundaries. When in-loop filtering is done on a picture though information across slice boundaries may be required.[1] Slices are CTUs decoded in the order of the raster scan and different coding types can be used for slices such as I types, P types, or B types.

Dependent slices can allow for data related to tiles or WPP to be accessed more quickly by the system than if the entire slice had to be decoded.[1] The main purpose of dependent slices is to allow for low delay video encoding due to its lower latency.

3. Entropy coding
HEVC uses a context-adaptive binary arithmetic coding (CABAC) algorithm that is fundamentally similar to CABAC in H.264/MPEG-4 AVC. CABAC is the only entropy encoder method that is allowed in HEVC while there are two entropy encoder methods allowed by H.264/MPEG-4 AVC. CABAC and the entropy coding of transform coefficients in HEVC were designed for a higher throughput than H.264/MPEG-4 AVC. For instance, the number of context coded bins have been reduced by 8x and the CABAC bypass-mode has been improved in terms of its design to increase throughput. Another improvement with HEVC is that the dependencies between the coded data has been changed to further increase

throughput. Context modeling in HEVC has also been improved so that CABAC can better select a context that increases efficiency when compared to H.264/MPEG-4 AVC.

4. Intra prediction
HEVC specifies 33 directional modes for intra prediction compared to the 8 directional modes for intra prediction specified by H.264/MPEG-4 AVC. HEVC also specifies planar and DC intra prediction modes.[1] The intra prediction modes use data from neighboring prediction blocks that have been previously decoded.

5. Motion compensation
For the interpolation of fractional luma sample positions HEVC uses separable application of one-dimensional half-sample interpolation with an 8-tap filter or quarter-sample interpolation with a 7-tap filter while, in comparison, H.264/MPEG-4 AVC uses a two-stage process that first derives values at half-sample positions using separable one-dimensional 6-tap interpolation followed by integer rounding and then applies linear interpolation between values at nearby half-sample positions to generate values at quarter-sample positions.[1] HEVC has improved precision due to the longer interpolation filter and the elimination of the intermediate rounding error. For 4:2:0 video, the chroma samples are interpolated with separable one-dimensional 4-tap filtering to generate eighth-sample precision, while in comparison H.264/MPEG-4 AVC uses only a 2-tap bilinear filter (also with eighth-sample precision).

As in H.264/MPEG-4 AVC, weighted prediction in HEVC can be used either with uni-prediction (in which a single prediction value is used) or bi-prediction (in which the prediction values from two prediction blocks are combined).

6. Motion vector prediction
HEVC defines a signed 16-bit range for both horizontal and vertical motion vectors (MVs). This was added to HEVC at the July 2012 HEVC meeting with the mvLX variables. HEVC horizontal/vertical MVs have a range of −32768 to 32767 which given the quarter pixel precision used by HEVC allows for a MV range of −8192 to 8191.75 luma samples. This compares to H.264/MPEG-4 AVC which allows for a horizontal MV range of −2048 to 2047.75 luma samples and a vertical MV range of −512 to 511.75 luma samples.

HEVC allows for two MV modes which are Advanced Motion Vector Prediction (AMVP) and merge mode. AMVP uses data from the reference picture and can also use data from adjacent prediction blocks. The merge mode allows for the MVs to be inherited from neighboring prediction blocks. Merge mode in HEVC is similar to "skipped" and "direct" motion inference modes in H.264/MPEG-4 AVC but with two improvements. The first improvement is that HEVC uses index information to select one of several available candidates. The second improvement is that HEVC uses information from the reference picture list and reference picture index.

7. Inverse transforms
HEVC specifies four transform units (TUs) sizes of 4x4, 8x8, 16x16, and 32x32 to code the prediction residual. A CTB may be recursively partitioned into 4 or more TUs.[1] TUs use integer basis functions that are similar to the discrete cosine transform (DCT). In addition 4x4 luma transform blocks that belong to an intra coded region are transformed using an integer transform that is derived from discrete sine transform (DST). This provides a 1% bit rate reduction but was restricted to 4x4 luma transform blocks due to marginal benefits for the other transform cases. Chroma uses the same TU sizes as luma so there is no 2x2 transform for chroma.

8. Loop filters
HEVC specifies two loop filters that are applied sequentially, with the deblocking filter (DBF) applied first and the sample adaptive offset (SAO) filter applied afterwards. Both loop filters are applied in the inter-picture prediction loop, i.e. the filtered image is stored in the

decoded picture buffer (DPB) as a reference for inter-picture prediction.

8.1 Deblocking filter
The DBF is similar to the one used by H.264/MPEG-4 AVC but with a simpler design and better support for parallel processing.[1] In HEVC the DBF only applies to a 8x8 sample grid while with H.264/MPEG-4 AVC the DBF applies to a 4x4 sample grid. DBF uses a 8x8 sample grid since it causes no noticeable degradation and significantly improves parallel processing because the DBF no longer causes cascading interactions with other operations. Another change is that HEVC only allows for three DBF strengths of 0 to 2. HEVC also requires that the DBF first apply horizontal filtering for vertical edges to the picture and only after that does it apply vertical filtering for horizontal edges to the picture. This allows for multiple parallel threads to be used for the DBF.

8.2 Sample adaptive offset
The SAO filter is applied after the DBF and is designed to allow for better reconstruction of the original signal amplitudes by applying offsets stored in a lookup table in the bitstream. Per CTB the SAO filter can be disabled or applied in one of two modes: edge offset mode or band offset mode. The edge offset mode operates by comparing the value of a sample to two of its eight neighbors using one of four directional gradient patterns. Based on a comparison with these two neighbors, the sample is classified into one of five categories: minimum, maximum, an edge with the sample having the lower value, an edge with the sample having the higher value, or monotonic. For each of the first four categories an offset is applied. The band offset mode applies an offset based on the amplitude of a single sample. A sample is categorized by its amplitude into one of 32 bands (histogram bins). Offsets are specified for four consecutive of the 32 bands, because in flat areas which are prone to banding artifacts, sample amplitudes tend to be clustered in a small range.[1][135] The SAO filter was designed to increase picture quality, reduce banding artifacts, and reduce ringing artifacts.

2.3.3 MMU

The MMU divides memory into 4KB pages, where each page can be individually configured. For each page the following parameters are specified:

- Address translation of virtual memory, this enables the processor to work using address that differ from the physical address in the memory system.
- The permitted types of accesses to that page. Each page can permit read, write, both, or none.

The MMU use 2-level page table structure:
1. The first level, the page directory consists of 1024 directory table entries(DTEs), each pointing to a page table.
2. The second level, the page table consists of 1024 page table entries(PTEs), each pointing to a page in memory.

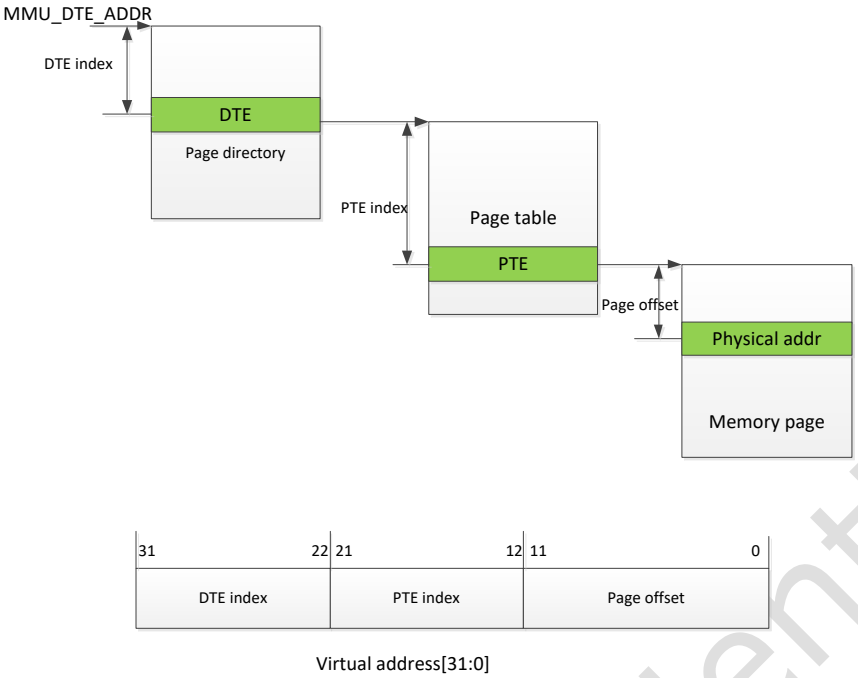


Fig. 2-3 structure of two-level page table

2.3.4 HEVC Working Mode

There are there working modes to be selected for HEVC decoder: RLC Mode, RLC Write Mode, Normal Mode.

The key differences among three working modes are whether CABAC module and Post-CABAC module are involved into the hardware decoding process.

For RLC mode, CABAC are bypassed and the input bitstream to the Post-CABAC module should be already decoded.

For RLC write mode, the decoded results by CABAC are output to the DDR, and the following decoding processes are stopped.

As for the normalmode , all the modules are involved into the decoding process, and complete decoding results are output. Normally, this mode should be selected.

2.3.5 H.264 decoder

The input of the decoder is H.264 standard bit stream in either plain NAL unit format or byte stream format. The input format in use will be automatically detected. The H.264 video encoding allows the use of multiple reference pictures, which means that the decoding order of the pictures may be different from their display order. The decoder can perform internally the display reordering of the decoded pictures or it can skip this and output all the pictures as soon as they are decoded.

The decoder has two operating modes: in the primary mode the HW performs entropy decoding, and in the secondary mode SW performs entropy decoding. Secondary mode is used in H.264 ASO or Slice Group stream decoding.

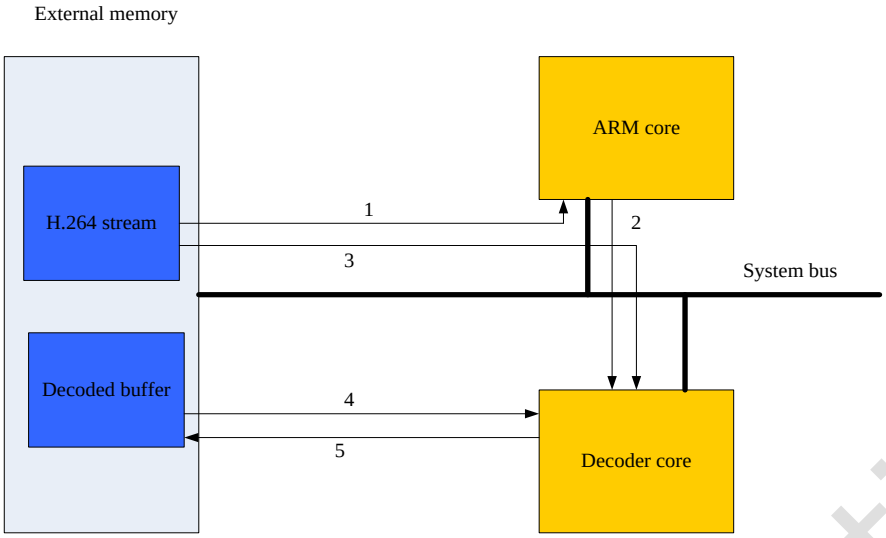


Fig. 2-4 Dataflow of HW performs entropy decoding in video decoder

The dataflow of HW performs entropy decoding is as Fig. 2-4 shown. The decoder software (SW) starts decoding the first picture by parsing the stream headers (1). Software then setups the hardware control registers (picture size, stream start address etc.) and enables the hardware (2).Hardware decodes the picture by reading stream (3) and the reference pictures (required for inter picture decoding)(4) from the external memory. Hardware writes the decoded output picture to memory one macroblock at a time (5). When the picture has been fully decodedor the hardware has run out of stream data, it gives an interrupt with a proper status flag and provides stream and address for software to continue and returns to initial state.

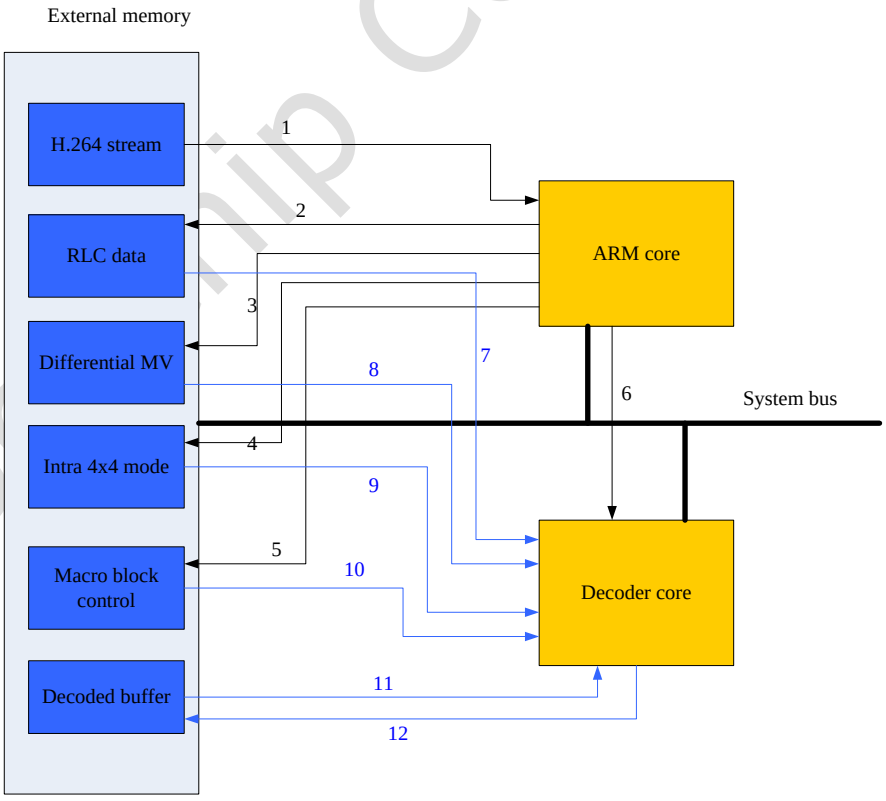


Fig. 2-5 Dataflow of SW performs entropy decoding in video decoder

SW entropy decoding mode (RLC mode) changes the input data format that is transferred

from SW to HW. The dataflow of this mode is asFig. 2-5.In this case:
the decoder software starts decoding the first picture by parsing the stream headers (1) and by performing entropy decoding. Software then writes the following items to external memory:Run-length-code (RLC) data (2)
Differential motion vectors (3)
Intra 4x4 prediction modes (4)
Macroblock control data (5)
Last step for the software is to write the hardware control registers and to enable the hardware (6).
Hardware decodes the picture by buffering control data for several macroblocks at a time, and reading then appropriate amount of RLC data, differential motion vectors and intra modes depending on each macroblock type (7)-(10).For the rest of the decoding process (11)-(12), the functionality is identical to the HW entropy decoding mode.When the picture has been fully decoded, hardware can raise an interrupt and write the status bits in the status register.

2.4 Register description

This section describes the control/status registers of the VPU_Combo.
If HEVC is chosen to work, HEVC register base address is the base address of the HEVC. The HEVC reading MMU master register base address is hevc_base+0x440, the writing MMU register base address is hevc_base + 0x480, and the cache control register base address is hevc_base + 0x400.
If H.264 decoder/encoder is chosen to work, VEPU(encoder) register base address is vpu_base, and VDPU(decoder) base address is vpu_base + 0x400. MMU base address is vpu_base+0x800, and VDPU cache control base address is vpu_base + 0xc00.

2.4.1 VDP_HEVC Register Summary

Name	Offset	Size	Reset Value	Description
hevc_swreg0_id	0x0000	W	0x68761100	ID register (read only)
hevc_swreg1_int	0x0004	W	0x00200022	interrupt and decoder enable register
hevc_swreg2_sysctrl	0x0008	W	0x00000000	Data input and output endian setting and sys ctrl
hevc_swreg3_picpar	0x000c	W	0x00000000	picture parameters
hevc_swreg4_strm_rlc_base	0x0010	W	0x00000000	the stream or rlc data base address
hevc_swreg5_stream_rlc_len	0x0014	W	0x00000000	amount of stream bytes or rlc data byte in the input buffer or the
hevc_swreg6_cabactbl_base	0x0018	W	0x00000000	the base address of cabac table
hevc_swreg7_decout_base	0x001c	W	0x00000000	base address of decoder output picture base address
hevc_swreg8_y_virstride	0x0020	W	0x00000000	the ouput picture y virtual stride
hevc_swreg9_yuv_virstride	0x0024	W	0x00000000	the ouput picture yuv virtual stride
hevc_swreg10_refer0_base	0x0028	W	0x00000000	base address for reference picture index 0
hevc_swreg11_refer1_base	0x002c	W	0x00000000	base address for reference picture index 1

Name	Offset	Size	Reset Value	Description
hevc_swreg12_refer2_base	0x0030	W	0x00000000	base address for reference picture index 2
hevc_swreg13_refer3_base	0x0034	W	0x00000000	base address for reference picture index 3
hevc_swreg14_refer4_base	0x0038	W	0x00000000	base address for reference picture index 4
hevc_swreg15_refer5_base	0x003c	W	0x00000000	base address for reference picture index 5
hevc_swreg16_refer6_base	0x0040	W	0x00000000	base address for reference picture index 6
hevc_swreg17_refer7_base	0x0044	W	0x00000000	base address for reference picture index 7
hevc_swreg18_refer8_base	0x0048	W	0x00000000	base address for reference picture index 8
hevc_swreg19_refer9_base	0x004c	W	0x00000000	base address for reference picture index 9
hevc_swreg20_refer10_base	0x0050	W	0x00000000	base address for reference picture index 10
hevc_swreg21_refer11_base	0x0054	W	0x00000000	base address for reference picture index 11
hevc_swreg22_refer12_base	0x0058	W	0x00000000	base address for reference picture index 12
hevc_swreg23_refer13_base	0x005c	W	0x00000000	base address for reference picture index 13
hevc_swreg24_refer14_base	0x0060	W	0x00000000	base address for reference picture index 14
hevc_swreg25_refer0_poc	0x0064	W	0x00000000	the poc of reference picture index 0
hevc_swreg26_refer1_poc	0x0068	W	0x00000000	the poc of reference picture index 1
hevc_swreg27_refer2_poc	0x006c	W	0x00000000	the poc of reference picture index 2
hevc_swreg28_refer3_poc	0x0070	W	0x00000000	the poc of reference picture index 3
hevc_swreg29_refer4_poc	0x0074	W	0x00000000	the poc of reference picture index 4
hevc_swreg30_refer5_poc	0x0078	W	0x00000000	the poc of reference picture index 5
hevc_swreg31_refer6_poc	0x007c	W	0x00000000	the poc of reference picture index 6
hevc_swreg32_refer7_poc	0x0080	W	0x00000000	the poc of reference picture index 7
hevc_swreg33_refer8_poc	0x0084	W	0x00000000	the poc of reference picture index 8

Name	Offset	Size	Reset Value	Description
hevc_swreg34_refer9_poc	0x0088	W	0x00000000	the poc of reference picture index 9
hevc_swreg35_refer10_poc	0x008c	W	0x00000000	the poc of reference picture index 10
hevc_swreg36_refer11_poc	0x0090	W	0x00000000	the poc of reference picture index 11
hevc_swreg37_refer12_poc	0x0094	W	0x00000000	the poc of reference picture index 12
hevc_swreg38_refer13_poc	0x0098	W	0x00000000	the poc of reference picture index 13
hevc_swreg39_refer14_poc	0x009c	W	0x00000000	the poc of reference picture index 14
hevc_swreg40_cur_poc	0x00a0	W	0x00000000	the poc of cur picture
hevc_swreg41_rlcwrite_base	0x00a4	W	0x00000000	the base address or rlcwrite base addr
hevc_swreg42_pps_base	0x00a8	W	0x00000000	the base address of pps
hevc_swreg43_rps_base	0x00ac	W	0x00000000	the base address of rps
hevc_swreg44_cabac_error_en	0x00b0	W	0x00000000	cabac error enable config
hevc_swreg45_cabac_error_status	0x00b4	W	0x00000000	cabac error status
hevc_swreg46_cabac_error_ctu	0x00b8	W	0x00400000	cabac error ctu
hevc_swreg47_sao_ctu_position	0x00bc	W	0x00000000	sao ctu position
hevc_swreg64_performance_cycle	0x0100	W	0x00000000	hevc performance cycle
hevc_swreg65_axi_ddr_rdata	0x0104	W	0x00000000	axi ddr read data num
hevc_swreg66_axi_ddr_wdata	0x0108	W	0x00000000	axi ddr write data number

*Notes:Size:**B**- Byte (8 bits) access, **HW**- Half WORD (16 bits) access, **W**-WORD (32 bits) access*

2.4.2 VDP_HEVC Detail Register Description

hevc_swreg0_id

Address: Operational Base + offset (0x0000)

ID register (read only)

Bit	Attr	Reset Value	Description
31:16	RO	0x6876	product number The ascii code of "hv", which is 0x6876
15	RO	0x0	reserved
14	RW	0x0	codec flag 0: only dec 1: dec + enc

Bit	Attr	Reset Value	Description
13	RO	0x0	reserved
12	RW	0x1	hevc profile 0: Main 1: Main10
11:9	RO	0x0	reserved
8	RO	0x1	level 0: FHD 1: UHD
7:0	RO	0x00	minor version

hevc_swreg1_int
Address: Operational Base + offset (0x0004)
interrupt and decoder enable register

Bit	Attr	Reset Value	Description
31:23	RO	0x0	reserved
22	RW	0x0	sw_softreset_rdy when it is 1'b1, it says that softreset has been done
21	RW	0x1	sw_force_softreset_valid when sw_force_softreset_valid is 1'b1, sw_softrst_en will always be valid to the system no matter that whether the axi bus is idle; when sw_force_softreset_valid is 1'b0, sw_softrst_en will only be valid when the axi bus is idle.
20	RW	0x0	sw_softrst_en_p softreset enable signal write 1 to soft reset, write 0 invalid puls register
19	RO	0x0	reserved
18	RW	0x0	sw_cabu_end_sta cabac decode end status
17	RW	0x0	sw_colmv_ref_error_sta colmv ref error status when it is 1'b1, it means that inter module read the invalid dpb frame
16	RW	0x0	sw_buf_empty_sta buffer empty status, only when sw_buf_empty_en is 1'b1 , this bit is valid
15	RW	0x0	sw_dec_timeout_sta decoder timeout interrupt status When high the decoder has been idling for too long. it will self reset the hardware only when sw_dec_timeout_e is 1'b1, this bit is valid
14	RW	0x0	sw_dec_error_sta status bit of input stream error when high, an error is found in input data stream decoding. It will self reset the hardware

Bit	Attr	Reset Value	Description
13	RW	0x0	sw_dec_bus_sta bus error status When this bit is high, there is error on the axi bus, it will self reset hardware
12	RW	0x0	sw_dec_rdy_sta decoder ready status when this bit is high, decoder has decoded a picture
11:10	RO	0x0	reserved
9	RW	0x0	sw_dec_irq_raw the raw status of sw_dec_irq the raw status of sw_dec_irq,SW should reset this bit after interrupt is handled
8	RO	0x0	sw_dec_irq decoder IRQ when high, decoder requests an interrrupt. sw_dec_irq = sw_dec_irq_raw && (sw_dec_irq_dis == 1'b0)
7	RW	0x0	sw_stmerror_waitdecfifo_empty whether the stream error process wait the decfifo empty when it is 1'b0, the stream error process will no wait the ca2decfifo empty when it is 1'b1, the stream error process will wait the ca2decfifo empty
6	RW	0x0	sw_buf_empty_en buffer empty interrupt enable
5	RW	0x1	sw_dec_timeout_e Timeout interrupt enable If enabled HW may return timeout interrupt in case HW gets stucked while decoding picture.
4	RW	0x0	sw_dec_irq_dis decoder IRQ disable When hight, there are no interrupts concerning decoder from HW. Polling must be used to see the interrupt status
3:2	RO	0x0	reserved
1	RW	0x1	sw_dec_clkgate_e decoder dynamic clock gating enable 0 = clock is running for all structures 1 = clock is gated for decoder structures that are not used
0	RW	0x0	sw_dec_e Decoder enable. Setting this bit high will start the decoding operation. HW will reset this when picture is processed or stream error is detected or bus error or time out interrupt is given

hevc_swreg2_sysctrl
Address: Operational Base + offset (0x0008)
Data input and output endian setting and sys ctrl

Bit	Attr	Reset Value	Description
31:19	RO	0x0	reserved
18:12	RW	0x00	sw_strm_start_bit exact bit of stream start exact bit of streamd start word where decoding can be started (asosiates with sw_str_rlc_base)
11	RW	0x0	sw_rlc_mode rlc mode enable 0 = HW decodes video from bit stream 1 = HW decodes video from RLC input data
10	RW	0x0	sw_rlc_mode_direct_write cabac decode output direct write cabac decode output direct write enable when this bit is enable , all the module other than cabac and busifd are not work
9	RO	0x0	reserved
8	RW	0x0	sw_out_cbcr_swap output cbcr swap 0: cb(u) is in the lower address, cr(v) is in the higher address 1: cb(u) is in the higher address,cr(v) is in the lower address sw_in_cbcr_swap is the same with sw_out_cbcr_swap
7	RW	0x0	sw_out_swap32_e decoder output data and dpb input data 32bit swap may be used for 64 or 128 bit environment 0 = no swapping of 32 bit words 1 = 32 bit data words are swapped
6	RW	0x0	sw_out_endian dec output data and colmv , dpb data and colmv input endian 0 = little endian 1 = big endian for litter enadian , a data 0x12345678, 0x78 is stored in lower address, 0x12 is stored in higher address
5	RW	0x0	sw_str_swap64_e stream 64bit data swap may be used for 128 bit environment 0 = no swapping of 64 bit words 1 = 64 bit data words are swapped
4	RW	0x0	sw_str_swap32_e stream 32bit data swap may be used for 64 or 128 bit environment 0 = no swapping of 32 bit words 1 = 32 bit data words are swapped

Bit	Attr	Reset Value	Description
3	RW	0x0	sw_str_endian stream data input endian mode 0 = little endian 1 = big endian for litter enadian , a data 0x12345678, 0x78 is stored in lower address, 0x12 is stored in higher address
2	RW	0x0	sw_in_swap64_e input 64bit data swap for other than stream and dpb data may be used for 128 bit environment 0 = no swapping of 64 bit words 1 = 64 bit data words are swapped
1	RW	0x0	sw_in_swap32_e input 32bit data swap for other than stream and dpb data may be used for 64 or 128 bit environment 0 = no swapping of 32 bit words 1 = 32 bit data words are swapped
0	RW	0x0	sw_in_endian decoder input endian mode for other than stream and dpb data 0 = little endian 1 = big endian for litter enadian , a data 0x12345678, 0x78 is stored in lower address, 0x12 is stored in higher address

hevc_swreg3_picpar
Address: Operational Base + offset (0x000c)
picture parameters

Bit	Attr	Reset Value	Description
31	RO	0x0	reserved
30:21	RW	0x000	sw_slice_num slice number in a frame (when it is 0, it real means 1 slice in a frame)
20:12	RW	0x000	sw_uv_hor_virstride picture horizontal virtual stride (the unit is 128bit) the max is (4096x1.5 + 128) /16 = 0x188 suggest this register to config to even for advance ddr performance
11:9	RO	0x0	reserved
8:0	RW	0x000	sw_y_hor_virstride picture horizontal virtual stride (the unit is 128bit) the max is (4096x1.5 + 128) /16 = 0x188 suggest this register to config to even for advance ddr performance

hevc_swreg4_strm_rlc_base
Address: Operational Base + offset (0x0010)

the stream or rlc data base address

Bit	Attr	Reset Value	Description
31:4	RW	0x0000000	sw_strm_rlc_base the stream or rlc data base address when swreg2.sw_rlc_mode =1, it is base address for rlc data when swreg2.sw_rlc_mode =0, it is base address for stream , after a frame is decoded ready or error (stream error , time out , bus error) , it is the last address of the stream the address should 128bit align
3:0	RO	0x0	reserved

hevc_swreg5_stream_rlc_len
Address: Operational Base + offset (0x0014)
amount of stream bytes or rlc data byte in the input buffer or the

Bit	Attr	Reset Value	Description
31:27	RO	0x0	reserved
26:0	RW	0x0000000	sw_stream_len amount of stream (unit is 8bit) in the input buffer amount of stream 8bits in the input buffer the max of sw_stream_len : 4096x2304x1.5x1.5 = 0x1440000 128bits unit: 0x1440000/16 = 0x144000 it is count from 0 2013.10.15 change to 23bit for zty's suggestion 2013.10.28, amount of stream data bytes in input buffer. it is count from 1, change to 27bits

hevc_swreg6_cabactbl_base
Address: Operational Base + offset (0x0018)
the base address of cabac table

Bit	Attr	Reset Value	Description
31:4	RW	0x0000000	sw_cabactbl_base the base address of cabac table the address should 128bit align
3:0	RO	0x0	reserved

hevc_swreg7_decout_base
Address: Operational Base + offset (0x001c)
base address of decoder output picture base address

Bit	Attr	Reset Value	Description
31:4	RW	0x0000000	sw_decout_base base address of decoder output picture addr the address should be 128bit align
3:0	RO	0x0	reserved

hevc_swreg8_y_virstride
Address: Operational Base + offset (0x0020)

the ouput picture y virtual stride

Bit	Attr	Reset Value	Description
31:20	RO	0x0	reserved
19:0	RW	0x00000	sw_y_virstride the output picture y virtual stride (the unit is 128bit) the max: (4096x1.5 +128) x 2304 = 0xdc8000 we can know the sw_uvout_base = sw_decout_base + (sw_y_virstride <<4)

hevc_swreg9_yuv_virstride
Address: Operational Base + offset (0x0024)
the ouput picture yuv virtual stride

Bit	Attr	Reset Value	Description
31:21	RO	0x0	reserved
20:0	RW	0x000000	sw_yuv_virstride the output picture yuv virtual stride (the unit is 128bit) the max : (4096x1.5 +128) x 2304 x1.5 = 0x14ac000 we can know the sw_mvout_base = sw_decout_base + (sw_yuv_virstride <<4)

hevc_swreg10_refer0_base
Address: Operational Base + offset (0x0028)
base address for reference picture index 0

Bit	Attr	Reset Value	Description
31:4	RW	0x0000000	sw_refer0_base base address for reference picture index 0 (the address should be 128bit align)
3:0	RW	0x0	sw_ref_valid_0_3 valid flag for picture index 0 ~3

hevc_swreg11_refer1_base
Address: Operational Base + offset (0x002c)
base address for reference picture index 1

Bit	Attr	Reset Value	Description
31:4	RW	0x0000000	sw_refer1_base base address for reference picture index 1 (the address should be 128bit align)
3:0	RW	0x0	sw_ref_valid_4_7 valid flag for picture index 4 ~7

hevc_swreg12_refer2_base
Address: Operational Base + offset (0x0030)
base address for reference picture index 2

Bit	Attr	Reset Value	Description
31:4	RW	0x0000000	sw_refer2_base base address for reference picture index 2 (the address should be 128bit align)
3:0	RW	0x0	sw_ref_valid_8_11 valid flag for picture index 8~11

hevc_swreg13_refer3_base
Address: Operational Base + offset (0x0034)
base address for reference picture index 3

Bit	Attr	Reset Value	Description
31:4	RW	0x0000000	sw_refer3_base base address for reference picture index 3 (the address should be 128bit align)
3	RO	0x0	reserved
2:0	RW	0x0	sw_ref_valid_12_14 valid flag for picture index 12~14

hevc_swreg14_refer4_base
Address: Operational Base + offset (0x0038)
base address for reference picture index 4

Bit	Attr	Reset Value	Description
31:4	RW	0x0000000	sw_refer4_base base address for reference picture index 4(the address should be 128bit align)
3:0	RO	0x0	reserved

hevc_swreg15_refer5_base
Address: Operational Base + offset (0x003c)
base address for reference picture index 5

Bit	Attr	Reset Value	Description
31:4	RW	0x0000000	sw_refer5_base base address for reference picture index 5(the address should be 128bit align)
3:0	RO	0x0	reserved

hevc_swreg16_refer6_base
Address: Operational Base + offset (0x0040)
base address for reference picture index 6

Bit	Attr	Reset Value	Description
31:4	RW	0x0000000	sw_refer6_base base address for reference picture index 6(the address should be 128bit align)
3:0	RO	0x0	reserved

hevc_swreg17_refer7_base
Address: Operational Base + offset (0x0044)
base address for reference picture index 7

Bit	Attr	Reset Value	Description
31:4	RW	0x0000000	sw_refer7_base base address for reference picture index 7(the address should be 128bit align)
3:0	RO	0x0	reserved

hevc_swreg18_refer8_base
Address: Operational Base + offset (0x0048)
base address for reference picture index 8

Bit	Attr	Reset Value	Description
31:4	RW	0x0000000	sw_refer8_base base address for reference picture index 8(the address should be 128bit align)
3:0	RO	0x0	reserved

hevc_swreg19_refer9_base
Address: Operational Base + offset (0x004c)
base address for reference picture index 9

Bit	Attr	Reset Value	Description
31:4	RW	0x0000000	sw_refer9_base base address for reference picture index 9(the address should be 128bit align)
3:0	RO	0x0	reserved

hevc_swreg20_refer10_base
Address: Operational Base + offset (0x0050)
base address for reference picture index 10

Bit	Attr	Reset Value	Description
31:4	RW	0x0000000	sw_refer10_base base address for reference picture index 10(the address should be 128bit align)
3:0	RO	0x0	reserved

hevc_swreg21_refer11_base
Address: Operational Base + offset (0x0054)
base address for reference picture index 11

Bit	Attr	Reset Value	Description
31:4	RW	0x0000000	sw_refer11_base base address for reference picture index 11(the address should be 128bit align)
3:0	RO	0x0	reserved

hevc_swreg22_refer12_base

Address: Operational Base + offset (0x0058)
base address for reference picture index 12

Bit	Attr	Reset Value	Description
31:4	RW	0x0000000	sw_refer12_base base address for reference picture index 12(the address should be 128bit align)
3:0	RO	0x0	reserved

hevc_swreg23_refer13_base
Address: Operational Base + offset (0x005c)
base address for reference picture index 13

Bit	Attr	Reset Value	Description
31:4	RW	0x0000000	sw_refer13_base base address for reference picture index 13(the address should be 128bit align)
3:0	RO	0x0	reserved

hevc_swreg24_refer14_base
Address: Operational Base + offset (0x0060)
base address for reference picture index 14

Bit	Attr	Reset Value	Description
31:4	RW	0x0000000	sw_refer14_base base address for reference picture index 14(the address should be 128bit align)
3:0	RO	0x0	reserved

hevc_swreg25_refer0_poc
Address: Operational Base + offset (0x0064)
the poc of reference picture index 0

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	sw_refer0_poc the poc of reference picture index 0

hevc_swreg26_refer1_poc
Address: Operational Base + offset (0x0068)
the poc of reference picture index 1

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	sw_refer1_poc the poc of reference picture index 1 the poc of reference picture index 1

hevc_swreg27_refer2_poc
Address: Operational Base + offset (0x006c)
the poc of reference picture index 2

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	sw_refer2_poc the poc of reference picture index 2

hevc_swreg28_refer3_poc
Address: Operational Base + offset (0x0070)
the poc of reference picture index 3

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	sw_refer3_poc the poc of reference picture index 3

hevc_swreg29_refer4_poc
Address: Operational Base + offset (0x0074)
the poc of reference picture index 4

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	sw_refer4_poc the poc of reference picture index 4

hevc_swreg30_refer5_poc
Address: Operational Base + offset (0x0078)
the poc of reference picture index 5

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	sw_refer5_poc the poc of reference picture index 5

hevc_swreg31_refer6_poc
Address: Operational Base + offset (0x007c)
the poc of reference picture index 6

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	sw_refer6_poc the poc of reference picture index 6

hevc_swreg32_refer7_poc
Address: Operational Base + offset (0x0080)
the poc of reference picture index 7

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	sw_refer7_poc the poc of reference picture index 7

hevc_swreg33_refer8_poc
Address: Operational Base + offset (0x0084)
the poc of reference picture index 8

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	sw_refer8_poc the poc of reference picture index 8

hevc_swreg34_refer9_poc
Address: Operational Base + offset (0x0088)
the poc of reference picture index 9

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	sw_refer9_poc the poc of reference picture index 9

hevc_swreg35_refer10_poc
Address: Operational Base + offset (0x008c)
the poc of reference picture index 10

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	sw_refer10_poc the poc of reference picture index 10

hevc_swreg36_refer11_poc
Address: Operational Base + offset (0x0090)
the poc of reference picture index 11

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	sw_refer11_poc the poc of reference picture index 11

hevc_swreg37_refer12_poc
Address: Operational Base + offset (0x0094)
the poc of reference picture index 12

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	sw_refer12_poc the poc of reference picture index 12

hevc_swreg38_refer13_poc
Address: Operational Base + offset (0x0098)
the poc of reference picture index 13

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	sw_refer13_poc the poc of reference picture index 13

hevc_swreg39_refer14_poc
Address: Operational Base + offset (0x009c)
the poc of reference picture index 14

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	sw_refer14_poc the poc of reference picture index 14

hevc_swreg40_cur_poc
Address: Operational Base + offset (0x00a0)
the poc of cur picture

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	sw_cur_poc the poc of the cur picture

hevc_swreg41_rlcwrite_base
Address: Operational Base + offset (0x00a4)
the base address or rlcwrite base addr

Bit	Attr	Reset Value	Description
31:3	RW	0x00000000	sw_rlcwrite_base the base address of rlcwrite(the address should 64bit align) cabac output write to this rlcwrite base address when sw_rlc_mode_direct_write in swreg2_sysctrl is valid
2:0	RO	0x0	reserved

hevc_swreg42_pps_base
Address: Operational Base + offset (0x00a8)
the base address of pps

Bit	Attr	Reset Value	Description
31:4	RW	0x0000000	sw_pps_base the base address of pps (the address should 128bit align) it is for storing sps(sequence parameter set) and pps(picture parameter set)
3:0	RO	0x0	reserved

hevc_swreg43_rps_base
Address: Operational Base + offset (0x00ac)
the base address of rps

Bit	Attr	Reset Value	Description
31:4	RW	0x0000000	sw_rps_base rps(reference picture set) base address (the address should 128bit align)
3:0	RO	0x0	reserved

hevc_swreg44_cabac_error_en
Address: Operational Base + offset (0x00b0)
cabac error enable config

Bit	Attr	Reset Value	Description
31:28	RO	0x0	reserved
27:0	RW	0x0000000	sw_cabac_error_e cabac error enable regs

hevc_swreg45_cabac_error_status
Address: Operational Base + offset (0x00b4)
cabac error status

Bit	Attr	Reset Value	Description
31:28	RW	0x0	sw_colmv_error_ref_picidx colmv error ref picidx when sw_colmv_ref_error_sta is 1'b1, these bits are used for tell which dpb frame is invalid but is read by inter module
27:0	RW	0x0000000	sw_cabac_error_status cabac error status

hevc_swreg46_cabac_error_ctu
Address: Operational Base + offset (0x00b8)
cabac error ctu

Bit	Attr	Reset Value	Description
31:23	RO	0x0	reserved
22:16	RW	0x40	sw_streamfifo_space2full stream fifo space to full It is for debug use, to tell the stream fifo space to full
15:8	RW	0x00	sw_cabac_error_ctu_yoffset cabac error ctu yoffset
7:0	RW	0x00	sw_cabac_error_ctu_xoffset cabac error ctu xoffset

hevc_swreg47_sao_ctu_position
Address: Operational Base + offset (0x00bc)
sao ctu position

Bit	Attr	Reset Value	Description
31:26	RO	0x0	reserved
25:16	RW	0x000	sw_saowr_yoffset saowr y offset , its unit is 4 pixels
15:9	RO	0x0	reserved
8:0	RW	0x000	sw_saowr_xoffet saowr x address offset, its unit is 128bit

hevc_swreg64_performance_cycle
Address: Operational Base + offset (0x0100)
hevc performance cycle

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	sw_performance_cycle hevc running cycle if just want to analys a frame performance cycle, should set the register 0 before start a frame

hevc_swreg65_axi_ddr_rdata
Address: Operational Base + offset (0x0104)
axi ddr read data num

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	sw_axi_ddr_rdata axi ddr rdata num, the unit is byte

hevc_swreg66_axi_ddr_wdata
Address: Operational Base + offset (0x0108)
axi ddr write data number

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	sw_axi_ddr_wdata hevc write data byte num

2.4.3 VDP_HEVC MMU Register Summary

Name	Offset	Size	Reset Value	Description
hevc_mmu_DTE_ADDR	0x0000	W	0x00000000	MMU current page Table address
hevc_mmu_STATUS	0x0004	W	0x00000018	MMU status register
hevc_mmu_COMMAND	0x0008	W	0x00000000	MMU command register
hevc_mmu_PAGE_FAULT_ADDR	0x000c	W	0x00000000	MMU logical address of last page fault
hevc_mmu_ZAP_ONE_LINE	0x0010	W	0x00000000	MMU Zap cache line register
hevc_mmu_INT_RAWSTAT	0x0014	W	0x00000000	MMU raw interrupt status register
hevc_mmu_INT_CLEAR	0x0018	W	0x00000000	MMU raw interrupt status register
hevc_mmu_INT_MASK	0x001c	W	0x00000000	MMU raw interrupt status register
hevc_mmu_INT_STATUS	0x0020	W	0x00000000	MMU raw interrupt status register
hevc_mmu_AUTO_GATING	0x0024	W	0x00000001	mmu auto gating

*Notes:Size:**B**- Byte (8 bits) access, **HW**- Half WORD (16 bits) access, **W**-WORD (32 bits) access*

2.4.4 VDP_HEVC MMU Detail Register Description

hevc_mmu_DTE_ADDR
Address: Operational Base + offset (0x0000)
MMU current page Table address

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	MMU_DTE_ADDR mmu dte base addr mmu dte base addr , the address must be 4kb aligned

hevc_mmu_STATUS
Address: Operational Base + offset (0x0004)
MMU status register

Bit	Attr	Reset Value	Description
31:11	RO	0x0	reserved

Bit	Attr	Reset Value	Description
10:6	RO	0x00	PAGE_FAULT_BUS_ID page fault bus id Index of master reponsible for last page fault
5	RO	0x0	PAGE_FAULT_IS_WRITE page fault acess The direction of access for last page fault: 0 = Read 1 = Write
4	RO	0x1	REPLAY_BUFFER_EMPTY replay buffer empty status 1'b1:The MMU replay buffer is empty
3	RO	0x1	MMU_IDLE mmu idle status The MMU is idle when accesses are being translated and there are no unfinished translated accesses. 1'b1: MMU is idle
2	RO	0x0	STAIL_ACTIVE stall active status MMU stall mode currently enabled. The mode is enabled by command 1'b1: MMU is in stall active status
1	RO	0x0	PAGE_FAULT_ACTIVE page fault active status MMU page fault mode currently enabled . The mode is enabled by command. 1'b1: page fault is active
0	RO	0x0	PAGING_ENABLED Paging enabled status 1'b0: paging is disabled 1'b1: Paging is enabled

hevc_mmu_COMMAND

Address: Operational Base + offset (0x0008)

MMU command register

Bit	Attr	Reset Value	Description
31:3	RO	0x0	reserved

Bit	Attr	Reset Value	Description
2:0	WO	0x0	MMU_CMD Field0000 Abstract MMU_CMD. This can be: 0: MMU_ENABLE_PAGING 1: MMU_DISABLE_PAGING 2: MMU_ENABLE_STALL 3: MMU_DISABLE_STALL 4: MMU_ZAP_CACHE 5: MMU_PAGE_FAULT_DONE 6: MMU_FORCE_RESET

hevc_mmu_PAGE_FAULT_ADDR
Address: Operational Base + offset (0x000c)
MMU logical address of last page fault

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	PAGE_FAULT_ADDR Field0000 Abstract address of last page fault

hevc_mmu_ZAP_ONE_LINE
Address: Operational Base + offset (0x0010)
MMU Zap cache line register

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	MMU_ZAP_ONE_LINE Field0000 Abstract address to be invalidated from the page table cache

hevc_mmu_INT_RAWSTAT
Address: Operational Base + offset (0x0014)
MMU raw interrupt status register

Bit	Attr	Reset Value	Description
31:2	RO	0x0	reserved
1	RW	0x0	READ_BUS_ERROR Field0000 Abstract read bus error
0	RW	0x0	PAGE_FAULT Field0000 Abstract page fault

hevc_mmu_INT_CLEAR
Address: Operational Base + offset (0x0018)
MMU raw interrupt status register

Bit	Attr	Reset Value	Description
31:2	RO	0x0	reserved

Bit	Attr	Reset Value	Description
1	WO	0x0	READ_BUS_ERROR Field0000 Abstract read bus error
0	WO	0x0	PAGE_FAULT Field0000 Abstract page fault

hevc_mmu_INT_MASK
Address: Operational Base + offset (0x001c)
MMU raw interrupt status register

Bit	Attr	Reset Value	Description
31:2	RO	0x0	reserved
1	RW	0x0	READ_BUS_ERROR Field0000 Abstract read bus error enable an interrupt source if the corresponding mask bit is set to 1
0	RW	0x0	PAGE_FAULT Field0000 Abstract page fault enable an interrupt source if the corresponding mask bit is set to 1

hevc_mmu_INT_STATUS
Address: Operational Base + offset (0x0020)
MMU raw interrupt status register

Bit	Attr	Reset Value	Description
31:2	RO	0x0	reserved
1	RO	0x0	READ_BUS_ERROR Field0000 Abstract read bus error
0	RO	0x0	PAGE_FAULT Field0000 Abstract page fault

hevc_mmu_AUTO_GATING
Address: Operational Base + offset (0x0024)
mmu auto gating

Bit	Attr	Reset Value	Description
31:1	RO	0x0	reserved
0	RW	0x1	mmu_auto_gating mmu auto gating when it is 1'b1, the mmu will auto gating it self

2.4.5 VDP_HEVC Pref_cache Register Summary

Name	Offset	Size	Reset Value	Description
pref_cache_VERSION	0x0000	W	0xcac20101	VERSION register
pref_cache_SIZE	0x0004	W	0x07110206	L2 cache SIZE
pref_cache_STATUS	0x0008	W	0x00000000	Status register
pref_cache_COMMAND	0x0010	W	0x00000000	Command setting register
pref_cache_CLEAR_PAGE	0x0014	W	0x00000000	clear page register
pref_cache_MAX_READS	0x0018	W	0x0000001c	maximum read register
pref_cache_ENABLE	0x001c	W	0x00000003	enables cacheable accesses and cache read allocation
pref_cache_PERFCNT_SRC0	0x0020	W	0x00000000	performance counter 0 source register
pref_cache_PERFCNT_VAL0	0x0024	W	0x00000000	performance counter 0 value register
pref_cache_PERFCNT_SRC1	0x0028	W	0x00000000	performance counter 0 source register
pref_cache_PERFCNT_VAL1	0x002c	W	0x00000000	performance counter 1 value register

*Notes:Size:**B**- Byte (8 bits) access, **HW**- Half WORD (16 bits) access, **W**-WORD (32 bits) access*

2.4.6 VDP_HEVC Pref_cache Detail Register Description

pref_cache_VERSION

Address: Operational Base + offset (0x0000)

VERSION register

Bit	Attr	Reset Value	Description
31:16	RO	0xcac2	PRODUCT_ID Field0000 Abstract Field0000 Description
15:8	RO	0x01	VERSION_MAJOR Field0000 Abstract Field0000 Description
7:0	RO	0x01	VERSION_MINOR Field0000 Abstract Field0000 Description

pref_cache_SIZE

Address: Operational Base + offset (0x0004)

L2 cache SIZE

Bit	Attr	Reset Value	Description
31:24	RO	0x07	External_bus_width Field0000 Abstract Log2 external bus width in bits
23:16	RO	0x11	CACHE_SIZE Field0000 Abstract Log2 cache size in bytes

Bit	Attr	Reset Value	Description
15:8	RO	0x02	ASSOCIATIVITY Field0000 Abstract Log2 associativity
7:0	RO	0x06	LINE_SIZE Field0000 Abstract Log2 line size in bytes

pref_cache_STATUS
Address: Operational Base + offset (0x0008)
Status register

Bit	Attr	Reset Value	Description
31:2	RO	0x0	reserved
1	RW	0x0	DATA_BUSY Field0000 Abstract set when the cache is busy handling data
0	RW	0x0	CMD_BUSY Field0000 Abstract set when the cache is busy handling commands

pref_cache_COMMAND
Address: Operational Base + offset (0x0010)
Command setting register

Bit	Attr	Reset Value	Description
31:6	RO	0x0	reserved
5:4	RW	0x0	sw_addrb_sel Field0000 Abstract 2'b00: to sel b[14:6] 2'b01: to sel b[15:9], b[7:6] 2'b10: to sel b[16:10], b[7:6] 2'b11: to sel b[17:11], b[7:6]
3:0	RW	0x0	COMMAND Field0000 Abstract The possible command is 1 = Clear entire cache

pref_cache_CLEAR_PAGE
Address: Operational Base + offset (0x0014)
clear page register

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	CLEAR_PAGE Field0000 Abstract writing an address, invlidates all lines in that page from the cache

pref_cache_MAX_READS
Address: Operational Base + offset (0x0018)

maximum read register

Bit	Attr	Reset Value	Description
31:5	RO	0x0	reserved
4:0	RW	0x1c	MAX_READS Field0000 Abstract Limit the number of outstanding read transactions to this amount

pref_cache_ENABLE

Address: Operational Base + offset (0x001c)

enables cacheable accesses and cache read allocation

Bit	Attr	Reset Value	Description
31:4	RO	0x0	reserved
3	RW	0x0	sw_cache_clk_disgate cache clk disgate cache clk disgate when it is 1'b0, enable cache clk auto clkgating when it is 1'b1, disable cache clk auto clkgating
2	RW	0x0	sw_readbuffer_counter_reject_en counter reject enable default is 1'b0, for enhance cacheable read performnace in readbuffer. 1'b1: normal origin counter reject
1	RW	0x1	permit_cache_read_allocate cache read allocate 1'b1: permit cache read allocate
0	RW	0x1	permit_cacheable_access cacheable access 1'b1: permit cacheable access

pref_cache_PERFCNT_SRC0

Address: Operational Base + offset (0x0020)

performance counter 0 source register

Bit	Attr	Reset Value	Description
31:7	RO	0x0	reserved

Bit	Attr	Reset Value	Description
6:0	RW	0x00	PERFCNT_SRC0 Field0000 Abstract This register holds all the possible source values for Performance Counter 0 0: disabled 1: total clock cycles 2: active clock cycles 3: read transactions, master 4: word reads, master 5: read transactions, slave 6: word reads, slave 7: read hit, slave 8: read misses, slave 9: read invalidates, slave 10: cacheable read transactions, slave 11: bad hit nmber, slave

pref_cache_PERFCNT_VAL0
Address: Operational Base + offset (0x0024)
performance counter 0 value register

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	PERFCNT_VAL0 Field0000 Abstract Performance counter 0 value

pref_cache_PERFCNT_SRC1
Address: Operational Base + offset (0x0028)
performance counter 0 source register

Bit	Attr	Reset Value	Description
31:7	RO	0x0	reserved

Bit	Attr	Reset Value	Description
6:0	RW	0x00	PERFCNT_SRC1 Field0000 Abstract This register holds all the possible source values for Performance Counter 1 0: disabled 1: total clock cycles 2: active clock cycles 3: read transactions, master 4: word reads, master 5: read transactions, slave 6: word reads, slave 7: read hit, slave 8: read misses, slave 9: read invalidates, slave 10: cacheable read transactions, slave 11: bad hit nmber, slave

pref_cache_PERFCNT_VAL1
Address: Operational Base + offset (0x002c)
performance counter 1 value register

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	PERFCNT_VAL1 Field0000 Abstract Performance counter 1 value

2.4.7 VEPU Register Summary

Name	Offset	Size	Reset Value	Description
VEPU_swreg0	0x0000	W	0x48310000	product ID
VEPU_swreg1	0x0004	W	0x00000000	interrrupt control and status
VEPU_swreg2	0x0008	W	0x01010010	axi control
VEPU_swreg3	0x000c	W	0x00000000	test control register
VEPU_swreg4	0x0010	W	0x00000000	reserverd
VEPU_swreg5	0x0014	W	0x00000000	addr_output_stream
VEPU_swreg6	0x0018	W	0x00000000	base address for output control
VEPU_swreg7	0x001c	W	0x00000000	base address for reference luma
VEPU_swreg8	0x0020	W	0x00000000	base address for reference chroma
VEPU_swreg9	0x0024	W	0x00000000	base address for reconstructed luma
VEPU_swreg10	0x0028	W	0x00000000	base address for reconstructed chroma
VEPU_swreg11	0x002c	W	0x00000000	base addr for input luma
VEPU_swreg12	0x0030	W	0x00000000	base address for input cb
VEPU_swreg13	0x0034	W	0x00000000	base address for input cr

Name	Offset	Size	Reset Value	Description
VEPU_swreg14	0x0038	W	0x00000000	enc control
VEPU_swreg15	0x003c	W	0x00000000	input image control
VEPU_swreg16	0x0040	W	0x00000000	encoder control regsiter 0
VEPU_swreg16_reuse	0x0040	W	0x00000000	
VEPU_swreg17_reuse	0x0044	W	0x00000000	
VEPU_swreg17	0x0044	W	0x00000000	encoder control register 1
VEPU_swreg18	0x0048	W	0x00000000	encoder control register 2
VEPU_swreg19	0x004c	W	0x00000000	encoder control register 3
VEPU_swreg20	0x0050	W	0x00000000	JPEG control regsiter
VEPU_swreg20_reuse	0x0050	W	0x00000000	JPEG control regsiter
VEPU_swreg21	0x0054	W	0x00000000	encoder control register 4
VEPU_swreg22	0x0058	W	0x00000000	stream header remainder bits MSB
VEPU_swreg23	0x005c	W	0x00000000	stream header remainder bits LSB
VEPU_swreg24	0x0060	W	0x00000000	stream buffer limit
VEPU_swreg25	0x0064	W	0x00000000	MAD control register
VEPU_swreg26	0x0068	W	0x00000000	
VEPU_swreg26_vp8	0x0068	W	0x00000000	
VEPU_swreg27	0x006c	W	0x00000000	QP register
VEPU_swreg27_vp8	0x006c	W	0x00000000	
VEPU_swreg28	0x0070	W	0x00000000	checkpoint 1 and 2
VEPU_swreg28_vp8	0x0070	W	0x00000000	
VEPU_swreg29_vp8	0x0074	W	0x00000000	
VEPU_swreg29	0x0074	W	0x00000000	checkpoint 3 and 4
VEPU_swreg30_vp8	0x0078	W	0x00000000	
VEPU_swreg30	0x0078	W	0x00000000	checkpoint 5 and 6
VEPU_swreg31_vp8	0x007c	W	0x00000000	
VEPU_swreg31	0x007c	W	0x00000000	checkpoint 7 and 8
VEPU_swreg32	0x0080	W	0x00000000	checkpoint 9 and 10
VEPU_swreg32_vp8	0x0080	W	0x00000000	
VEPU_swreg33	0x0084	W	0x00000000	checkpoint word error 1 and 2
VEPU_swreg33_vp8	0x0084	W	0x00000000	
VEPU_swreg34_vp8	0x0088	W	0x00000000	
VEPU_swreg34	0x0088	W	0x00000000	checkpoint word error 1 and 2
VEPU_swreg35	0x008c	W	0x00000000	checkpoint word error 1 and 2
VEPU_swreg35_vp8	0x008c	W	0x00000000	checkpoint word error 1 and 2
VEPU_swreg36	0x0090	W	0x00000000	checkpoint delta QP register
VEPU_swreg36_vp8	0x0090	W	0x00000000	checkpoint delta QP register
VEPU_swreg37	0x0094	W	0x00000000	rlc control
VEPU_swreg38	0x0098	W	0x00000000	mb control register
VEPU_swreg39	0x009c	W	0x00000000	Base address for next pic
VEPU_swreg40	0x00a0	W	0x00000000	Stabilization minimum value
VEPU_swreg41	0x00a4	W	0x00000000	Stabilization motion sum

Name	Offset	Size	Reset Value	Description
VEPU_swreg42	0x00a8	W	0x00000000	stab_matrix1 and stab_gmv_hor
VEPU_swreg43	0x00ac	W	0x00000000	stab_matrix2 and stab_gmv_ver
VEPU_swreg44	0x00b0	W	0x00000000	stab_matrix3
VEPU_swreg45	0x00b4	W	0x00000000	stab_matrix4
VEPU_swreg46	0x00b8	W	0x00000000	stab_matrix5
VEPU_swreg47	0x00bc	W	0x00000000	stab_matrix6
VEPU_swreg48	0x00c0	W	0x00000000	stab_matrix7
VEPU_swreg49	0x00c4	W	0x00000000	stab_matrix8
VEPU_swreg50	0x00c8	W	0x00000000	stab_matrix9
VEPU_swreg51	0x00cc	W	0x00000000	cabac_table_addr
VEPU_swreg52	0x00d0	W	0x00000000	Base address for MV output
VEPU_swreg53	0x00d4	W	0x00000000	RGB to YUV conversion coefficien A and B
VEPU_swreg54	0x00d8	W	0x00000000	RGB to YUV conversion coefficien C and E
VEPU_swreg55	0x00dc	W	0x00000000	RGB mask MSB bit
VEPU_swreg56	0x00e0	W	0x00000000	intra area control register
VEPU_swreg57	0x00e4	W	0x00000000	CIR intra control reg
VEPU_swreg58_vp8	0x00e8	W	0x00000000	
VEPU_swreg58	0x00e8	W	0x00000000	Intra slice bitmap for slices 0..31
VEPU_swreg59_vp8	0x00ec	W	0x00000000	
VEPU_swreg59	0x00ec	W	0x00000000	Intra slice bitmap for slices32..63
VEPU_swreg60	0x00f0	W	0x00000000	1st ROI area register
VEPU_swreg61	0x00f4	W	0x00000000	Register0061 Abstract
VEPU_swreg62	0x00f8	W	0x00000000	MVC control reg
VEPU_swreg63	0x00fc	W	0x00000780	Register0063 Abstract
VEPU_swreg64	0x0100	W	0x00000000	JPEG luma quantization 1
VEPU_swreg64_vp8	0x0100	W	0x00000000	
VEPU_swreg65_vp8	0x0104	W	0x00000000	
VEPU_swreg65	0x0104	W	0x00000000	JPEG luma quantization 2
VEPU_swreg66_vp8	0x0108	W	0x00000000	
VEPU_swreg66	0x0108	W	0x00000000	JPEG luma quantization 3
VEPU_swreg67_vp8	0x010c	W	0x00000000	
VEPU_swreg67	0x010c	W	0x00000000	JPEG luma quantization 4
VEPU_swreg68_vp8	0x0110	W	0x00000000	
VEPU_swreg68	0x0110	W	0x00000000	JPEG luma quantization 5
VEPU_swreg69_vp8	0x0114	W	0x00000000	
VEPU_swreg69	0x0114	W	0x00000000	JPEG luma quantization 6
VEPU_swreg70_vp8	0x0118	W	0x00000000	
VEPU_swreg70	0x0118	W	0x00000000	JPEG luma quantization 7
VEPU_swreg71_vp8	0x011c	W	0x00000000	JPEG luma quantization 7
VEPU_swreg71	0x011c	W	0x00000000	JPEG luma quantization 8
VEPU_swreg72	0x0120	W	0x00000000	JPEG luma quantization 9

Name	Offset	Size	Reset Value	Description
VEPU_swreg72_vp8	0x0120	W	0x00000000	
VEPU_swreg73	0x0124	W	0x00000000	Register0073 Abstract
VEPU_swreg73_vp8	0x0124	W	0x00000000	
VEPU_swreg74_vp8	0x0128	W	0x00000000	
VEPU_swreg74	0x0128	W	0x00000000	JPEG luma quantization 11
VEPU_swreg75	0x012c	W	0x00000000	JPEG luma quantization 12
VEPU_swreg75_vp8	0x012c	W	0x00000000	
VEPU_swreg76_vp8	0x0130	W	0x00000000	
VEPU_swreg76	0x0130	W	0x00000000	JPEG luma quantization 13
VEPU_swreg77_vp8	0x0134	W	0x00000000	
VEPU_swreg77	0x0134	W	0x00000000	JPEG luma quantization 14
VEPU_swreg78_vp8	0x0138	W	0x00000000	
VEPU_swreg78	0x0138	W	0x00000000	JPEG luma quantization 15
VEPU_swreg79	0x013c	W	0x00000000	JPEG luma quantization 16
VEPU_swreg79_vp8	0x013c	W	0x00000000	
VEPU_swreg80_vp8	0x0140	W	0x00000000	
VEPU_swreg80	0x0140	W	0x00000000	JPEG chroma quantization 1
VEPU_swreg81	0x0144	W	0x00000000	JPEG chroma quantization 2
VEPU_swreg81_vp8	0x0144	W	0x00000000	
VEPU_swreg82_vp8	0x0148	W	0x00000000	
VEPU_swreg82	0x0148	W	0x00000000	JPEG chroma quantization 3
VEPU_swreg83	0x014c	W	0x00000000	JPEG chroma quantization 4
VEPU_swreg83_vp8	0x014c	W	0x00000000	
VEPU_swreg84_vp8	0x0150	W	0x00000000	
VEPU_swreg84	0x0150	W	0x00000000	JPEG chroma quantization 5
VEPU_swreg85	0x0154	W	0x00000000	JPEG chroma quantization 6
VEPU_swreg85_vp8	0x0154	W	0x00000000	
VEPU_swreg86_vp8	0x0158	W	0x00000000	
VEPU_swreg86	0x0158	W	0x00000000	JPEG chroma quantization 7
VEPU_swreg87	0x015c	W	0x00000000	JPEG chroma quantization 8
VEPU_swreg87_vp8	0x015c	W	0x00000000	
VEPU_swreg88_vp8	0x0160	W	0x00000000	
VEPU_swreg88	0x0160	W	0x00000000	JPEG chroma quantization 9
VEPU_swreg89_vp8	0x0164	W	0x00000000	
VEPU_swreg89	0x0164	W	0x00000000	JPEG chroma quantization 10
VEPU_swreg90_vp8	0x0168	W	0x00000000	
VEPU_swreg90	0x0168	W	0x00000000	JPEG chroma quantization 11
VEPU_swreg91_vp8	0x016c	W	0x00000000	
VEPU_swreg91	0x016c	W	0x00000000	JPEG chroma quantization 12
VEPU_swreg92_vp8	0x0170	W	0x00000000	
VEPU_swreg92	0x0170	W	0x00000000	JPEG chroma quantization 13
VEPU_swreg93_vp8	0x0174	W	0x00000000	
VEPU_swreg93	0x0174	W	0x00000000	JPEG chroma quantization 14

Name	Offset	Size	Reset Value	Description
VEPU_swreg94_vp8	0x0178	W	0x00000000	
VEPU_swreg94	0x0178	W	0x00000000	JPEG chroma quantization 15
VEPU_swreg95_vp8	0x017c	W	0x00000000	
VEPU_swreg95	0x017c	W	0x00000000	JPEG chroma quantization 16
VEPU_swreg96	0x0180	W	0x00000000	DMV 4p/1p penalty values 0-3
VEPU_swreg97	0x0184	W	0x00000000	DMV 4p/1p penalty values 4-7
VEPU_swreg98	0x0188	W	0x00000000	DMV 4p/1p penalty values
VEPU_swreg99	0x018c	W	0x00000000	DMV 4p/1p penalty values
VEPU_swreg100	0x0190	W	0x00000000	DMV 4p/1p penalty values
VEPU_swreg101	0x0194	W	0x00000000	DMV 4p/1p penalty values
VEPU_swreg102	0x0198	W	0x00000000	DMV 4p/1p penalty values
VEPU_swreg103	0x019c	W	0x00000000	DMV 4p/1p penalty values
VEPU_swreg104	0x01a0	W	0x00000000	DMV 4p/1p penalty values
VEPU_swreg105	0x01a4	W	0x00000000	DMV 4p/1p penalty values
VEPU_swreg106	0x01a8	W	0x00000000	DMV 4p/1p penalty values
VEPU_swreg107	0x01ac	W	0x00000000	DMV 4p/1p penalty values
VEPU_swreg108	0x01b0	W	0x00000000	DMV 4p/1p penalty values
VEPU_swreg109	0x01b4	W	0x00000000	DMV 4p/1p penalty values
VEPU_swreg110	0x01b8	W	0x00000000	DMV 4p/1p penalty values
VEPU_swreg111	0x01bc	W	0x00000000	DMV 4p/1p penalty values
VEPU_swreg112	0x01c0	W	0x00000000	DMV 4p/1p penalty values
VEPU_swreg113	0x01c4	W	0x00000000	DMV 4p/1p penalty values
VEPU_swreg114	0x01c8	W	0x00000000	DMV 4p/1p penalty values
VEPU_swreg115	0x01cc	W	0x00000000	DMV 4p/1p penalty values
VEPU_swreg116	0x01d0	W	0x00000000	DMV 4p/1p penalty values
VEPU_swreg117	0x01d4	W	0x00000000	DMV 4p/1p penalty values
VEPU_swreg118	0x01d8	W	0x00000000	DMV 4p/1p penalty values
VEPU_swreg119	0x01dc	W	0x00000000	DMV 4p/1p penalty values
VEPU_swreg120	0x01e0	W	0x00000000	DMV 4p/1p penalty values
VEPU_swreg121	0x01e4	W	0x00000000	DMV 4p/1p penalty values
VEPU_swreg122	0x01e8	W	0x00000000	DMV 4p/1p penalty values
VEPU_swreg123	0x01ec	W	0x00000000	DMV 4p/1p penalty values
VEPU_swreg124	0x01f0	W	0x00000000	DMV 4p/1p penalty values
VEPU_swreg125	0x01f4	W	0x00000000	DMV 4p/1p penalty values
VEPU_swreg126	0x01f8	W	0x00000000	DMV 4p/1p penalty values
VEPU_swreg127	0x01fc	W	0x00000000	DMV 4p/1p penalty values
VEPU_swreg128	0x0200	W	0x00000000	DMV 4p/1p penalty values
VEPU_swreg129	0x0204	W	0x00000000	DMV 4p/1p penalty values
VEPU_swreg130	0x0208	W	0x00000000	DMV 4p/1p penalty values
VEPU_swreg131	0x020c	W	0x00000000	DMV 4p/1p penalty values
VEPU_swreg132	0x0210	W	0x00000000	DMV 4p/1p penalty values
VEPU_swreg133	0x0214	W	0x00000000	DMV 4p/1p penalty values
VEPU_swreg134	0x0218	W	0x00000000	DMV 4p/1p penalty values

Name	Offset	Size	Reset Value	Description
VEPU_swreg135	0x021c	W	0x00000000	DMV 4p/1p penalty values
VEPU_swreg136	0x0220	W	0x00000000	DMV 4p/1p penalty values
VEPU_swreg137	0x0224	W	0x00000000	DMV 4p/1p penalty values
VEPU_swreg138	0x0228	W	0x00000000	DMV 4p/1p penalty values
VEPU_swreg139	0x022c	W	0x00000000	DMV 4p/1p penalty values
VEPU_swreg140	0x0230	W	0x00000000	DMV 4p/1p penalty values
VEPU_swreg141	0x0234	W	0x00000000	DMV 4p/1p penalty values
VEPU_swreg142	0x0238	W	0x00000000	DMV 4p/1p penalty values
VEPU_swreg143	0x023c	W	0x00000000	DMV 4p/1p penalty values
VEPU_swreg144	0x0240	W	0x00000000	DMV 4p/1p penalty values
VEPU_swreg145	0x0244	W	0x00000000	DMV 4p/1p penalty values
VEPU_swreg146	0x0248	W	0x00000000	DMV 4p/1p penalty values
VEPU_swreg147	0x024c	W	0x00000000	DMV 4p/1p penalty values
VEPU_swreg148	0x0250	W	0x00000000	DMV 4p/1p penalty values
VEPU_swreg149	0x0254	W	0x00000000	DMV 4p/1p penalty values
VEPU_swreg150	0x0258	W	0x00000000	DMV 4p/1p penalty values
VEPU_swreg151	0x025c	W	0x00000000	DMV 4p/1p penalty values
VEPU_swreg152	0x0260	W	0x00000000	DMV 4p/1p penalty values
VEPU_swreg153	0x0264	W	0x00000000	DMV 4p/1p penalty values
VEPU_swreg154	0x0268	W	0x00000000	DMV 4p/1p penalty values
VEPU_swreg155	0x026c	W	0x00000000	DMV 4p/1p penalty values
VEPU_swreg156	0x0270	W	0x00000000	DMV 4p/1p penalty values
VEPU_swreg157	0x0274	W	0x00000000	DMV 4p/1p penalty values
VEPU_swreg158	0x0278	W	0x00000000	DMV 4p/1p penalty values
VEPU_swreg159	0x027c	W	0x00000000	DMV 4p/1p penalty values
VEPU_swreg160	0x0280	W	0x00000000	vp8 control
VEPU_swreg161	0x0284	W	0x00000000	VP8 bit cost of golden ref frame
VEPU_swreg162	0x0288	W	0x00000000	vp8 loop filter delta registers
VEPU_swreg163	0x028c	W	0x00000000	vp8 loop filter delta register

*Notes:Size:**B**- Byte (8 bits) access, **HW**- Half WORD (16 bits) access, **W**-WORD (32 bits) access*

2.4.8 VEPU Detail Register Description

VEPU_swreg0

Address: Operational Base + offset (0x0000)

product ID

Bit	Attr	Reset Value	Description
31:16	RO	0x4831	prod_id Product ID
15:12	RW	0x0	major_num Major number
11:4	RW	0x00	minor_num Minor number
3:0	RW	0x0	synthesis

VEPU_swreg1
Address: Operational Base + offset (0x0004)
interrrupt control and status

Bit	Attr	Reset Value	Description
31:9	RO	0x0	reserved
8	RW	0x0	irq_slice_ready IRQ slice ready status bit
7	RO	0x0	reserved
6	RW	0x0	irq_timeout IRQ HW timeout status bit
5	RW	0x0	irq_buffer_full irq buffer full
4	RW	0x0	irq_reset irq SW reset
3	RW	0x0	irq_bus_error
2	RW	0x0	irq_frame_rdy IRQ frame ready status bit. Encoder has finished a frame
1	RW	0x0	irq_dis IRQ disable. No interrupts from HW. SW must use polling
0	RW	0x0	enc_irq HINTenc interrupt from HW. SW resets at IRQ handler.

VEPU_swreg2
Address: Operational Base + offset (0x0008)
axi control

Bit	Attr	Reset Value	Description
31:24	RW	0x01	axi_write_id axi write id axi write id
23:16	RW	0x01	axi_rd_id axi read id axi read id
15	RW	0x0	output_swap16 enable output swap 16-bits
14	RW	0x0	input_swap16 enable input swap 16-bits
13:8	RW	0x00	burst_len burst length
7	RW	0x0	disable_burst disable burst mode for AXI disable burst mode for AXI bus

Bit	Attr	Reset Value	Description
6	RW	0x0	burst_incr burst increment burst increment. 1: INCR burst allowed 0: use SINGLE burst
5	RW	0x0	burst_discard enable burst data dicard enable burst data dicard. 2 or 3 long reads are using BURST4
4	RW	0x1	clk_gating_en enable clock gating enable clock gating
3	RW	0x0	output_swap32 enable output swap 32-bits enable output swap 32-bits
2	RW	0x0	input_swap32 enable input swap 32-bits enable input swap 32-bits
1	RW	0x0	output_swap8 enable output swap 8-bits enable output swap 8-bits
0	RW	0x0	input_swap8 enable input swap 8-bits enable input swap 8-bits

VEPU_swreg3
Address: Operational Base + offset (0x000c)
test control register

Bit	Attr	Reset Value	Description
31:28	RW	0x0	test_counter test counter
27:21	RO	0x0	reserved
20:3	RW	0x00000	test_len test data length
2	RW	0x0	test_memory test memory coherency
1	RW	0x0	test_reg test register coherency
0	RW	0x0	test_irq test irq

VEPU_swreg4
Address: Operational Base + offset (0x0010)
reserverd

Bit	Attr	Reset Value	Description
31:1	RO	0x0	reserved

Bit	Attr	Reset Value	Description
0	RW	0x0	Field0000 Abstract Field0000 Description

VEPU_swreg5
Address: Operational Base + offset (0x0014)
addr_output_stream

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	addr_output_stream base address for output stream

VEPU_swreg6
Address: Operational Base + offset (0x0018)
base address for output control

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	addr_output_control base address for output control base address for output control

VEPU_swreg7
Address: Operational Base + offset (0x001c)
base address for reference luma

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	addr_reflum base address for reference luma base address for reference luma

VEPU_swreg8
Address: Operational Base + offset (0x0020)
base address for reference chroma

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	addr_refchroma base address for reference chroma base address for reference chroma

VEPU_swreg9
Address: Operational Base + offset (0x0024)
base address for reconstructed luma

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	addr_recon_luma base address for reconstructed luma

VEPU_swreg10
Address: Operational Base + offset (0x0028)
base address for reconstructed chroma

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	addr_recon_chroma base address for reconstructed chroma

VEPU_swreg11
Address: Operational Base + offset (0x002c)
base addr for input luma

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	addr_input_luma base addr for input luma

VEPU_swreg12
Address: Operational Base + offset (0x0030)
base address for input cb

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	addr_input_cb base address for input cb

VEPU_swreg13
Address: Operational Base + offset (0x0034)
base address for input cr

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	addr_input_cr base address for input cr

VEPU_swreg14
Address: Operational Base + offset (0x0038)
enc control

Bit	Attr	Reset Value	Description
31	RW	0x0	int_timeout_en enable interrupt for timeout
30	RW	0x0	mv_sad_wren enabel writing MV and SAD of each MB to BaseMvWrite
29	RW	0x0	nal_mode NAL size output to base control
28	RW	0x0	slice_rdyint_en enable interruupt for slice ready
27:19	RW	0x000	enc_width encoder width. lumwidth (macroblocks) H264: [9...255) JPEG: [6...511]
18:10	RW	0x000	enc_height encoderd height, lumHeight (macroblocks) H264: [6..255] JPEG: [6..511]
9:7	RO	0x0	reserved

Bit	Attr	Reset Value	Description
6	RW	0x0	rocon_write_dis disable writing of reconstructed image. recWriteDisable
5	RO	0x0	reserved
4:3	RW	0x0	enc_pic_type encoder picture type. frame type 2'd0: INTER 2'd1: INTRA(IDR) 2'd2: MVC-INTER
2:1	RW	0x0	enc_mode encoding mode. stream type , 1=VP8, 2=JPEG,3=H264
0	RW	0x0	enc_en encoder enable

VEPU_swreg15
Address: Operational Base + offset (0x003c)
input image control

Bit	Attr	Reset Value	Description
31:29	RW	0x0	input_chroma_offset input chrominance offset (bytes)
28:26	RW	0x0	input_lum_offset input luminance offset(bytes)
25:12	RW	0x0000	input_row_len input luminance row length
11:10	RW	0x0	overfill_right overfill pixels on right edge of image div4[0...3]
9:6	RW	0x0	overfill_bot overfill pixels on bottom edge of image. YFill [0..15]
5:2	RW	0x0	input_format input image format. YUV420P/YUV420SP/YUV422/UYVY422/RGB565/RGB444/RGB888/RGB101010
1:0	RW	0x0	imagein_rotmode input image rotation 2'd0: disabled 2'd1: 90 degress rigth 2'd2: 90 degress left

VEPU_swreg16
Address: Operational Base + offset (0x0040)
encoder control regsiter 0

Bit	Attr	Reset Value	Description
31:26	RW	0x00	init_qp H.264 pic init qp in PPS[0...51]
25:22	RW	0x0	slice_alpha H.264 slice filter alpha c0 offset div2 [-6 ....6]

Bit	Attr	Reset Value	Description
21:18	RW	0x0	slice_beta h.264 slice filter beta offset div2 [-6 ...6]
17:13	RW	0x00	chroma_qp_offset H264 chroma qp index offset [-12...12]
12:9	RO	0x0	reserved
8	RW	0x0	sw_qpass jpeg enc quant bypass
7:5	RO	0x0	reserved
4:1	RW	0x0	idr_picid IDR pic ID
0	RW	0x0	constr_intra_pred constrained intra prediction

VEPU_swreg16_reuse
Address: Operational Base + offset (0x0040)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	luma_sec_ref_addr Base address for second reference luma

VEPU_swreg17_reuse
Address: Operational Base + offset (0x0044)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	chroma_sec_ref_addr Base address for second reference chroma

VEPU_swreg17
Address: Operational Base + offset (0x0044)
encoder control register 1

Bit	Attr	Reset Value	Description
31:24	RW	0x00	pic_parameter_set_id H.264 pic_parameter_set_id
23:16	RW	0x00	intra_pred_mode H.264 intra prediction previous 4x4 mode favor
15:0	RW	0x0000	frame_num H.264 frame number

VEPU_swreg18
Address: Operational Base + offset (0x0048)
encoder control register 2

Bit	Attr	Reset Value	Description
31:30	RW	0x0	deblocking_filter_mode Deblocking filter mode. 2'd0=enabled. 2'd1=disabled (vp8=simple). 2'd2=disabled on slice
29:23	RW	0x00	h264_slice_size H.264 Slice size. mbRowPerSlice (mb rows) [0..127] 0=one slice per picture
22	RW	0x0	disable_quarter_pixmv H.264 Disable quarter pixel MVs. disableQuarterPixelMv
21	RW	0x0	transform8x8_mode_en H.264 Transform 8x8 enable. High Profile H.264. transform8x8Mode
20:19	RW	0x0	cabac_int_idc H.264 CABAC initial IDC. [0..2]
18	RW	0x0	entropy_coding_mode H.264 CABAC / VP8 boolenc enable. entropyCodingMode. 0=CAVLC (Baseline Profile H.264). 1=CABAC (Main Profile H.264)
17	RW	0x0	h264_inter4x4_mode H.264 Inter 4x4 mode restriction. restricted4x4Mode
16	RW	0x0	h264_stream_mode H.264 Stream mode. 0=NAL unit stream. 1=Byte stream
15:0	RW	0x0000	intra16x16_mode Intra prediction intra 16x16 mode favor

VEPU_swreg19
Address: Operational Base + offset (0x004c)
encoder control register 3

Bit	Attr	Reset Value	Description
31	RO	0x0	reserved
30	RW	0x0	mutimv_en Enable using more than 1 MV per macroblock.
29:20	RW	0x000	mv_penalty_1_4p Differential MV penalty for 1/4p ME. DMVPenaltyQp
19:10	RW	0x000	mv_penalty_4p Differential MV penalty for 4p ME. DMVPenalty4p
9:0	RW	0x000	mv_penalty_1p differential MV penalty for 1p

VEPU_swreg20
Address: Operational Base + offset (0x0050)
JPEG control regsiter

Bit	Attr	Reset Value	Description
31:30	RO	0x0	reserved
29:20	RW	0x000	mv_penalty_16x8_8x16 Penalty for using 16x8 or 8x16 MV.
19:10	RW	0x000	mv_penalty_8x8 Penalty for using 8x8 MV
9:0	RW	0x000	mv_penalty_8x4_4x8 Penalty for using 8x4 or 4x8 MV.

VEPU_swreg20_reuse
Address: Operational Base + offset (0x0050)
JPEG control regsiter

Bit	Attr	Reset Value	Description
31:26	RO	0x0	reserved
25	RW	0x0	jpeg_mode JPEG mode. 0=4:2:0 (4lum+2chr blocks/MCU). 1=4:2:2 (2lum+2chr blocks/MCU)
24	RW	0x0	jpeg_slic_en JPEG slice enable. 0=picture ends with EOI. 1=slice ends with RST
23:16	RW	0x00	jpeg_rst_mark_inter JPEG restart marker interval when slices are disabled (mb rows) [0..255]
15:0	RW	0x0000	jpeg_rst_mark_1 JPEG restart marker for first RST. incremented by HW for next RST

VEPU_swreg21
Address: Operational Base + offset (0x0054)
encoder control register 4

Bit	Attr	Reset Value	Description
31:24	RW	0x00	macroblock_penalty H.264 SKIP macroblock mode / VP8 zero/nearest/near mode penalty
23:16	RW	0x00	completed_slices H.264 amount of completed slices.
15:0	RW	0x0000	inter_mode inter MB mode favor in intra/inter selection

VEPU_swreg22
Address: Operational Base + offset (0x0058)

stream header remainder bits MSB

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	strm_hdr_rem stream header remainder bits MSB stream header remainder bits MSB

VEPU_swreg23
Address: Operational Base + offset (0x005c)
stream header remainder bits LSB

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	strm_hdr_rem2 stream header remainder bits LSB

VEPU_swreg24
Address: Operational Base + offset (0x0060)
stream buffer limit

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	strm_buffer_limit stream buffer limit

VEPU_swreg25
Address: Operational Base + offset (0x0064)
MAD control register

Bit	Attr	Reset Value	Description
31:28	RW	0x0	QP_adjust MAD based QP adjustment. madQpChange [-8..7]
27:22	RW	0x00	MAD_thredhold MAD threshold div256
21	RO	0x0	reserved
20:0	RW	0x000000	qp_sum_div2 QP sum div2 output

VEPU_swreg26
Address: Operational Base + offset (0x0068)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	intra_slice_bitmap Intra slice bitmap for slices 64..95. LSB=slice64. MSB=slice95. 1=intra.

VEPU_swreg26_vp8
Address: Operational Base + offset (0x0068)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	vp8_pro_update_cnt_addr Base address for VP8 counters for probability updates

VEPU_swreg27
Address: Operational Base + offset (0x006c)
QP register

Bit	Attr	Reset Value	Description
31:26	RW	0x00	qp_lum H.264 Initial QP. qpLum [0..51]
25:20	RW	0x00	qp_max H.264 Minimum QP. qpMax [0..51]
19:14	RW	0x00	qp_min H.264 Minimum QP. qpMin [0..51]
13	RO	0x0	reserved
12:0	RW	0x0000	checkpoint_distan checkpoint distance

VEPU_swreg27_vp8
Address: Operational Base + offset (0x006c)

Bit	Attr	Reset Value	Description
31	RO	0x0	reserved
30:23	RW	0x00	vp8_y1_round_dc VP8 qpY1RoundDc 8b
22:14	RW	0x000	vp8_y1_zbin_dc VP8 qpY1ZbinDc
13:0	RW	0x0000	vp8_y1_quant_dc VP8 qpY1QuantDc

VEPU_swreg28
Address: Operational Base + offset (0x0070)
checkpoint 1 and 2

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	checkp_1 checkpoint 1 word target/usage
15:0	RW	0x0000	checkp_2 checkpoint 2 word target/usage

VEPU_swreg28_vp8
Address: Operational Base + offset (0x0070)

Bit	Attr	Reset Value	Description
31	RO	0x0	reserved
30:23	RW	0x00	vp8_y1_round_ac VP8 qpY1RoundAc
22:14	RW	0x000	vp8_y1_zbin_ac VP8 qpY1ZbinAc

Bit	Attr	Reset Value	Description
13:0	RW	0x0000	vp8_y1_quant_ac VP8 qpY1QuantAc

VEPU_swreg29_vp8
Address: Operational Base + offset (0x0074)

Bit	Attr	Reset Value	Description
31	RO	0x0	reserved
30:23	RW	0x00	vp8_y2_round_dc VP8 qpY2RoundDc
22:14	RW	0x000	vp8_y2_zbin_dc VP8 qpY2ZbinDc
13:0	RW	0x0000	vp8_y2_quant_dc VP8 qpY2QuantDc

VEPU_swreg29
Address: Operational Base + offset (0x0074)
checkpoint 3 and 4

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	checkp_3 checkpoint 3 word target/usage
15:0	RW	0x0000	checkp_4 checkpoint 4 word target/usage

VEPU_swreg30_vp8
Address: Operational Base + offset (0x0078)

Bit	Attr	Reset Value	Description
31	RO	0x0	reserved
30:23	RW	0x00	vp8_y2_round_ac VP8 qpY2RoundAc
22:14	RW	0x000	vp8_y2_zbin_ac VP8 qpY2ZbinAc
13:0	RW	0x0000	vp8_y2_quant_ac VP8 qpY2QuantAc

VEPU_swreg30
Address: Operational Base + offset (0x0078)
checkpoint 5 and 6

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	checkp_5 checkpoint 5 word target/usage
15:0	RW	0x0000	checkp_6 checkpoint 6 word target/usage

VEPU_swreg31_vp8
Address: Operational Base + offset (0x007c)

Bit	Attr	Reset Value	Description
31	RO	0x0	reserved
30:23	RW	0x00	vp8_ch_round_dc VP8 qpChRoundDc
22:14	RW	0x000	vp8_ch_zbin_dc VP8 qpChZbinDc
13:0	RW	0x0000	vp8_ch_quant_dc VP8 qpChQuantDc

VEPU_swreg31
Address: Operational Base + offset (0x007c)
checkpoint 7 and 8

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	checkp_7 checkpoint 7 word target/usage
15:0	RW	0x0000	checkp_8 checkpoint 8 word target/usage

VEPU_swreg32
Address: Operational Base + offset (0x0080)
checkpoint 9 and 10

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	checkp_9 checkpoint 9 word target/usage
15:0	RW	0x0000	checkp_10 checkpoint 10 word target/usage

VEPU_swreg32_vp8
Address: Operational Base + offset (0x0080)

Bit	Attr	Reset Value	Description
31	RO	0x0	reserved
30:23	RW	0x00	vp8_ch_round_ac VP8 qpChRoundAc
22:14	RW	0x000	vp8_ch_zbin_ac VP8 qpChZbinAc
13:0	RW	0x0000	vp8_ch_quant_ac checkpoint distance VP8 qpChQuantAc

VEPU_swreg33
Address: Operational Base + offset (0x0084)

checkpoint word error 1 and 2

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	checkp_error1 checkpoint word error 1
15:0	RW	0x0000	checkp_error2 checkpoint word error 2

VEPU_swreg33_vp8
Address: Operational Base + offset (0x0084)

Bit	Attr	Reset Value	Description
31:28	RO	0x0	reserved
27:26	RW	0x0	vp8_1st_reffme_mvrefidx VP8 mvRefIdx for first reference frame. 2'd0=ipf. 2'd1=grf. 2'd2=arf.
25:17	RW	0x000	vp8_y2_dequant_dc VP8 qpY2DequantDc
16:8	RW	0x000	vp8_y1_dequant_ac VP8 qpY1DequantAc
7:0	RW	0x00	vp8_y1_dequant_dc VP8 qpY1DequantDc

VEPU_swreg34_vp8
Address: Operational Base + offset (0x0088)

Bit	Attr	Reset Value	Description
31	RO	0x0	reserved
30	RW	0x0	vp8_seg_update_en VP8 enable for segmentation map update. Map is different from previous frame and is written in stream.
29	RW	0x0	vp8_seg_en VP8 enable for segmentation. Segmentation map is stored in BaseVp8SegmentMap.
28	RW	0x0	vp8_2st_reffme_en VP8 enable for second reference frame.
27:26	RW	0x0	vp8_2st_reffme_mvrefidx VP8 mvRefIdx for second reference frame. 2'd0=ipf. 2'd1=grf. 2'd2=arf.
25:17	RW	0x000	vp8_ch_dequant_ac VP8 qpChDequantAc

Bit	Attr	Reset Value	Description
16:9	RW	0x00	vp8_ch_dequant_dc VP8 qpChDequantDc
8:0	RW	0x000	vp8_y2_dequant_ac checkpoint distance VP8 qpY2DequantAc

VEPU_swreg34
Address: Operational Base + offset (0x0088)
checkpoint word error 1 and 2

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	checkp_error3 checkpoint word error 3
15:0	RW	0x0000	checkp_error4 checkpoint word error 4

VEPU_swreg35
Address: Operational Base + offset (0x008c)
checkpoint word error 1 and 2

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	checkp_error5 checkpoint word error 5
15:0	RW	0x0000	checkp_error6 checkpoint word error 6

VEPU_swreg35_vp8
Address: Operational Base + offset (0x008c)
checkpoint word error 1 and 2

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	vp8_bool_value VP8 boolEncValue VP8 Penalty value for second reference frame zero-mv [0..255]

VEPU_swreg36
Address: Operational Base + offset (0x0090)
checkpoint delta QP register

Bit	Attr	Reset Value	Description
31:28	RO	0x0	reserved
27:24	RW	0x0	checkp_qp_1 checkpoint delta QP 1
23:20	RW	0x0	checkp_qp_2 checkpoint delta QP 2
19:16	RW	0x0	checkp_qp_3 checkpoint delta QP 3

Bit	Attr	Reset Value	Description
15:12	RW	0x0	checkp_qp_4 checkpoint delta QP 4
11:8	RW	0x0	checkp_qp_5 checkpoint delta QP 5
7:4	RW	0x0	checkp_qp_6 checkpoint delta QP 6
3:0	RW	0x0	checkp_qp_7 checkpoint delta QP 7

VEPU_swreg36_vp8
Address: Operational Base + offset (0x0090)
checkpoint delta QP register

Bit	Attr	Reset Value	Description
31:24	RW	0x00	vp8_2st_reffme_pena VP8 Penalty value for second reference frame zero-mv [0..255]
23:21	RW	0x0	vp8_dblk_fil_shps VP8 Deblocking filter sharpness [0..7]
20:15	RW	0x00	vp8_dblk_fil_lev VP8 Deblocking filter level [0..63]
14:13	RW	0x0	vp8_dct_ptton_cnt VP8 DCT partition count. 2'd0=1. 2'd1=2 [0..1]
12:8	RW	0x00	vp8_bool_bits_minus8 VP8 boolEncValueBitsMinus8 [0..23]
7:0	RW	0x00	vp8_bool_range VP8 boolEncRange [0..255]

VEPU_swreg37
Address: Operational Base + offset (0x0094)
rlc control

Bit	Attr	Reset Value	Description
31:29	RO	0x0	reserved
28:23	RW	0x00	stream_st_offset
22	RO	0x0	reserved
21:0	RW	0x000000	rlc_sum

VEPU_swreg38
Address: Operational Base + offset (0x0098)
mb control register

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	macroblock_count
15:0	RW	0x0000	mb_count_out

VEPU_swreg39
Address: Operational Base + offset (0x009c)
Base address for next pic

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	next_pic_addr Base address for next pic luminance

VEPU_swreg40
Address: Operational Base + offset (0x00a0)
Stabilization minimum value

Bit	Attr	Reset Value	Description
31:30	RW	0x0	stab_mode Stabilization mode. 2'd0=disabled. 2'd1=stab only. 2'd2=stab+encode
29:24	RO	0x0	reserved
23:0	RW	0x000000	stab_min_value Stabilization minimum value output. max 253*253*255

VEPU_swreg41
Address: Operational Base + offset (0x00a4)
Stabilization motion sum

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	stab_motion_sum Stabilization motion sum div8 output. max 53*253*255*1089/8

VEPU_swreg42
Address: Operational Base + offset (0x00a8)
stab_matrix1 and stab_gmv_hor

Bit	Attr	Reset Value	Description
31:26	RW	0x00	stab_gmv_hor Stabilization GMV horizontal output [-16..16]
25:24	RO	0x0	reserved
23:0	RW	0x000000	stab_matrix1 Stabilization matrix 1 (up-left position) output

VEPU_swreg43
Address: Operational Base + offset (0x00ac)
stab_matrix2 and stab_gmv_ver

Bit	Attr	Reset Value	Description
31:26	RW	0x00	stab_gmv_ver Stabilization GMV vertical output [-16..16]
25:24	RO	0x0	reserved

Bit	Attr	Reset Value	Description
23:0	RW	0x000000	stab_matrix2 Stabilization matrix 2 (up position) output

VEPU_swreg44
Address: Operational Base + offset (0x00b0)
stab_matrix3

Bit	Attr	Reset Value	Description
31:24	RO	0x0	reserved
23:0	RW	0x000000	stab_matrix3 Stabilization matrix 3 (up-right position) output

VEPU_swreg45
Address: Operational Base + offset (0x00b4)
stab_matrix4

Bit	Attr	Reset Value	Description
31:24	RO	0x0	reserved
23:0	RW	0x000000	stab_matrix4 Stabilization matrix 4 (left position) output

VEPU_swreg46
Address: Operational Base + offset (0x00b8)
stab_matrix5

Bit	Attr	Reset Value	Description
31:24	RO	0x0	reserved
23:0	RW	0x000000	stab_matrix5 Stabilization matrix 5 (GMV position) output

VEPU_swreg47
Address: Operational Base + offset (0x00bc)
stab_matrix6

Bit	Attr	Reset Value	Description
31:24	RO	0x0	reserved
23:0	RW	0x000000	stab_matrix6 Stabilization matrix 6 (right position) output

VEPU_swreg48
Address: Operational Base + offset (0x00c0)
stab_matrix7

Bit	Attr	Reset Value	Description
31:24	RO	0x0	reserved
23:0	RW	0x000000	stab_matrix7 Stabilization matrix 7 (down-left position) output

VEPU_swreg49

Address: Operational Base + offset (0x00c4)
stab_matrix8

Bit	Attr	Reset Value	Description
31:24	RO	0x0	reserved
23:0	RW	0x000000	stab_matrix8 Stabilization matrix 8 (down position) output

VEPU_swreg50
Address: Operational Base + offset (0x00c8)
stab_matrix9

Bit	Attr	Reset Value	Description
31:24	RO	0x0	reserved
23:0	RW	0x000000	stab_matrix9 Stabilization matrix 9 (down- right position) output

VEPU_swreg51
Address: Operational Base + offset (0x00cc)
cabac_table_addr

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	cabac_table_addr Base address for cabac context tables (H264) or probability tables (VP8)

VEPU_swreg52
Address: Operational Base + offset (0x00d0)
Base address for MV output

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	mv_out_addr Base address for MV output writing

VEPU_swreg53
Address: Operational Base + offset (0x00d4)
RGB to YUV conversion coefficien A and B

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	coeffB_RGB2YUV RGB to YUV conversion B RGB to YUV conversion coefficient B
15:0	RW	0x0000	coeffA_RGB2YUV RGB to YUV conversion coefficient A

VEPU_swreg54
Address: Operational Base + offset (0x00d8)
RGB to YUV conversion coefficien C and E

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	coeffE_RGB2YUV RGB to YUV conversion coefficient E

Bit	Attr	Reset Value	Description
15:0	RW	0x0000	coeffC_RGB2YUV RGB to YUV conversion coefficient C

VEPU_swreg55
Address: Operational Base + offset (0x00dc)
RGB mask MSB bit

Bit	Attr	Reset Value	Description
31	RO	0x0	reserved
30:26	RW	0x00	Bmask_MST RGB B-component mask MSB bit position [0..31]
25:21	RW	0x00	Gmask_MST RGB G-component mask MSB bit position [0..31]
20:16	RW	0x00	Rmask_MST RGB R-component mask MSB bit position [0..31]
15:0	RW	0x0000	coeffF_RGB2YUV RGB to YUV conversion coefficien F

VEPU_swreg56
Address: Operational Base + offset (0x00e0)
intra area control register

Bit	Attr	Reset Value	Description
31:24	RW	0x00	intra_left_mb Intra area left mb column (inside area) [0..255]
23:16	RW	0x00	intra_right_mb Intra area right mb column (inside area) [0..255]
15:8	RW	0x00	intra_top_mb Intra area top mb row (inside area) [0..255]
7:0	RW	0x00	intra_bot_mb Intra area bottom mb row (inside area) [0..255]

VEPU_swreg57
Address: Operational Base + offset (0x00e4)
CIR intra control reg

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	CIR_first_intra CIR first intra mb. 0=disabled [0..65535]
15:0	RW	0x0000	CIR_intra_mbinterval CIR intra mb interval. 0=disabled [0..65535]

VEPU_swreg58_vp8
Address: Operational Base + offset (0x00e8)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	vp8_1st_dct_pa_addr Base address for VP8 1st DCT partition

VEPU_swreg58
Address: Operational Base + offset (0x00e8)
Intra slice bitmap for slices 0..31

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	intra_slice_bitmap Intra slice bitmap for slices 0..31. LSB=slice0. MSB=slice31. 1=intra.

VEPU_swreg59_vp8
Address: Operational Base + offset (0x00ec)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	vp8_2st_dct_pa_addr Base address for VP8 2st DCT partition

VEPU_swreg59
Address: Operational Base + offset (0x00ec)
Intra slice bitmap for slices32..63

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	intra_slice_bitmap Intra slice bitmap for slices32..63. LSB=slice32. MSB=slice63 . 1=intra.

VEPU_swreg60
Address: Operational Base + offset (0x00f0)
1st ROI area register

Bit	Attr	Reset Value	Description
31:24	RW	0x00	ROI_1st_leftmb 1st ROI area left mb column (inside area) qp+=Roi1DeltaQp
23:16	RW	0x00	ROI_1st_rightmb 1st ROI area right mb column (outside area) qp-=Roi1DeltaQp
15:8	RW	0x00	ROI_1st_topmb 1st ROI area top mb row (inside area)
7:0	RW	0x00	ROI_1st_botmb 1st ROI area bottom mb row (outside area)

VEPU_swreg61
Address: Operational Base + offset (0x00f4)
Register0061 Abstract

Bit	Attr	Reset Value	Description
31:24	RW	0x00	ROI_2st_leftmb 2st ROI area left mb column (inside area) qp+=Roi1DeltaQp
23:16	RW	0x00	ROI_2st_rightmb 2st ROI area top mb row (inside area)
15:8	RW	0x00	ROI_2st_topmb 2st ROI area right mb column (outside area) qp-=Roi1DeltaQp
7:0	RW	0x00	ROI_2st_botmb 2st ROI area bottom mb row (outside area)

VEPU_swreg62
Address: Operational Base + offset (0x00f8)
MVC control reg

Bit	Attr	Reset Value	Description
31:28	RW	0x0	mv16x16_favor Zero 16x16 MV favor div2.
27:19	RW	0x000	penalty_4x4mv Penalty for using 4x4 MV.
18:16	RW	0x0	mvc_priority_id MVC priority_id [0..7]
15:13	RW	0x0	mvc_view_id MVC view_id [0..7]
12:10	RW	0x0	mvc_temporal_id MVC temporal_id [0..7]
9	RW	0x0	mvc_anchor_pic_flag MVC anchor_pic_flag. Specifies that the picture is part of an anchor access unit.
8	RW	0x0	mvc_inter_view_flag MVC inter_view_flag. Specifies that the picture is used for inter-view prediction.
7:4	RW	0x0	delta_qp_1st 1st ROI area delta QP. qp = Qp - Roi1DeltaQp [0..15]
3:0	RW	0x0	delta_qp_2st 2nd ROI area delta QP. qp = Qp - Roi2DeltaQp [0..15]

VEPU_swreg63
Address: Operational Base + offset (0x00fc)
Register0063 Abstract

Bit	Attr	Reset Value	Description
31	RO	0x0	reserved
30	RW	0x0	flag_tile_4x4 Tiled 4x4 input mode supported by HW. 0=not supported. 1=supported

Bit	Attr	Reset Value	Description
29	RW	0x0	search_area_h HW search area height. 0=5MB rows. 1=3MB rows
28	RW	0x0	flag_rgb2yuv_cov RGB to YUV conversion supported by HW. 0=not supported. 1=supported
27	RW	0x0	flag_h264_enc H.264 encoding supported by HW. 0=not supported. 1=supported
26	RW	0x0	flag_vp8_enc VP8 encoding supported by HW. 0=not supported. 1=supported
25	RW	0x0	flag_jpeg_enc JPEG encoding supported by HW. 0=not supported. 1=supported
24:16	RO	0x0	reserved
15:12	RW	0x0	bus_width Bus width of HW. 4'd0=32b. 4'd1=64b. 4'd2=128b
11:0	RO	0x780	MAX_VID_WIDTH Field0000 Description

VEPU_swreg64
Address: Operational Base + offset (0x0100)
JPEG luma quantization 1

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	jpeg_lum_qua1 JPEG luma quantization 1

VEPU_swreg64_vp8
Address: Operational Base + offset (0x0100)

Bit	Attr	Reset Value	Description
31:24	RO	0x0	reserved
23:12	RW	0x000	vp8_intra16_mode1 VP8 intra 16x16 mode 1 penalty
11:0	RW	0x000	vp8_intra16_mode0 VP8 intra 16x16 mode 0 penalty

VEPU_swreg65_vp8
Address: Operational Base + offset (0x0104)

Bit	Attr	Reset Value	Description
31:24	RO	0x0	reserved
23:12	RW	0x000	vp8_intra16_mode3 VP8 intra 16x16 mode 3 penalty
11:0	RW	0x000	vp8_intra16_mode2 VP8 intra 16x16 mode 2 penalty

VEPU_swreg65
Address: Operational Base + offset (0x0104)
JPEG luma quantization 2

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	jpeg_lum_qua2 JPEG luma quantization 2

VEPU_swreg66_vp8
Address: Operational Base + offset (0x0108)

Bit	Attr	Reset Value	Description
31:24	RO	0x0	reserved
23:12	RW	0x000	vp8_intra4_mode1 VP8 intra 4x4 mode 1 penalty
11:0	RW	0x000	vp8_intra4_mode0 VP8 intra 4x4 mode 0 penalty

VEPU_swreg66
Address: Operational Base + offset (0x0108)
JPEG luma quantization 3

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	jpeg_lum_qua3 JPEG luma quantization 3

VEPU_swreg67_vp8
Address: Operational Base + offset (0x010c)

Bit	Attr	Reset Value	Description
31:24	RO	0x0	reserved
23:12	RW	0x000	vp8_intra4_mode3 VP8 intra 4x4 mode 3 penalty
11:0	RW	0x000	vp8_intra4_mode2 VP8 intra 4x4 mode 2 penalty

VEPU_swreg67
Address: Operational Base + offset (0x010c)
JPEG luma quantization 4

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	jpeg_lum_qua4 JPEG luma quantization 4

VEPU_swreg68_vp8
Address: Operational Base + offset (0x0110)

Bit	Attr	Reset Value	Description
31:24	RO	0x0	reserved
23:12	RW	0x000	vp8_intra4_mode5 VP8 intra 4x4 mode 5 penalty
11:0	RW	0x000	vp8_intra4_mode4 VP8 intra 4x4 mode 4 penalty

VEPU_swreg68
Address: Operational Base + offset (0x0110)
JPEG luma quantization 5

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	jpeg_lum_qua5 JPEG luma quantization 5

VEPU_swreg69_vp8
Address: Operational Base + offset (0x0114)

Bit	Attr	Reset Value	Description
31:24	RO	0x0	reserved
23:12	RW	0x000	vp8_intra4_mode7 VP8 intra 4x4 mode 7 penalty
11:0	RW	0x000	vp8_intra4_mode6 VP8 intra 4x4 mode 6 penalty

VEPU_swreg69
Address: Operational Base + offset (0x0114)
JPEG luma quantization 6

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	jpeg_lum_qua6 JPEG luma quantization 6 JPEG luma quantization 6

VEPU_swreg70_vp8
Address: Operational Base + offset (0x0118)

Bit	Attr	Reset Value	Description
31:24	RO	0x0	reserved
23:12	RW	0x000	vp8_intra4_mode9 VP8 intra 4x4 mode 9 penalty
11:0	RW	0x000	vp8_intra4_mode8 VP8 intra 4x4 mode 8 penalty

VEPU_swreg70
Address: Operational Base + offset (0x0118)
JPEG luma quantization 7

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	jpeg_lum_qua7 JPEG luma quantization 7

VEPU_swreg71_vp8
Address: Operational Base + offset (0x011c)
JPEG luma quantization 7

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	vp8_seg_addr Base address for VP8 segmentation map, segmentId 2-bits / macroblock

VEPU_swreg71
Address: Operational Base + offset (0x011c)
JPEG luma quantization 8

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	jpeg_lum_qua8 JPEG luma quantization 8

VEPU_swreg72
Address: Operational Base + offset (0x0120)
JPEG luma quantization 9

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	jpeg_lum_qua9 JPEG luma quantization 9

VEPU_swreg72_vp8
Address: Operational Base + offset (0x0120)

Bit	Attr	Reset Value	Description
31	RO	0x0	reserved
30:23	RW	0x00	vp8_seg1_y1_round_dc VP8 segment1 qpY1RoundDc
22:14	RW	0x000	vp8_seg1_y1_zbin_dc VP8 segment1 qpY1ZbinDc

Bit	Attr	Reset Value	Description
13:0	RW	0x0000	vp8_seg1_y1_quant_dc VP8 segment1 qpY1QuantDc

VEPU_swreg73
Address: Operational Base + offset (0x0124)
Register0073 Abstract

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	jpeg_lum_qua10 JPEG luma quantization 10 JPEG luma quantization 10

VEPU_swreg73_vp8
Address: Operational Base + offset (0x0124)

Bit	Attr	Reset Value	Description
31	RO	0x0	reserved
30:23	RW	0x00	vp8_seg1_y1_round_ac VP8 segment1 qpY1RoundAc
22:14	RW	0x000	vp8_seg1_y1_zbin_ac VP8 segment1 qpY1ZbinAc
13:0	RW	0x0000	vp8_seg1_y1_quant_ac VP8 segment1 qpY1QuantAc

VEPU_swreg74_vp8
Address: Operational Base + offset (0x0128)

Bit	Attr	Reset Value	Description
31	RO	0x0	reserved
30:23	RW	0x00	vp8_seg1_y2_round_dc VP8 segment1 qpY2RoundDc
22:14	RW	0x000	vp8_seg1_y2_zbin_dc VP8 segment1 qpY2ZbinDc
13:0	RW	0x0000	vp8_seg1_y2_quant_dc VP8 segment1 qpY2QuantDc

VEPU_swreg74
Address: Operational Base + offset (0x0128)
JPEG luma quantization 11

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	jpeg_lum_qua11 JPEG luma quantization 11

VEPU_swreg75
Address: Operational Base + offset (0x012c)

JPEG luma quantization 12

Bit	Attr	Reset Value	Description
31:1	RO	0x0	reserved
0	RW	0x0	jpeg_lum_qua12 JPEG luma quantization 12

VEPU_swreg75_vp8
Address: Operational Base + offset (0x012c)

Bit	Attr	Reset Value	Description
31	RO	0x0	reserved
30:23	RW	0x00	vp8_seg1_y1_round_ac VP8 segment1 qpY2RoundAc
22:14	RW	0x000	vp8_seg1_y2_zbin_ac VP8 segment1 qpY2ZbinAc
13:0	RW	0x0000	vp8_seg1_y2_quant_ac VP8 segment1 qpY2QuantAc

VEPU_swreg76_vp8
Address: Operational Base + offset (0x0130)

Bit	Attr	Reset Value	Description
31	RO	0x0	reserved
30:23	RW	0x00	vp8_seg1_ch_round_dc VP8 segment1 qpchRoundDc
22:14	RW	0x000	vp8_seg1_ch_zbin_dc VP8 segment1 qpchZbinDc
13:0	RW	0x0000	vp8_seg1_ch_quant_dc VP8 segment1 qpchQuantDc

VEPU_swreg76
Address: Operational Base + offset (0x0130)
JPEG luma quantization 13

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	jpeg_lum_qua13 JPEG luma quantization 13

VEPU_swreg77_vp8
Address: Operational Base + offset (0x0134)

Bit	Attr	Reset Value	Description
31	RO	0x0	reserved
30:23	RW	0x00	vp8_seg1_ch_round_ac VP8 segment1 qpchRoundAc
22:14	RW	0x000	vp8_seg1_ch_zbin_ac VP8 segment1 qpchZbinAc

Bit	Attr	Reset Value	Description
13:0	RW	0x0000	vp8_seg1_ch_quant_ac VP8 segment1 qpchQuantAc

VEPU_swreg77
Address: Operational Base + offset (0x0134)
JPEG luma quantization 14

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	jpeg_lum_qua14 JPEG luma quantization 14

VEPU_swreg78_vp8
Address: Operational Base + offset (0x0138)

Bit	Attr	Reset Value	Description
31:26	RO	0x0	reserved
25:17	RW	0x000	vp8_seg1_y2_dequant_dc VP8 segment1 qpY2DequantDc
16:8	RW	0x000	vp8_seg1_y1_dequant_ac VP8 segment1 qpY1DequantAc
7:0	RW	0x00	vp8_seg1_y1_dequant_dc VP8 segment1 qpY1DequantDc

VEPU_swreg78
Address: Operational Base + offset (0x0138)
JPEG luma quantization 15

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	jpeg_lum_qua15 JPEG luma quantization 15

VEPU_swreg79
Address: Operational Base + offset (0x013c)
JPEG luma quantization 16

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	jpeg_lum_qua16 JPEG luma quantization 16

VEPU_swreg79_vp8
Address: Operational Base + offset (0x013c)

Bit	Attr	Reset Value	Description
31:26	RW	0x00	vp8_seg1_fil_lev VP8 segment1 filter level
25:17	RW	0x000	vp8_seg1_ch_dequant_ac VP8 segment1 qpChDequantAc

Bit	Attr	Reset Value	Description
16:9	RW	0x00	vp8_seg1_ch_dequant_dc VP8 segment1 qpChDequantDc
8:0	RW	0x000	vp8_seg1_y2_dequant_ac VP8 segment1 qpY2DequantAc

VEPU_swreg80_vp8
Address: Operational Base + offset (0x0140)

Bit	Attr	Reset Value	Description
31	RO	0x0	reserved
30:23	RW	0x00	vp8_seg2_y1_round_dc VP8 segment2 qpY1RoundDc
22:14	RW	0x000	vp8_seg2_y1_zbin_dc VP8 segment2 qpY1ZbinDc
13:0	RW	0x0000	vp8_seg2_y1_quant_dc VP8 segment2 qpY1QuantDc

VEPU_swreg80
Address: Operational Base + offset (0x0140)
JPEG chroma quantization 1

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	jpeg_chrom_qua1 JPEG chroma quantization 1

VEPU_swreg81
Address: Operational Base + offset (0x0144)
JPEG chroma quantization 2

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	jpeg_chrom_qua2 JPEG chroma quantization 2

VEPU_swreg81_vp8
Address: Operational Base + offset (0x0144)

Bit	Attr	Reset Value	Description
31	RO	0x0	reserved
30:23	RW	0x00	vp8_seg2_y1_round_ac VP8 segment2 qpY1RoundAc
22:14	RW	0x000	vp8_seg2_y1_zbin_ac VP8 segment2 qpY1ZbinAc
13:0	RW	0x0000	vp8_seg2_y1_quant_ac VP8 segment2 qpY1QuantAc

VEPU_swreg82_vp8

Address: Operational Base + offset (0x0148)

Bit	Attr	Reset Value	Description
31	RO	0x0	reserved
30:23	RW	0x00	vp8_seg2_y2_round_dc VP8 segment2 qpY2RoundDc
22:14	RW	0x000	vp8_seg2_y2_zbin_dc VP8 segment2 qpY2ZbinDc
13:0	RW	0x0000	vp8_seg2_y2_quant_dc VP8 segment2 qpY2QuantDc

VEPU_swreg82
Address: Operational Base + offset (0x0148)
JPEG chroma quantization 3

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	jpeg_chrom_qua3 JPEG chroma quantization 3

VEPU_swreg83
Address: Operational Base + offset (0x014c)
JPEG chroma quantization 4

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	jpeg_chrom_qua4 JPEG chroma quantization 4

VEPU_swreg83_vp8
Address: Operational Base + offset (0x014c)

Bit	Attr	Reset Value	Description
31	RO	0x0	reserved
30:23	RW	0x00	vp8_seg2_y1_round_ac VP8 segment2 qpY2RoundAc
22:14	RW	0x000	vp8_seg2_y2_zbin_ac VP8 segment2 qpY2ZbinAc
13:0	RW	0x0000	vp8_seg2_y2_quant_ac VP8 segment2 qpY2QuantAc

VEPU_swreg84_vp8
Address: Operational Base + offset (0x0150)

Bit	Attr	Reset Value	Description
31	RO	0x0	reserved
30:23	RW	0x00	vp8_seg2_ch_round_dc VP8 segment2 qpchRoundDc
22:14	RW	0x000	vp8_seg2_ch_zbin_dc VP8 segment2 qpchZbinDc

Bit	Attr	Reset Value	Description
13:0	RW	0x0000	vp8_seg2_ch_quant_dc VP8 segment2 qpchQuantDc

VEPU_swreg84
Address: Operational Base + offset (0x0150)
JPEG chroma quantization 5

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	jpeg_chrom_qua5 JPEG chroma quantization 5

VEPU_swreg85
Address: Operational Base + offset (0x0154)
JPEG chroma quantization 6

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	jpeg_chrom_qua6 JPEG chroma quantization 6

VEPU_swreg85_vp8
Address: Operational Base + offset (0x0154)

Bit	Attr	Reset Value	Description
31	RO	0x0	reserved
30:23	RW	0x00	vp8_seg2_ch_round_ac VP8 segment2 qpchRoundAc
22:14	RW	0x000	vp8_seg2_ch_zbin_ac VP8 segment2 qpchZbinAc
13:0	RW	0x0000	vp8_seg2_ch_quant_ac VP8 segment2 qpchQuantAc

VEPU_swreg86_vp8
Address: Operational Base + offset (0x0158)

Bit	Attr	Reset Value	Description
31:26	RO	0x0	reserved
25:17	RW	0x000	vp8_seg2_y2_dequant_dc VP8 segment2 qpY2DequantDc
16:8	RW	0x000	vp8_seg2_y1_dequant_ac VP8 segment2 qpY1DequantAc
7:0	RW	0x00	vp8_seg2_y1_dequant_dc VP8 segment2 qpY2DequantDc

VEPU_swreg86
Address: Operational Base + offset (0x0158)
JPEG chroma quantization 7

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	jpeg_chrom_qua7 JPEG chroma quantization 7

VEPU_swreg87
Address: Operational Base + offset (0x015c)
JPEG chroma quantization 8

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	jpeg_chrom_qua8 JPEG chroma quantization 8

VEPU_swreg87_vp8
Address: Operational Base + offset (0x015c)

Bit	Attr	Reset Value	Description
31:26	RW	0x00	vp8_seg2_fil_lev VP8 segment2 filter level
25:17	RW	0x000	vp8_seg2_ch_dequant_ac VP8 segment2 qpChDequantAc
16:9	RW	0x00	vp8_seg2_ch_dequant_dc VP8 segment2 qpChDequantDc
8:0	RW	0x000	vp8_seg2_y2_dequant_ac VP8 segment2qpY2DequantAc

VEPU_swreg88_vp8
Address: Operational Base + offset (0x0160)

Bit	Attr	Reset Value	Description
31	RO	0x0	reserved
30:23	RW	0x00	vp8_seg3_y1_round_dc VP8 segment3 qpY1RoundDc
22:14	RW	0x000	vp8_seg3_y1_zbin_dc VP8 segment3 qpY1ZbinDc
13:0	RW	0x0000	vp8_seg3_y1_quant_dc VP8 segment3 qpY1QuantDc

VEPU_swreg88
Address: Operational Base + offset (0x0160)
JPEG chroma quantization 9

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	jpeg_chrom_qua9 JPEG chroma quantization 9

VEPU_swreg89_vp8
Address: Operational Base + offset (0x0164)

Bit	Attr	Reset Value	Description
31	RO	0x0	reserved
30:23	RW	0x00	vp8_seg3_y1_round_ac VP8 segment3 qpY1RoundAc
22:14	RW	0x000	vp8_seg3_y1_zbin_ac VP8 segment3 qpY1ZbinAc
13:0	RW	0x0000	vp8_seg3_y1_quant_ac VP8 segment3 qpY1QuantAc

VEPU_swreg89
Address: Operational Base + offset (0x0164)
JPEG chroma quantization 10

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	jpeg_chrom_qua10 JPEG chroma quantization 10

VEPU_swreg90_vp8
Address: Operational Base + offset (0x0168)

Bit	Attr	Reset Value	Description
31	RO	0x0	reserved
30:23	RW	0x00	vp8_seg3_y2_round_dc VP8 segment3 qpY2RoundDc
22:14	RW	0x000	vp8_seg3_y2_zbin_dc VP8 segment3 qpY2ZbinDc
13:0	RW	0x0000	vp8_seg3_y2_quant_dc VP8 segment3 qpY2QuantDc

VEPU_swreg90
Address: Operational Base + offset (0x0168)
JPEG chroma quantization 11

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	jpeg_chrom_qua11 JPEG chroma quantization 11

VEPU_swreg91_vp8
Address: Operational Base + offset (0x016c)

Bit	Attr	Reset Value	Description
31	RO	0x0	reserved
30:23	RW	0x00	vp8_seg3_y1_round_ac VP8 segment3 qpY2RoundAc
22:14	RW	0x000	vp8_seg3_y2_zbin_ac VP8 segment3 qpY2ZbinAc

Bit	Attr	Reset Value	Description
13:0	RW	0x0000	vp8_seg3_y2_quant_ac VP8 segment3 qpY2QuantAc

VEPU_swreg91
Address: Operational Base + offset (0x016c)
JPEG chroma quantization 12

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	jpeg_chrom_qua12 JPEG chroma quantization 12

VEPU_swreg92_vp8
Address: Operational Base + offset (0x0170)

Bit	Attr	Reset Value	Description
31	RO	0x0	reserved
30:23	RW	0x00	vp8_seg3_ch_round_dc VP8 segment3 qpchRoundDc
22:14	RW	0x000	vp8_seg3_ch_zbin_dc VP8 segment3 qpchZbinDc
13:0	RW	0x0000	vp8_seg3_ch_quant_dc VP8 segment3 qpchQuantDc

VEPU_swreg92
Address: Operational Base + offset (0x0170)
JPEG chroma quantization 13

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	jpeg_chrom_qua13 JPEG chroma quantization 13

VEPU_swreg93_vp8
Address: Operational Base + offset (0x0174)

Bit	Attr	Reset Value	Description
31	RO	0x0	reserved
30:23	RW	0x00	vp8_seg3_ch_round_ac VP8 segment3 qpchRoundAc
22:14	RW	0x000	vp8_seg3_ch_zbin_ac VP8 segment3 qpchZbinAc
13:0	RW	0x0000	vp8_seg3_ch_quant_ac VP8 segment3 qpchQuantAc

VEPU_swreg93
Address: Operational Base + offset (0x0174)
JPEG chroma quantization 14

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	jpeg_chrom_qua14 JPEG chroma quantization 14

VEPU_swreg94_vp8
Address: Operational Base + offset (0x0178)

Bit	Attr	Reset Value	Description
31:26	RO	0x0	reserved
25:17	RW	0x000	vp8_seg3_y2_dequant_dc VP8 segment3 qpY2DequantDc
16:8	RW	0x000	vp8_seg3_y1_dequant_ac VP8 segment3 qpY1DequantAc
7:0	RW	0x00	vp8_seg3_y1_dequant_dc VP8 segment3 qpY2DequantDc

VEPU_swreg94
Address: Operational Base + offset (0x0178)
JPEG chroma quantization 15

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	jpeg_chrom_qua15 JPEG chroma quantization 15

VEPU_swreg95_vp8
Address: Operational Base + offset (0x017c)

Bit	Attr	Reset Value	Description
31:26	RW	0x00	vp8_seg3_fil_lev VP8 segment3 filter level
25:17	RW	0x000	vp8_seg3_ch_dequant_ac VP8 segment3 qpChDequantAc
16:9	RW	0x00	vp8_seg3_ch_dequant_dc VP8 segment3 qpChDequantDc
8:0	RW	0x000	vp8_seg3_y2_dequant_ac VP8 segment3qp Y2DequantAc

VEPU_swreg95
Address: Operational Base + offset (0x017c)
JPEG chroma quantization 16

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	jpeg_chrom_qua16 JPEG chroma quantization 16

VEPU_swreg96
Address: Operational Base + offset (0x0180)

DMV 4p/1p penalty values 0-3

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	DMV_4p_1p_penalty DMV 4p/1p penalty values 0-3

VEPU_swreg97
Address: Operational Base + offset (0x0184)
DMV 4p/1p penalty values 4-7

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	DMV_4p_1p_penalty DMV 4p/1p penalty values 4-7

VEPU_swreg98
Address: Operational Base + offset (0x0188)
DMV 4p/1p penalty values

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	DMV_4p_1p_penalty DMV 4p/1p penalty values

VEPU_swreg99
Address: Operational Base + offset (0x018c)
DMV 4p/1p penalty values

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	DMV_4p_1p_penalty DMV 4p/1p penalty values

VEPU_swreg100
Address: Operational Base + offset (0x0190)
DMV 4p/1p penalty values

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	DMV_4p_1p_penalty DMV 4p/1p penalty values

VEPU_swreg101
Address: Operational Base + offset (0x0194)
DMV 4p/1p penalty values

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	DMV_4p_1p_penalty DMV 4p/1p penalty values

VEPU_swreg102
Address: Operational Base + offset (0x0198)
DMV 4p/1p penalty values

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	DMV_4p_1p_penalty DMV 4p/1p penalty values

VEPU_swreg103
Address: Operational Base + offset (0x019c)
DMV 4p/1p penalty values

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	DMV_4p_1p_penalty DMV 4p/1p penalty values

VEPU_swreg104
Address: Operational Base + offset (0x01a0)
DMV 4p/1p penalty values

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	DMV_4p_1p_penalty DMV 4p/1p penalty values

VEPU_swreg105
Address: Operational Base + offset (0x01a4)
DMV 4p/1p penalty values

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	DMV_4p_1p_penalty DMV 4p/1p penalty values

VEPU_swreg106
Address: Operational Base + offset (0x01a8)
DMV 4p/1p penalty values

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	DMV_4p_1p_penalty DMV 4p/1p penalty values

VEPU_swreg107
Address: Operational Base + offset (0x01ac)
DMV 4p/1p penalty values

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	DMV_4p_1p_penalty DMV 4p/1p penalty values

VEPU_swreg108
Address: Operational Base + offset (0x01b0)
DMV 4p/1p penalty values

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	DMV_4p_1p_penalty DMV 4p/1p penalty values

VEPU_swreg109
Address: Operational Base + offset (0x01b4)

DMV 4p/1p penalty values

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	DMV_4p_1p_penalty DMV 4p/1p penalty values

VEPU_swreg110
Address: Operational Base + offset (0x01b8)
DMV 4p/1p penalty values

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	DMV_4p_1p_penalty DMV 4p/1p penalty values

VEPU_swreg111
Address: Operational Base + offset (0x01bc)
DMV 4p/1p penalty values

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	DMV_4p_1p_penalty DMV 4p/1p penalty values

VEPU_swreg112
Address: Operational Base + offset (0x01c0)
DMV 4p/1p penalty values

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	DMV_4p_1p_penalty DMV 4p/1p penalty values

VEPU_swreg113
Address: Operational Base + offset (0x01c4)
DMV 4p/1p penalty values

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	DMV_4p_1p_penalty DMV 4p/1p penalty values

VEPU_swreg114
Address: Operational Base + offset (0x01c8)
DMV 4p/1p penalty values

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	DMV_4p_1p_penalty DMV 4p/1p penalty values

VEPU_swreg115
Address: Operational Base + offset (0x01cc)
DMV 4p/1p penalty values

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	DMV_4p_1p_penalty DMV 4p/1p penalty values

VEPU_swreg116
Address: Operational Base + offset (0x01d0)
DMV 4p/1p penalty values

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	DMV_4p_1p_penalty DMV 4p/1p penalty values

VEPU_swreg117
Address: Operational Base + offset (0x01d4)
DMV 4p/1p penalty values

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	DMV_4p_1p_penalty DMV 4p/1p penalty values

VEPU_swreg118
Address: Operational Base + offset (0x01d8)
DMV 4p/1p penalty values

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	DMV_4p_1p_penalty DMV 4p/1p penalty values

VEPU_swreg119
Address: Operational Base + offset (0x01dc)
DMV 4p/1p penalty values

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	DMV_4p_1p_penalty DMV 4p/1p penalty values

VEPU_swreg120
Address: Operational Base + offset (0x01e0)
DMV 4p/1p penalty values

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	DMV_4p_1p_penalty DMV 4p/1p penalty values

VEPU_swreg121
Address: Operational Base + offset (0x01e4)
DMV 4p/1p penalty values

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	DMV_4p_1p_penalty DMV 4p/1p penalty values

VEPU_swreg122
Address: Operational Base + offset (0x01e8)

DMV 4p/1p penalty values

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	DMV_4p_1p_penalty DMV 4p/1p penalty values

VEPU_swreg123
Address: Operational Base + offset (0x01ec)
DMV 4p/1p penalty values

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	DMV_4p_1p_penalty DMV 4p/1p penalty values

VEPU_swreg124
Address: Operational Base + offset (0x01f0)
DMV 4p/1p penalty values

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	DMV_4p_1p_penalty DMV 4p/1p penalty values

VEPU_swreg125
Address: Operational Base + offset (0x01f4)
DMV 4p/1p penalty values

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	DMV_4p_1p_penalty DMV 4p/1p penalty values

VEPU_swreg126
Address: Operational Base + offset (0x01f8)
DMV 4p/1p penalty values

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	DMV_4p_1p_penalty DMV 4p/1p penalty values

VEPU_swreg127
Address: Operational Base + offset (0x01fc)
DMV 4p/1p penalty values

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	DMV_4p_1p_penalty DMV 4p/1p penalty values

VEPU_swreg128
Address: Operational Base + offset (0x0200)
DMV 4p/1p penalty values

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	DMV_4p_1p_penalty DMV 4p/1p penalty values

VEPU_swreg129
Address: Operational Base + offset (0x0204)
DMV 4p/1p penalty values

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	DMV_4p_1p_penalty DMV 4p/1p penalty values

VEPU_swreg130
Address: Operational Base + offset (0x0208)
DMV 4p/1p penalty values

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	DMV_4p_1p_penalty DMV 4p/1p penalty values

VEPU_swreg131
Address: Operational Base + offset (0x020c)
DMV 4p/1p penalty values

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	DMV_4p_1p_penalty DMV 4p/1p penalty values

VEPU_swreg132
Address: Operational Base + offset (0x0210)
DMV 4p/1p penalty values

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	DMV_4p_1p_penalty DMV 4p/1p penalty values

VEPU_swreg133
Address: Operational Base + offset (0x0214)
DMV 4p/1p penalty values

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	DMV_4p_1p_penalty DMV 4p/1p penalty values

VEPU_swreg134
Address: Operational Base + offset (0x0218)
DMV 4p/1p penalty values

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	DMV_4p_1p_penalty DMV 4p/1p penalty values

VEPU_swreg135
Address: Operational Base + offset (0x021c)

DMV 4p/1p penalty values

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	DMV_4p_1p_penalty DMV 4p/1p penalty values

VEPU_swreg136
Address: Operational Base + offset (0x0220)
DMV 4p/1p penalty values

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	DMV_4p_1p_penalty DMV 4p/1p penalty values

VEPU_swreg137
Address: Operational Base + offset (0x0224)
DMV 4p/1p penalty values

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	DMV_4p_1p_penalty DMV 4p/1p penalty values

VEPU_swreg138
Address: Operational Base + offset (0x0228)
DMV 4p/1p penalty values

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	DMV_4p_1p_penalty DMV 4p/1p penalty values

VEPU_swreg139
Address: Operational Base + offset (0x022c)
DMV 4p/1p penalty values

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	DMV_4p_1p_penalty DMV 4p/1p penalty values

VEPU_swreg140
Address: Operational Base + offset (0x0230)
DMV 4p/1p penalty values

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	DMV_4p_1p_penalty DMV 4p/1p penalty values

VEPU_swreg141
Address: Operational Base + offset (0x0234)
DMV 4p/1p penalty values

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	DMV_4p_1p_penalty DMV 4p/1p penalty values

VEPU_swreg142
Address: Operational Base + offset (0x0238)
DMV 4p/1p penalty values

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	DMV_4p_1p_penalty DMV 4p/1p penalty values

VEPU_swreg143
Address: Operational Base + offset (0x023c)
DMV 4p/1p penalty values

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	DMV_4p_1p_penalty DMV 4p/1p penalty values

VEPU_swreg144
Address: Operational Base + offset (0x0240)
DMV 4p/1p penalty values

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	DMV_4p_1p_penalty DMV 4p/1p penalty values

VEPU_swreg145
Address: Operational Base + offset (0x0244)
DMV 4p/1p penalty values

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	DMV_4p_1p_penalty DMV 4p/1p penalty values

VEPU_swreg146
Address: Operational Base + offset (0x0248)
DMV 4p/1p penalty values

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	DMV_4p_1p_penalty DMV 4p/1p penalty values

VEPU_swreg147
Address: Operational Base + offset (0x024c)
DMV 4p/1p penalty values

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	DMV_4p_1p_penalty DMV 4p/1p penalty values

VEPU_swreg148
Address: Operational Base + offset (0x0250)

DMV 4p/1p penalty values

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	DMV_4p_1p_penalty DMV 4p/1p penalty values

VEPU_swreg149
Address: Operational Base + offset (0x0254)
DMV 4p/1p penalty values

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	DMV_4p_1p_penalty DMV 4p/1p penalty values

VEPU_swreg150
Address: Operational Base + offset (0x0258)
DMV 4p/1p penalty values

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	DMV_4p_1p_penalty DMV 4p/1p penalty values

VEPU_swreg151
Address: Operational Base + offset (0x025c)
DMV 4p/1p penalty values

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	DMV_4p_1p_penalty DMV 4p/1p penalty values

VEPU_swreg152
Address: Operational Base + offset (0x0260)
DMV 4p/1p penalty values

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	DMV_4p_1p_penalty DMV 4p/1p penalty values

VEPU_swreg153
Address: Operational Base + offset (0x0264)
DMV 4p/1p penalty values

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	DMV_4p_1p_penalty DMV 4p/1p penalty values

VEPU_swreg154
Address: Operational Base + offset (0x0268)
DMV 4p/1p penalty values

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	DMV_4p_1p_penalty DMV 4p/1p penalty values

VEPU_swreg155
Address: Operational Base + offset (0x026c)
DMV 4p/1p penalty values

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	DMV_4p_1p_penalty DMV 4p/1p penalty values

VEPU_swreg156
Address: Operational Base + offset (0x0270)
DMV 4p/1p penalty values

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	DMV_4p_1p_penalty DMV 4p/1p penalty values

VEPU_swreg157
Address: Operational Base + offset (0x0274)
DMV 4p/1p penalty values

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	DMV_4p_1p_penalty DMV 4p/1p penalty values

VEPU_swreg158
Address: Operational Base + offset (0x0278)
DMV 4p/1p penalty values

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	DMV_4p_1p_penalty DMV 4p/1p penalty values

VEPU_swreg159
Address: Operational Base + offset (0x027c)
DMV 4p/1p penalty values

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	DMV_4p_1p_penalty DMV 4p/1p penalty values 124-127

VEPU_swreg160
Address: Operational Base + offset (0x0280)
vp8 control

Bit	Attr	Reset Value	Description
31:24	RO	0x0	reserved
23:12	RW	0x000	vp8_coeff_dmv_penalty VP8 coeff for dmv penalty for intra/inter selection
11:0	RW	0x000	vp8_inter_type VP8 bit cost of inter type

VEPU_swreg161
Address: Operational Base + offset (0x0284)
VP8 bit cost of golden ref frame

Bit	Attr	Reset Value	Description
31:12	RO	0x0	reserved
11:0	RW	0x000	vp8_ref_frame VP8 bit cost of golden ref frame

VEPU_swreg162
Address: Operational Base + offset (0x0288)
vp8 loop filter delta registers

Bit	Attr	Reset Value	Description
31:28	RO	0x0	reserved
27:21	RW	0x00	vp8_loopfilter_altref VP8 loop filter delta for alt ref
20:14	RW	0x00	vp8_loopfilter_goldenref VP8 loop filter delta for golden ref
13:7	RW	0x00	vp8_loopfilter_lastref VP8 loop filter delta for last ref
6:0	RW	0x00	vp8_loopfilter_intra VP8 loop filter delta for intra mb

VEPU_swreg163
Address: Operational Base + offset (0x028c)
vp8 loop filter delta register

Bit	Attr	Reset Value	Description
31:28	RO	0x0	reserved
27:21	RW	0x00	vp8_loopfilter_splitmv VP8 loop filter delta for SPLITMV
20:14	RW	0x00	vp8_loopfilter_newmv VP8 loop filter delta for NEWMV
13:7	RW	0x00	vp8_loopfilter_zeromv VP8 loop filter delta for ZEROMV
6:0	RW	0x00	vp8_loopfilter_bpred VP8 loop filter delta for BPRED

2.4.9 VDPU Register Summary

Name	Offset	Size	Reset Value	Description
VDPU_SWREG0	0x0000	W	0x67313688	ID register(read only)
VDPU_SWREG1	0x0004	W	0x00000000	interrupt register decoder
VDPU_SWREG2	0x0008	W	0x01000400	device configuration register decoder
VDPU_SWREG3	0x000c	W	0x00000001	Device control register 0(decmode, picture type etc)

Name	Offset	Size	Reset Value	Description
VDPU_SWREG4_H264	0x0010	W	0x00000000	decoder control register 1(picture parameters)
VDPU_SWREG4	0x0010	W	0x00000000	decoder control register 1(picture parameters)
VDPU_SWREG5	0x0014	W	0x00000000	decoder control register2 (stream decoding table selects)
VDPU_SWREG5_H264	0x0014	W	0x00000000	decoder control register2 (stream decoding table selects)
VDPU_SWREG5_VP8	0x0014	W	0x00000000	decoder control register2 (stream decoding table selects)
VDPU_SWREG6	0x0018	W	0x00000000	decoder control register 3(stream buffer information)
VDPU_SWREG7	0x001c	W	0x00000000	decoder control register 4(H264, VC-1 control)
VDPU_SWREG7_VP8	0x001c	W	0x00000000	decoder control register 4(H264, VC-1 control)
VDPU_SWREG8	0x0020	W	0x00000000	decoder control register 5(H264, VC-1 and RV control)
VDPU_SWREG8_VP8	0x0020	W	0x00000000	decoder control register 5(H264, VC-1 and RV control)
VDPU_SWREG9	0x0024	W	0x00000000	decoder control register 6
VDPU_SWREG9_VP8	0x0024	W	0x00000000	decoder control register 6
VDPU_SREG10_H264_RLC	0x0028	W	0x00000000	Base address for differential motion vector base address
VDPU_SREG10_H264	0x0028	W	0x00000000	Base address for differential motion vector base address
VDPU_SREG10_VP8	0x0028	W	0x00000000	Base address for differential motion vector base address
VDPU_SWREG11_H264_RLC	0x002c	W	0x00000000	decoder control register 7
VDPU_SWREG11_H264	0x002c	W	0x00000000	decoder control register 7
VDPU_SWREG11_VP8	0x002c	W	0x00000000	
VDPU_SWREG12	0x0030	W	0x00000000	Base address for RLC data (RLC) / stream start address/decoded
VDPU_SWREG13	0x0034	W	0x00000000	Base address for decoded picture / base address for JPEG deco
VDPU_SWREG14	0x0038	W	0x00000000	Base address for reference picture index 0 / base address for J
VDPU_SWREG14_VP8	0x0038	W	0x00000000	Base address for reference picture index 0 / base address for J
VDPU_SWREG15_JPEG_ROI	0x003c	W	0x00000000	JPEG roi control
VDPU_SWREG15	0x003c	W	0x00000000	Base address for reference picture index 1 / JPEG control

Name	Offset	Size	Reset Value	Description
VDPU_SWREG15_VP8	0x003c	W	0x00000000	Base address for reference picture index 1 / JPEG control
VDPU_SWREG16	0x0040	W	0x00000000	base address for reference picture index 2 / List of VLC code len
VDPU_SWREG17	0x0044	W	0x00000000	Base address for reference picture index 3 / List of VLC code le
VDPU_SWREG18	0x0048	W	0x00000000	Base address for reference picture index 4 / VC1 control / MPE
VDPU_SWREG18_VP8	0x0048	W	0x00000000	Base address for reference picture index 4 / VC1 control / MPE
VDPU_SWREG19	0x004c	W	0x00000000	Base address for reference picture index 5
VDPU_SWREG19_VP8	0x004c	W	0x00000000	Base address for reference picture index 5
VDPU_SWREG20	0x0050	W	0x00000000	Base address for reference picture index 6
VDPU_SWREG21	0x0054	W	0x00000000	Base address for reference picture index 7
VDPU_SWREG22	0x0058	W	0x00000000	Base address for reference picture index 8
VDPU_SWREG22_VP8	0x0058	W	0x00000000	Base address for reference picture index 8
VDPU_SWREG23	0x005c	W	0x00000000	Base address for reference picture index 9
VDPU_SWREG23_VP8	0x005c	W	0x00000000	Base address for reference picture index 9
VDPU_SWREG24	0x0060	W	0x00000000	Base address for reference picture index 10
VDPU_SWREG24_VP8	0x0060	W	0x00000000	Base address for reference picture index 10
VDPU_SWREG25_VP8	0x0064	W	0x00000000	Base address for reference picture index 10
VDPU_SWREG25	0x0064	W	0x00000000	Base address for reference picture index 11
VDPU_SWREG26_VP8	0x0068	W	0x00000000	Base address for reference picture index 10
VDPU_SWREG26	0x0068	W	0x00000000	Base address for reference picture index 12
VDPU_SWREG27_VP8	0x006c	W	0x00000000	Base address for reference picture index 13
VDPU_SWREG27	0x006c	W	0x00000000	Base address for reference picture index 13
VDPU_SWREG28_VP8	0x0070	W	0x00000000	Base address for reference picture index 10

Name	Offset	Size	Reset Value	Description
VDPU_SWREG28	0x0070	W	0x00000000	Base address for reference picture index14
VDPU_SWREG29_VP8	0x0074	W	0x00000000	Base address for reference picture index 10
VDPU_SWREG29	0x0074	W	0x00000000	Base address for reference picture index15
VDPU_SWREG30	0x0078	W	0x00000000	Reference picture numbers for index 0 and 1 (H264 VLC
VDPU_SWREG30_VP8	0x0078	W	0x00000000	Reference picture numbers for index 0 and 1 (H264 VLC
VDPU_SWREG31	0x007c	W	0x00000000	Reference picture numbers for index 2 and 3 (H264 VLC) /
VDPU_SWREG31_VP8	0x007c	W	0x00000000	Reference picture numbers for index 2 and 3 (H264 VLC) /
VDPU_SWREG32	0x0080	W	0x00000000	Reference picture numbers for index 4 and 5 (H264 VLC)
VDPU_SWREG32_VP8	0x0080	W	0x00000000	Reference picture numbers for index 4 and 5 (H264 VLC)
VDPU_SWREG33	0x0084	W	0x00000000	Reference picture numbers for index 6 and 7 (H264 VLC)
VDPU_SWREG33_VP8	0x0084	W	0x00000000	Reference picture numbers for index 6 and 7 (H264 VLC)
VDPU_SWREG34	0x0088	W	0x00000000	Reference picture numbers for index 8 and 9 (H264 VLC)
VDPU_SWREG34_VP8	0x0088	W	0x00000000	Reference picture numbers for index 8 and 9 (H264 VLC)
VDPU_SWREG35_JPEG_ROI	0x008c	W	0x00000000	JPEG roi offest/dc base address
VDPU_SWREG35	0x008c	W	0x00000000	Reference picture numbers for index 10 and 11 (H264 VLC)
VDPU_SWREG35_VP8	0x008c	W	0x00000000	Reference picture numbers for index 10 and 11 (H264 VLC)
VDPU_SWREG36	0x0090	W	0x00000000	Reference picture numbers for index 12 and 13 (H264 VLC)
VDPU_SWREG36_JPEG_ROI	0x0090	W	0x00000000	JPEG roi offset/dc length
VDPU_SWREG36_VP8	0x0090	W	0x00000000	Reference picture numbers for index 12 and 13 (H264 VLC)
VDPU_SWREG37_VP8	0x0094	W	0x00000000	Reference picture numbers for index 12 and 13 (H264 VLC)
VDPU_SWREG37	0x0094	W	0x00000000	Reference picture numbers for index 14 and 15 (H264 VLC)
VDPU_SWREG38	0x0098	W	0x00000000	Reference picture long term flags (H264 VLC) / VPx prediction filt

Name	Offset	Size	Reset Value	Description
VDPU_SWREG38_H264	0x0098	W	0x00000000	Reference picture numbers for index 12 and 13 (H264 VLC)
VDPU_SWREG39	0x009c	W	0x00000000	Reference picture valid flags (H264 VLC) /VPx prediction filter ta
VDPU_SWREG39_H264	0x009c	W	0x00000000	Reference picture numbers for index 12 and 13 (H264 VLC)
VDPU_SWREG40	0x00a0	W	0x00000000	Base address for standard dependent tables
VDPU_SWREG41	0x00a4	W	0x00000000	Base address for direct mode motion vectors
VDPU_SWREG42_VP8	0x00a8	W	0x00000000	
VDPU_SWREG42	0x00a8	W	0x00000000	bi_dir initial ref pic list register (0-2)
VDPU_SWREG43_VP8	0x00ac	W	0x00000000	
VDPU_SWREG43	0x00ac	W	0x00000000	bi-dir initial ref pic list register (3-5)
VDPU_SWREG44_VP8	0x00b0	W	0x00000000	
VDPU_SWREG44	0x00b0	W	0x00000000	bi-dir initial ref pic list register (6-8)
VDPU_SWREG45_VP8	0x00b4	W	0x00000000	
VDPU_SWREG45	0x00b4	W	0x00000000	bi-dir initial ref pic list register (9-11)
VDPU_SWREG46	0x00b8	W	0x00000000	bi-dir initial ref pic list register (12- 14) / VP7,VP8 quantization v
VDPU_SWREG46_VP8	0x00b8	W	0x00000000	bi-dir initial ref pic list register (12- 14) / VP7,VP8 quantization v
VDPU_SWREG47	0x00bc	W	0x00000000	bi-dir and P fwd initial ref pic list register (15 and P 0-3) / VP7,V
VDPU_SWREG47_VP8	0x00bc	W	0x00000000	bi-dir and P fwd initial ref pic list register (15 and P 0-3) / VP7,V
VDPU_SWREG48	0x00c0	W	0x00000000	Error concealment register
VDPU_SWREG49	0x00c4	W	0x00000000	Prediction filter tap register for H264, MPEG4, VC1
VDPU_SWREG50	0x00c8	W	0xfbb56f80	Synthesis configuration register decoder 0 (read only)
VDPU_SWREG51	0x00cc	W	0x00000000	Reference picture buffer control register
VDPU_SWREG52	0x00d0	W	0x00000000	Reference picture buffer information register 1 (read only)
VDPU_SWREG53	0x00d4	W	0x00000000	Reference picture buffer information register 2 (read only)

Name	Offset	Size	Reset Value	Description
VDPU_SWREG54	0x00d8	W	0xe5da0000	Synthesis configuration register decoder 1 (read only)
VDPU_SWREG55	0x00dc	W	0x00000000	Reference picture buffer 2 / Advanced prefetch control register
VDPU_SWREG56	0x00e0	W	0x00000000	Reference buffer information register 3 (read only)
VDPU_SWREG57_INTRA_INTER	0x00e4	W	0x00000000	intra_dll3t,intra_dblspeed,inter_dblspeed,stream_len_hi
VDPU_SWREG57	0x00e4	W	0x00000000	intra_dll3t,intra_dblspeed,inter_dblspeed,stream_len_hi
VDPU_SWREG58	0x00e8	W	0x00000000	Decoder debug register 0 (read only)
VDPU_SWREG59	0x00ec	W	0x00000000	H264 Chrominance 8 pixel interleaved data base
VDPU_SWREG60	0x00f0	W	0x00000000	Interrupt register post-processor
VDPU_SWREG61	0x00f4	W	0x01010100	Device configuration register post-processor
VDPU_SWREG62	0x00f8	W	0x00000000	Deinterlace control register
VDPU_SWREG63	0x00fc	W	0x00000000	base address for reading post-processing input picture uminan
VDPU_SWREG64	0x0100	W	0x00000000	Base address for reading post-processing input picture Cb/Ch
VDPU_SWREG65	0x0104	W	0x00000000	Base address for reading post-processing input picture Cr
VDPU_SWREG66	0x0108	W	0x00000000	Base address for writing post-processed picture luminance/RGB
VDPU_SWREG67	0x010c	W	0x00000000	Base address for writing post-processed picture Ch
VDPU_SWREG68	0x0110	W	0x00000000	Register for contrast adjusting
VDPU_SWREG69	0x0114	W	0x00000000	Register for colour conversion and contrast adjusting
VDPU_SWREG70	0x0118	W	0x00000000	Register for colour conversion 0
VDPU_SWREG71	0x011c	W	0x00000000	Register for colour conversion 1 + rotation mode
VDPU_SWREG72	0x0120	W	0x00000000	PP input size and -cropping register
VDPU_SWREG73	0x0124	W	0x00000000	PP input picture base address for Y bottom field
VDPU_SWREG74	0x0128	W	0x00000000	PP input picture base for Ch bottom field
VDPU_SWREG79	0x013c	W	0x00000000	Scaling ratio register 1 & padding for B
VDPU_SWREG80	0x0140	W	0x00000000	Scaling register 0 ratio & padding for R and G

Name	Offset	Size	Reset Value	Description
VDPU_SWREG81	0x0144	W	0x00000000	Scaling ratio register 2
VDPU_SWREG82	0x0148	W	0x00000000	Rmask register
VDPU_SWREG83	0x014c	W	0x00000000	Gmask register
VDPU_SWREG84	0x0150	W	0x00000000	Bmask register
VDPU_SWREG85	0x0154	W	0x00000000	Post-processor control register
VDPU_SWREG86	0x0158	W	0x00000000	Mask 1 start coordinate register
VDPU_SWREG87	0x015c	W	0x00000000	Mask 2 start coordinate register
VDPU_SWREG88	0x0160	W	0x00000000	Mask 1 size and PP original width register
VDPU_SWREG89	0x0164	W	0x00000000	Mask 2 size register
VDPU_SWREG90	0x0168	W	0x00000000	PiP register 0
VDPU_SWREG91	0x016c	W	0x00000000	PiP register 1 and dithering control
VDPU_SWREG92	0x0170	W	0x00000000	Display width and PP input size extension register
VDPU_SWREG93	0x0174	W	0x00000000	Display width and PP input size extension register
VDPU_SWREG94	0x0178	W	0x00000000	Base address for alpha blend 2 gui component
VDPU_SWREG95	0x017c	W	0x00000000	Base address for alpha blend 2 gui component
VDPU_SWREG98	0x0188	W	0x00000000	PP outupt width/height extension
VDPU_SWREG99	0x018c	W	0xe000f000	PP fuse register (read only)
VDPU_SWREG100	0x0190	W	0xff874780	Synthesis configuration register post-processor (read only)
VDPU_SWREG101	0x0194	W	0x00000000	soft reset signals

*Notes:Size:**B**- Byte (8 bits) access, **HW**- Half WORD (16 bits) access, **W**-WORD (32 bits) access*

2.4.10 VDPU Detail Register Description

VDPU_SWREG0

Address: Operational Base + offset (0x0000)
Register0000 Abstract

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	pro_num product number
15:12	RW	0x0	major_version major_version major_version
11:4	RW	0x00	minor_version minor_version minor_version
3	RW	0x0	ID_ASCII_EN ASCII type product ID enable ASCII type product ID enable

Bit	Attr	Reset Value	Description
2:0	RW	0x0	build_version build_version build_version

VDPU_SWREG1
Address: Operational Base + offset (0x0004)
interrupt register decoder

Bit	Attr	Reset Value	Description
31:25	RO	0x0	reserved
24	RW	0x0	sw_dec_pic_inf B slice detected. This signal is driven high during picture ready interrupt if B-type slice is found. This bit does not launch interrupt but is used to inform SW about h264 tools. [note]:the h264 decoder will use these bits.
23:19	RO	0x0	reserved
18	RW	0x0	sw_dec_timeout Interrupt status bit decoder timeout. When high the decoder 0 has been idling for too long. HW will self reset. Possible only if timeout interrupt is enabled [note]:the h264 and vp8 decoder will use these bits.
17	RW	0x0	sw_dec_slice_int Interrupt status bit dec_slice_decoded. When high SW must set new base addresses for sw_dec_out_base and sw_jpg_ch_out_base before reseting this status bit. Used for JPEG and VP8 snapshot modes [note]:the JPEG decoder will use these bits.
16	RW	0x0	sw_dec_error_int Interrupt status bit input stream error. When high, an error is found in input data stream decoding. HW will self reset. [note]:the h264 and vp8 decoder will use these bits.
15	RW	0x0	sw_dec_aso_int Interrupt status bit ASO (Arbitrary Slice Ordering) detected.When high, ASO detected in input data stream decoding. HW will self reset. [note]:the h264 decoder will use these bits.
14	RW	0x0	sw_dec_buffer_int Interrupt status bit input buffer empty. When high, input stream buffer is empty but picture is not ready. HW will not self reset. [note]:the h264 and vp8 decoder will use these bits.
13	RW	0x0	sw_dec_bus_int Interrupt status bit bus. Error response from bus. HW will self reset [note]:the h264 and vp8 decoder will use these bits.

Bit	Attr	Reset Value	Description
12	RW	0x0	sw_dec_rdy_int Interrupt status bit decoder. When this bit is high decoder has decoded a picture. HW will self reset. [note]:the h264 and vp8 decoder will use these bits.
11:9	RO	0x0	reserved
8	RW	0x0	sw_dec_irq Decoder IRQ. When high, decoder requests an interrupt. SW will reset this after interrupt is handled. [note]:the h264 and vp8 decoder will use these bits.
7:5	RO	0x0	reserved
4	RW	0x0	sw_dec_irq_dis Decoder IRQ disable. When high, there are no interrupts concerning decoder from HW. Polling must be used to see the interrupt statuses. [note]:the h264 and vp8 decoder will use these bits.
3:1	RO	0x0	reserved
0	RW	0x0	sw_dec_en decoder enable. Decoder enable. Setting this bit high will start the decoding operation. HW will reset this when picture is processed or ASO or stream error is detected or bus error or timeout interrupt is given. [note]:the h264 and vp8 decoder will use these bits.

VDPU_SWREG2
Address: Operational Base + offset (0x0008)
device configuration register decoder

Bit	Attr	Reset Value	Description
31:24	RW	0x01	sw_dec_axi_rd_id Read ID used for decoder reading services in AXI bus (if connected to AXI) [note]:the h264 and vp8 decoder will use these bits.
23	RW	0x0	sw_dec_timeout_e Timeout interrupt enable. If enabled HW may return timeout interrupt in case HW gets stucked while decoding picture. [note]:the h264 and vp8 decoder will use these bits.
22	RW	0x0	sw_dec_strswap32_e Decoder input 32bit data swap for stream data (may be used for 64 bit environment): 1'b0 = no swapping of 32 bit words 1'b1 = 32 bit data words are swapped (needed in 64 bit environment to achieve 7-6-5-4-3-2-1-0 byte order(also little endian should be enabled)) [note]:the h264 and vp8 decoder will use these bits.

Bit	Attr	Reset Value	Description
21	RW	0x0	sw_dec_strendian_e Decoder input endian mode for stream data: 1'b0 = Big endian (0-1-2-3 order) 1'b1 = Little endian (3-2-1-0 order) [note]:the h264 and vp8 decoder will use these bits.
20	RW	0x0	sw_dec_inswap32_e Decoder input 32bit data swap for other than stream data (may be used for 64 bit environment): 1'b0 = no swapping of 32 bit words 1'b1 = 32 bit data words are swapped (needed in 64 bit environment to achieve 7-6-5-4-3-2-1-0 byte order(also little endian should be enabled)) [note]:the h264 and vp8 decoder will use these bits.
19	RW	0x0	sw_dec_outswap32_e Decoder output 32bit data swap (may be used for 64 bit environment): 1'b0 = no swapping of 32 bit words 1'b1 = 32 bit data words are swapped (needed in 64 bit environment to achieve 7-6-5-4-3-2-1-0 byte order(also little endian should be enabled)) [note]:the h264 and vp8 decoder will use these bits.
18	RW	0x0	sw_dec_data_disc_e Data discard enable. Precise burst lengths are used with reading services. Extra data is discarded internally. [note]:the h264 and vp8 decoder will use these bits.
17	RW	0x0	sw_tiled_mode_msb Tiled mode msb. Concanated to Tiled mode lsb which form 2 bit tiled mode. Definition of tiledmode: 1'b0 = Tiled mode not enabled 1'b1 = Tiled mode enabled for 8x4 tile size [note]:the h264 and vp8 decoder will use these bits.
16:11	RW	0x00	sw_dec_latency Decoder master interface additional latency. Can be used to slow down decoder HW between services in steps of 8 clock cycles: 6'd0 = no latency 6'd1 = minimum 8 cycles of IDLE between services 6'd2 = minimum 16 cycles of IDLE between services ... 6'd63 = minimum latency of 504 cycles of IDLE between services [note]:the h264 and vp8 decoder will use these bits.

Bit	Attr	Reset Value	Description
10	RW	0x1	sw_dec_clk_gate_e Decoder dynamic clock gating enable: 0 = Clock is running for all structures 1 = Clock is gated for decoder structures that are not used Note: Clock gating value can be changed only when decoder is disabled
9	RW	0x0	sw_dec_in_endian Decoder input endian mode for other than stream data: 0 = Big endian (0-1-2-3 order) 1 = Little endian (3-2-1-0 order) [note]:the h264 and vp8 decoder will use these bits.
8	RW	0x0	sw_dec_out_endian Decoder output endian mode: 0 = Big endian (0-1-2-3 order) 1 = Little endian (3-2-1-0 order) [note]:the h264 and vp8 decoder will use these bits.
7	RW	0x0	sw_tiled_mode_lsb Tiled mode lsb. Concanated to Tiled mode msb which form 2 bit tiled mode. Defined in tiled_mode_msb [note]:the h264 and vp8 decoder will use these bits.
6	RW	0x0	sw_dec_adv_pre_dis Advanced PREFETCH mode disable (advanced reference picture reading mode for video) [note]:the h264 and vp8 decoder will use these bits.
5	RW	0x0	sw_dec_scmd_dis AXI Single Command Multiple Data 0 disable. (where only the first addresses of the burst are given from address generator). This bit is used to disable the feature (possible SW workaround if something is not working correctly) [note]:the h264 and vp8 decoder will use these bits.
4:0	RW	0x00	sw_dec_max_burst Maximum burst length for decoder bus transactions. Valid values: AXI: 1-16 [note]:the h264 and vp8 decoder will use these bits.

VDPU_SWREG3
Address: Operational Base + offset (0x000c)
Device control register 0(decmode, picture type etc)

Bit	Attr	Reset Value	Description
31:28	RW	0x0	sw_dec_mode Decoding mode: 4'd0 = H.264, 4'd1 = MPEG-4, 4'd2 = H.263, 4'd3 = JPEG, 4'd4 = VC-1, 4'd5 = MPEG-2, 4'd6 = MPEG-1, 4'd8 = RV, 4'd9 = VP7, 4'd10 = VP8, 4'd11 = AVS, others = reserved [note]:all the decoder mode will use these bits.
27	RW	0x0	sw_rlc_mode_e RLC mode enable: 1 = HW decodes video from RLC input data + side information (Differential MV's, separate DC coeffs, Intra 4x4 modes, MB control). Valid only for H.264 Baseline and MPEG- 4 SP. 0 = HW decodes video from bit stream (VLC mode) + side information (bitplane data in VC-1) [note]:the h264 and MPEG4 decoder will use these bits.
26	RW	0x0	sw_skip_mode AVS: 0: special MB type code indicates skipped mbs 1 means that skipped mbs are indicaed using skip_run -syntax elememnt like in VP8: 0 : HW decodes mb_coeff_skip -flag 1 : HW does not decode mb_coeff_skip -flag [note]:the vp8 decoder will use these bits.
24	RW	0x0	sw_pjpeg_e Progressive JPEG enable: 0 = baseline JPEG 1 = progressive JPEG
23	RW	0x0	sw_pic_interlace_e Coding mode of the current picture: 0 = progressive 1 = interlaced [note]:the h264 decoder will use these bits.

Bit	Attr	Reset Value	Description
22	RW	0x0	sw_pic_fieldmode_e Structure of the current picture (residual structure) 0 = frame structure, this means MBAFF structured picture for interlaced sequence 1 = field structure [note]:the h264 decoder will use these bits.
21	RW	0x0	sw_pic_b_e B picture enable for current picture: 0=picture type is I or P depending on sw_pic_inter_e 1=picture type is BI (vc1)/D (mpeg1) or B depending on sw_pic_inter_e (not valid for H264 since its slice based information)
20	RW	0x0	sw_pic_inter_e Picture type. 1= Inter type (P) 0= Intra type (I) See also sw_pic_b_e [note]:the vp8 decoder will use these bits.
19	RW	0x0	sw_pic_topfield_e If field structure is enabled this bit informs which one of the fields is being decoded: 0 = bottom field 1 = top field [note]:the h264 decoder will use these bits.
18	RW	0x0	sw_fwd_interlace_e Coding mode of forward reference picture: 0 = progressive 1 = interlaced Note: for backward reference picture the coding mode is always same as for current picture.
16	RW	0x0	sw_ref_topfield_e Indicates which field should be used as reference if sw_ref_frames = 0 : 0 = bottom field 1 = top field used only in VC-1 mode
15	RW	0x0	sw_dec_out_dis Disable decoder output picture writing: 0 = Decoder output picture is written to external memory 1 = Decoder output picture is not written to external memory [note]:the h264 and vp8 decoder will use these bits.

Bit	Attr	Reset Value	Description
14	RW	0x0	sw_filtering_dis De-block filtering disable 1 = filtering is disabled for current picture 0 = filtering is enabled for current picture [note]:the h264 decoder will use these bits.
13	RW	0x0	sw_pic_fixed_quant sw_pic_fixed_quant (DEC mode is VC-1 and AVS) 0 = Quantization parameter can vary inside picture 1 = Quantization parameter is fixed (pquant) sw_mvc_e(DEC mode is H264) multi view coding enable. Possible for H264 only [note]:the h264 decoder will use these bits.
12	RW	0x0	sw_write_mvs_e Direct mode motion vector write enable for current picture / MPEG2 motion vector write enable for error concealment purposes: 0 = writing disabled for current picture 1 = the direct mode motion vectors are written to extrenal memory. H264 direct mode motion vectors are written to DPB aside with the corresponding reference picture. Other decoding mode dir mode mvs are written to external memory starting from sw_dir_mv_base [note]:the h264 decoder will use these bits.
11	RW	0x0	sw_reftopfirst_e Indicates which FWD reference field has been decoded first. 0 = FWD reference bottom field 1 = FWD reference top field [note]:the h264 decoder will use these bits.
10	RW	0x0	sw_seq_mbaff_e Sequence includes MBAFF coded pictures [note]:the h264 decoder will use these bits.
9	RW	0x0	sw_picord_count_e h264_high config: Picture order count table read enable. If enabled HW will read picture order counts from memory in the beginning of picture [note]:the h264 decoder will use these bits.
8	RW	0x0	sw_dec_timeout_mode dec timeout mode selset when 1'b0 , timeout cycle is 181'b1 when 1'b1, timeout cycle is 221'b1 [note]:the h264 and vp8 decoder will use these bits.

Bit	Attr	Reset Value	Description
7:0	RW	0x01	sw_dec_axi_wr_id Write ID used for decoder writing services in AXI bus (if connected to AXI) [note]:the h264 and vp8 decoder will use these bits.

VDPU_SWREG4_H264
Address: Operational Base + offset (0x0010)
decoder control register 1(picture parameters)

Bit	Attr	Reset Value	Description
31:23	RW	0x000	sw_pic_mb_width Picture width in macroblocks = ((width in pixels + 15) /16) [note]:the h264 decoder will use these bits.
22:19	RO	0x0	reserved
18:11	RW	0x00	sw_pic_mb_height_p Picture height in macroblocks =((height in pixels+15)/16). Picture height is informed as size of the (progressive) frame also for single field (of interlaced content) is being decoded [note]:the h264 decoder will use these bits.
10:5	RO	0x0	reserved
4:0	RW	0x00	sw_ref_frames 264: num_ref_frames, maximum number of short and long term reference frames in decoded picture buffer.

VDPU_SWREG4
Address: Operational Base + offset (0x0010)
decoder control register 1(picture parameters)

Bit	Attr	Reset Value	Description
31:23	RW	0x000	sw_pic_mb_width Picture width in macroblocks = ((width in pixels + 15) /16) [note]:the vp8 decoder will use these bits.
22:19	RW	0x0	sw_mb_width_off The amount of meaningfull horizontal pixels in last MB (width offset) 0 if exactly 16 pixels multiple picture and all the horizontal pixels in last MB are meaningfull
18:11	RW	0x00	sw_pic_mb_height_p Picture height in macroblocks =((height in pixels+15)/16). Picture height is informed as size of the (progressive) frame also for single field (of interlaced content) is being decoded [note]:the vp8 decoder will use these bits.
10:7	RW	0x0	sw_mb_height_off The amount of menaingfull vertical pixels in last MB (height offset 0 if exactly 16 pixels multiple picture and all the vertical pixels in last MB are meaningfull

Bit	Attr	Reset Value	Description
6	RW	0x0	sw_alt_scan_e indicates alternative vertical scan method used for interlacedd frames
5:3	RW	0x0	sw_pic_mb_w_ext Picture mb width extension. If sw_pic_mb_width does not fit tod 9 bits then these bits are used to increase the range upto 11 bits (used as 3 msb)
2:0	RW	0x0	sw_pic_mb_h_ext Picture mb height extension. If sw_pic_mb_height_p does not fit to 9 bits then these bits are used to increase the range upto 11 bits (used as 3 msb)

VDPU_SWREG5
Address: Operational Base + offset (0x0014)
decoder control register2 (stream decoding table selects)

Bit	Attr	Reset Value	Description
31:26	RW	0x00	sw_strm_start_bit Exact bit of stream start word where decoding can be started (assosiates with sw_rlc_vlc_base)
25	RW	0x0	sw_sync_marker_e Sync markers enable: '0' = synch markers are not used, '1' = synch markers are used. For progressive JPEG this indicates that there are restart markers in the stream after restart interval steps
24	RW	0x0	sw_type1_quant_e MPEG4: Type 1 quantization enable '0' = type 2 inverse Q method '1' = type 1 inverse Q method (Q-tables used) H264 (h264_high config:): scaling matrix enable: '0' = normal transform '1' = use scaling matrix for transform (read from external
23:19	RW	0x00	sw_ch_qp_offset Chroma Qp filter offset. (For H.264 this offset concerns Cb only)
18:14	RW	0x00	sw_ch_qp_offset2 Chroma Qp filter offset for cr type
13:1	RO	0x0	reserved
0	RW	0x0	sw_fieldpic_flag_e Flag for streamd that field_pic_flag exists in stream

VDPU_SWREG5_H264
Address: Operational Base + offset (0x0014)
decoder control register2 (stream decoding table selects)

Bit	Attr	Reset Value	Description
31:26	RW	0x00	sw_strm_start_bit Exact bit of stream start word where decoding can be started (assosiates with sw_rlc_vlc_base)
25	RO	0x0	reserved
24	RW	0x0	sw_type1_quant_e MPEG4: Type 1 quantization enable '0' = type 2 inverse Q method '1' = type 1 inverse Q method (Q-tables used) H264 (h264_high config:): scaling matrix enable: '0' = normal transform '1' = use scaling matrix for transform (read from external
23:19	RW	0x00	sw_ch_qp_offset Chroma Qp filter offset. (For H.264 this offset concerns Cb only)
18:14	RW	0x00	sw_ch_qp_offset2 Chroma Qp filter offset for cr type
13:1	RO	0x0	reserved
0	RW	0x0	sw_fieldpic_flag_e Flag for streamd that field_pic_flag exists in stream

VDPU_SWREG5_VP8
Address: Operational Base + offset (0x0014)
decoder control register2 (stream decoding table selects)

Bit	Attr	Reset Value	Description
31:26	RW	0x00	sw_strm_start_bit Exact bit of stream start word where decoding can be started (assosiates with sw_rlc_vlc_base)
25:24	RO	0x0	reserved
23:18	RW	0x00	sw_strm1_start_bit it for ctrl-stream (needed if multistream is enabled, assosiates with sw_bitpl_ctrl_base)
17:16	RO	0x0	reserved
15:8	RW	0x00	sw_boolean_value initial value for boolean dec 0
7:0	RW	0x00	sw_boolean_range initial range for boolean dec 0

VDPU_SWREG6
Address: Operational Base + offset (0x0018)
decoder control register 3(stream buffer information)

Bit	Attr	Reset Value	Description
31	RW	0x0	sw_start_code_e Bit for indicating stream start code existence: '0' = stream doesn't contain start codes '1' = stream contains start codes [note]:the h264 decoder will use these bits.

Bit	Attr	Reset Value	Description
30:25	RW	0x00	sw_init_qp Initial value for quantization parameter (picture quantizer). [note]:the h264 decoder will use these bits.
24	RW	0x0	sw_ch_8pix_ileav_e Enable for additional chrominance data format writing where decoder writes chrominance in group of 8 pixels of Cb and then corresponding 8 pixels of Cr. Data is written to sw_dec_ch8pix_base. Cannot be used if tiled mode is enabled [note]:the h264 decoder will use these bits.
23:0	RW	0x000000	sw_stream_len Amount of stream data bytes in input buffer. If the given buffer size is not enough for finishing the picture the corresponding interrupt is given and new stream buffer base address and stream buffer size information should be given (assosiates with sw_rlc_vlc_base). For VC-1 the buffer must include data for one picture/slice of the picture For H264/MPEG4/H263/MPEG2/MPEG1 the buffer must include at least data for one slice/VP of the picture For JPEG the buffer size must be a multiple of 256 bytes or the amount of data for onepicture. [note]:the h264 and vp8 decoder will use these bits.

VDPU_SWREG7
Address: Operational Base + offset (0x001c)
decoder control register 4(H264, VC-1 control)

Bit	Attr	Reset Value	Description
31	RW	0x0	sw_cabac_e CABAC enable [note]:the h264 decoder will use these bits.
30	RW	0x0	sw_blackwhite_e '0' = 4:2:0 sampling format '1' = 4:0:0 sampling format (H264 monochroma) [note]:the h264 decoder will use these bits.
29	RW	0x0	sw_dir_8x8_infer_e Specifies the method to use to derive luma motion vectors in B_skip, B_Direct_16x16 and B_direct_8x8_inference_flag (see direct_8x8_inference flag) [note]:the h264 decoder will use these bits.
28	RW	0x0	sw_weight_pred_e Weighted prediction enable for P slices [note]:the h264 decoder will use these bits.

Bit	Attr	Reset Value	Description
27:26	RW	0x0	sw_weight_bipr_idc weighted prediction specification for B slices: "2'b00" = default weighted prediction is aplied to B slices "2'b01" = explicit weighted prediction shall be applied to B slices "2'b10" = implicit weighted prediction shall be applied to B slices [note]:the h264 decoder will use these bits.
25:21	RO	0x0	reserved
20:16	RW	0x00	sw_framenum_len H.264: Bit length of frame_num in data stream RV: frame size length. Informs how many bits in stream are used for frame size (HW discards these bits) [note]:the h264 decoder will use these bits.
15:0	RW	0x0000	sw_framenum current frame_num, used to identify short-term reference frames. Used in reference picture reordering [note]:the h264 decoder will use these bits.

VDPU_SWREG7_VP8

Address: Operational Base + offset (0x001c)

decoder control register 4(H264, VC-1 control)

Bit	Attr	Reset Value	Description
31:26	RW	0x00	sw_dct1_start_bit start bits for VP7/VP8 DCT stream partition index 1
25:20	RW	0x00	sw_dct2_start_bit start bits for VP7/VP8 DCT stream partition index 2
19:14	RO	0x0	reserved
13	RW	0x0	sw_ch_mv_res VP7/VP8 Chrominance motion vector resolution: '0' = Full pixel '1' = 1/8 pixel
12	RW	0x0	sw_bilin_mc_e bilinear motion compensation enable: '0' = Bicubic interpolation used '1' = Bilinear interpolation used
11:9	RW	0x0	sw_init_dc_match0 initial DC prediction mach count 0. After HW has decoded a picture HW returns the final match count0 information which is read by SW
8:6	RW	0x0	sw_init_dc_match1 initial DC prediction mach count 1. After HW has decoded a picture HW returns the final match count1 information which is read by SW

Bit	Attr	Reset Value	Description
5	RW	0x0	sw_vp7_version VP7 version information to streamd: '0'= vp7 version 7.0 '1'= vp7 version 7.1 or better
4:0	RO	0x0	reserved

VDPU_SWREG8
Address: Operational Base + offset (0x0020)
decoder control register 5(H264, VC-1 and RV control)

Bit	Attr	Reset Value	Description
31	RW	0x0	sw_const_intra_e constrained_intra_pred_flag equal to 1 specifies that intra prediction uses only neighbouring intra macroblocks in prediction. When equal to 0 also neighbouring inter macroblocks are used in intra prediction process. [note]:the h264 decoder will use these bits.
30	RW	0x0	sw_filt_ctrl_pres deblocking_filter_control_present_flag indicates whether extra variables controlling characteristics of the deblocking filter are present in the slice header. [note]:the h264 decoder will use these bits.
29	RW	0x0	sw_rdpic_cnt_pres redundant_pic_cnt_present_flag specifies whether redundant_pic_cnt syntax elements [note]:the h264 decoder will use these bits.
28	RW	0x0	sw_8x8trans_flag_e 8x8 transform flag enable for stream decoding [note]:the h264 decoder will use these bits.
27:17	RW	0x000	sw_refpic_mk_len Length of decoded reference picture marking bits [note]:the h264 decoder will use these bits.
16	RW	0x0	sw_idr_pic_e IDR (instantaneous decoding refresh) picture flag. [note]:the h264 decoder will use these bits.
15:0	RW	0x0000	sw_idr_pic_id idr_pic_id, identifies IDR (instantaneous decoding refresh) picture [note]:the h264 decoder will use these bits.

VDPU_SWREG8_VP8
Address: Operational Base + offset (0x0020)
decoder control register 5(H264, VC-1 and RV control)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	sw_init_dc_comp0 initial DC predictor value 0. After HW has decoded a picture HW returns the final predictor value information which is read by SW
15:0	RW	0x0000	sw_init_dc_comp1 initial DC predictor value 1. After HW has decoded a picture HW returns the final predictor value information which is read by SW

VDPU_SWREG9
Address: Operational Base + offset (0x0024)
decoder control register 6

Bit	Attr	Reset Value	Description
31:24	RW	0x00	sw_pps_id pic_parameter_set_id, identifies the picture parameter set that is referred to in the slice header. [note]:the h264 decoder will use these bits.
23:19	RW	0x00	sw_refidx1_active pecifies the maximum reference index that can be used while decoding inter predicted macro blocks. [note]:the h264 decoder will use these bits.
18:14	RW	0x00	sw_refidx0_active Specifies the maximum reference index that can be used while decoding inter predicted macro blocks. This is same as in previous decoders (width increased with q bit) [note]:the h264 decoder will use these bits.
13:8	RO	0x0	reserved
7:0	RW	0x00	sw_poc_length Length of picture order count field in stream [note]:the h264 decoder will use these bits.

VDPU_SWREG9_VP8
Address: Operational Base + offset (0x0024)
decoder control register 6

Bit	Attr	Reset Value	Description
31:28	RO	0x0	reserved
27:24	RW	0x0	sw_coeffs_part_am VP7/VP8 number of coefficient partitions (should it be number of additional DCT partitions with range 0-7??)
23:0	RW	0x000000	sw_stream1_len amount of CTRL stream data bytes in input buffer. (needed if multistream is enabled or VP7/VP8 format, assosiates with sw_bitpl_ctrl_base)

VDPU_SREG10_H264_RLC
Address: Operational Base + offset (0x0028)
Base address for differential motion vector base address

Bit	Attr	Reset Value	Description
31:2	RW	0x00000000	sw_diff_mv_base for H264 and MPEG4, RLC mode: Differential motion vector base address.
1:0	RO	0x0	reserved

VDPU_SREG10_H264
Address: Operational Base + offset (0x0028)
Base address for differential motion vector base address

Bit	Attr	Reset Value	Description
31:30	RO	0x0	reserved
29:25	RW	0x00	sw_pinit_rlist_f9 initial reference picture list for P forward picid 9
24:20	RW	0x00	sw_pinit_rlist_f8 initial reference picture list for P forward picid 8
19:15	RW	0x00	sw_pinit_rlist_f7 initial reference picture list for P forward picid 7
14:10	RW	0x00	sw_pinit_rlist_f6 initial reference picture list for P forward picid 6
9:5	RW	0x00	sw_pinit_rlist_f5 initial reference picture list for P forward picid 5
4:0	RW	0x00	sw_pinit_rlist_f4 initial reference picture list for P forward picid 4

VDPU_SREG10_VP8
Address: Operational Base + offset (0x0028)
Base address for differential motion vector base address

Bit	Attr	Reset Value	Description
31:2	RW	0x00000000	sw_segment_base VP7/VP8: base address for segmentation map values 0
1	RW	0x0	sw_segment_upd_e VP7/VP8 Segmentation map update enable: '0': segmentation values are read from external memory (from segment_base) '1': segmentation update is included in stream
0	RW	0x0	sw_segment_e segmentation enable: '0': segmentation is not enabled '1': segmentation is enabled (sw_segment_upd_e value is used)

VDPU_SWREG11_H264_RLC

Address: Operational Base + offset (0x002c)
decoder control register 7

Bit	Attr	Reset Value	Description
31:2	RW	0x00000000	sw_i4x4_or_dc_base RLC mode: H.264: Intra prediction 4x4 mode base address. RLC mode: MPEG-4: DC component base address.
1:0	RO	0x0	reserved

VDPU_SWREG11_H264
Address: Operational Base + offset (0x002c)
decoder control register 7

Bit	Attr	Reset Value	Description
31:30	RO	0x0	reserved
29:25	RW	0x00	sw_pinit_rlist_f15 Initial reference picture list for P forward picid 15
24:20	RW	0x00	sw_pinit_rlist_f14 Initial reference picture list for P forward picid 14
19:15	RW	0x00	sw_pinit_rlist_f13 Initial reference picture list for P forward picid 13
14:10	RW	0x00	sw_pint_rlist_f12 Initial reference picture list for P forward picid 12
9:5	RW	0x00	sw_pint_rlist_f11 Initial reference picture list for P forward picid 11
4:0	RW	0x00	sw_pint_rlist_f10 Initial reference picture list for P forward picid 10

VDPU_SWREG11_VP8
Address: Operational Base + offset (0x002c)

Bit	Attr	Reset Value	Description
31:30	RO	0x0	reserved
29:24	RW	0x00	sw_dct3_start_bit Start for VP7/VP8 DCT stream partition index 3
23:18	RW	0x00	sw_dct4_start_bit Start for VP7/VP8 DCT stream partition index 4
17:12	RW	0x00	sw_dct5_start_bit Start for VP7/VP8 DCT stream partition index 5
11:6	RW	0x00	sw_dct6_start_bit Start for VP7/VP8 DCT stream partition index 6
5:0	RW	0x00	sw_dct7_start_bit Start for VP7/VP8 DCT stream partition index 7

VDPU_SWREG12
Address: Operational Base + offset (0x0030)
Base address for RLC data (RLC) / stream start address/decoded

Bit	Attr	Reset Value	Description
31:2	RW	0x00000000	sw_rlc_vlc_base RLC mode: Base address for RLC data (swreg3.sw_rlc_mode_e = 1). VLC mode: Stream start address / end addr+I288ess with byte precision (swreg4.rlc_mode_en = 0), start bit number in swreg5.stream_start_bit. When sw_dec_buffer_int is high or sw_dec_e is low this register contains HW return value of last_byte_address (not valid for jpeg) where stream has been read (and used) in accuracy of byte. For debug purposses the last_byte_address is also written when stream error/ASO is detected even though it may not be accurate. VP7/VP8: This base address is used as sw_dct_strm0_base including DCT stream for MB rows 0,2n [note]:the h264 and vp8 decoder will use these bits.
1:0	RO	0x0	reserved

VDPU_SWREG13
Address: Operational Base + offset (0x0034)
Base address for decoded picture / base address for JPEG deco

Bit	Attr	Reset Value	Description
31:2	RW	0x00000000	sw_dec_out_base Video: Base address for decoder output picture. Points directly to start of decoder output picture or field. JPEG/VP8 snapshot: Base address for decoder output luminance picture [note]:the h264 and vp8 decoder will use these bits.
1:0	RO	0x0	reserved

VDPU_SWREG14
Address: Operational Base + offset (0x0038)
Base address for reference picture index 0 / base address for J

Bit	Attr	Reset Value	Description
31:2	RW	0x00000000	sw_refer0_base Base address for reference picture index 0. See picture index definition from toplevel_sp [note]:the h264 decoder will use these bits.
1	RW	0x0	sw_refer0_field_e Refer picture consist of single fields or frame: '0' = reference picture consists of frame '1' = reference picture consists of fields [note]:the h264 decoder will use these bits.

Bit	Attr	Reset Value	Description
0	RW	0x0	sw_refer0_topc_e Which field of reference picture is closer to current picture: '0' = bottom field is closer to current picture '1' = top field is closer to current picture [note]:the h264 decoder will use these bits.

VDPU_SWREG14_VP8

Address: Operational Base + offset (0x0038)

Base address for reference picture index 0 / base address for J

Bit	Attr	Reset Value	Description
31:2	RW	0x00000000	sw_ch_out_base base address for decoder output chrominance picture. Used in JPEG and VP8 intra picture mode (not needed if decoder output is not written)
1:0	RO	0x0	reserved

VDPU_SWREG15_JPEG_ROI

Address: Operational Base + offset (0x003c)

JPEG roi control

Bit	Attr	Reset Value	Description
31:20	RO	0x0	reserved
19	RW	0x0	sw_jpegroi_in_endian jpeg offset input endian sw_jpegroi_in_endian 0 = big endian (0-1-2-3 order) 1 = little endian (3-2-1-0 order)
18	RW	0x0	sw_jpegroi_in_swap32 jpeg offset input 32-bit swap sw_jpegroi_in_swap32 0: no swapping of 32 bit words 1: 32bit data words are swapped (needed in 64 bit environment to achieve 7-6-5-4-3-2-1 byte order (also little endian should be enabled)
17:16	RW	0x0	sw_roi_sample_size ROI MB num sample each time ROI MB num sample each time 2'b00:1 2'b01:8 2'b10:16 2'b11:8
15:12	RW	0x0	sw_roi_distance roi distance The distance between the sample MB and ROI start MB

Bit	Attr	Reset Value	Description
11:10	RW	0x0	sw_roi_out_sel roi output selection ROI output selection 2'b00: output offset/dc 2'b01: output picture 2'b10: output offset/dc and picture 2'b11: output offset/dc
9	RW	0x0	sw_roi_decode roi decode JPEG ROI decode 0: build offset/dc table 1: ROI decode
8	RW	0x0	sw_roi_en roi enable JPEG roi mode enable 0: normal jpeg decode mode 1: JPEG roi mode
7:0	RO	0x0	reserved

VDPU_SWREG15
Address: Operational Base + offset (0x003c)
Base address for reference picture index 1 / JPEG control

Bit	Attr	Reset Value	Description
31:2	RW	0x00000000	sw_refer1_base Base address for reference picture index 1. See picture index definition from toplevel_sp [note]:the h264 and vp8 decoder will use these bits.
1	RW	0x0	sw_refer1_field_e Refer picture consist of single fields or frame: '0' = reference picture consists of frame '1' = reference picture consists of fields
0	RW	0x0	sw_refer1_topc_e Which field of reference picture is closer to current picture: '0' = bottom field is closer to current picture '1' = top field is closer to current picture

VDPU_SWREG15_VP8
Address: Operational Base + offset (0x003c)
Base address for reference picture index 1 / JPEG control

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:0	RW	0x00	sw_slice_h Slice mode must be used if picture size is more than 16 Mpixels. However for bigger than 4096 MBs the slice mode usage is recommended

VDPU_SWREG16
Address: Operational Base + offset (0x0040)
base address for reference picture index 2 / List of VLC code len

Bit	Attr	Reset Value	Description
31:2	RW	0x00000000	sw_refer2_base Base address for reference picture index 2. See picture index definition from toplevel_sp [note]:the h264 decoder will use these bits.
1	RW	0x0	sw_refer2_field_e Refer picture consist of single fields or frame: '0' = reference picture consists of frame '1' = reference picture consists of fields [note]:the h264 decoder will use these bits.
0	RW	0x0	sw_refer2_topc_e Which field of reference picture is closer to current picture: '0' = bottom field is closer to current picture '1' = top field is closer to current picture [note]:the h264 decoder will use these bits.

VDPU_SWREG17
Address: Operational Base + offset (0x0044)
Base address for reference picture index 3 / List of VLC code le

Bit	Attr	Reset Value	Description
31:2	RW	0x00000000	sw_refer3_base Base address for reference picture index 3. See picture index definition from toplevel_sp [note]:the h264 decoder will use these bits.
1	RW	0x0	sw_refer3_field_e Refer picture consist of single fields or frame: '0' = reference picture consists of frame '1' = reference picture consists of fields [note]:the h264 decoder will use these bits.
0	RW	0x0	sw_refer3_topc_e Which field of reference picture is closer to current picture: '0' = bottom field is closer to current picture '1' = top field is closer to current picture [note]:the h264 decoder will use these bits.

VDPU_SWREG18
Address: Operational Base + offset (0x0048)
Base address for reference picture index 4 / VC1 control / MPE

Bit	Attr	Reset Value	Description
31:2	RW	0x00000000	sw_refer4_base Base address for reference picture index 4. See picture index definition from toplevel_sp [note]:the h264 and vp8 decoder will use these bits.
1	RW	0x0	sw_refer4_field_e Refer picture consist of single fields or frame: '0' = reference picture consists of frame '1' = reference picture consists of fields [note]:the h264 decoder will use these bits.
0	RW	0x0	sw_refer4_topc_e Which field of reference picture is closer to current picture: '0' = bottom field is closer to current picture '1' = top field is closer to current picture [note]:the h264 decoder will use these bits.

VDPU_SWREG18_VP8
Address: Operational Base + offset (0x0048)
Base address for reference picture index 4 / VC1 control / MPE

Bit	Attr	Reset Value	Description
31:2	RW	0x00000000	sw_refer4_base Base address for reference picture index 4. See picture index definition from toplevel_sp
1	RO	0x0	reserved
0	RW	0x0	sw_gref_sign_bias reference picture sign bias for Golden reference frame 0

VDPU_SWREG19
Address: Operational Base + offset (0x004c)
Base address for reference picture index 5

Bit	Attr	Reset Value	Description
31:2	RW	0x00000000	sw_refer5_base Base address for reference picture index 5. See picture index definition from toplevel_sp [note]:the h264 decoder will use these bits.
1	RW	0x0	sw_refer5_field_e Refer picture consist of single fields or frame: '0' = reference picture consists of frame '1' = reference picture consists of fields [note]:the h264 decoder will use these bits.
0	RW	0x0	sw_refer5_topc_e Which field of reference picture is closer to current picture: '0' = bottom field is closer to current picture '1' = top field is closer to current picture [note]:the h264 decoder will use these bits.

VDPU_SWREG19_VP8
Address: Operational Base + offset (0x004c)
Base address for reference picture index 5

Bit	Attr	Reset Value	Description
31:2	RW	0x00000000	sw_refer5_base Base address for reference picture index 5. See picture index definition from toplevel_sp [note]:the h264 decoder will use these bits.
1	RO	0x0	reserved
0	RW	0x0	sw_aref_sign_bias Reference picture sign bias for Alternate reference frame

VDPU_SWREG20
Address: Operational Base + offset (0x0050)
Base address for reference picture index 6

Bit	Attr	Reset Value	Description
31:2	RW	0x00000000	sw_refer6_base Base address for reference picture index 6. See picture index definition from toplevel_sp [note]:the h264 decoder will use these bits.
1	RW	0x0	sw_refer6_field_e Refer picture consist of single fields or frame: '0' = reference picture consists of frame '1' = reference picture consists of fields [note]:the h264 decoder will use these bits.
0	RW	0x0	sw_refer6_topc_e Which field of reference picture is closer to current picture: '0' = bottom field is closer to current picture '1' = top field is closer to current picture [note]:the h264 decoder will use these bits.

VDPU_SWREG21
Address: Operational Base + offset (0x0054)
Base address for reference picture index 7

Bit	Attr	Reset Value	Description
31:2	RW	0x00000000	sw_refer7_base Base address for reference picture index 7. See picture index definition from toplevel_sp [note]:the h264 decoder will use these bits.
1	RW	0x0	sw_refer7_field_e Refer picture consist of single fields or frame: '0' = reference picture consists of frame '1' = reference picture consists of fields [note]:the h264 decoder will use these bits.

Bit	Attr	Reset Value	Description
0	RW	0x0	sw_refer7_topc_e Which field of reference picture is closer to current picture: '0' = bottom field is closer to current picture '1' = top field is closer to current picture [note]:the h264 decoder will use these bits.

VDPU_SWREG22
Address: Operational Base + offset (0x0058)
Base address for reference picture index 8

Bit	Attr	Reset Value	Description
31:2	RW	0x00000000	sw_refer8_base Base address for reference picture index 8. See picture index definition from toplevel_sp
1	RW	0x0	sw_refer8_field_e Refer picture consist of single fields or frame: '0' = reference picture consists of frame '1' = reference picture consists of fields
0	RW	0x0	sw_refer8_topc_e Which field of reference picture is closer to current picture: '0' = bottom field is closer to current picture '1' = top field is closer to current picture

VDPU_SWREG22_VP8
Address: Operational Base + offset (0x0058)
Base address for reference picture index 8

Bit	Attr	Reset Value	Description
31:2	RW	0x00000000	sw_dct_strm1_base base address for VP7/VP8 DCT stream MB row 1,2n+1 0 30 R/
1:0	RO	0x0	reserved

VDPU_SWREG23
Address: Operational Base + offset (0x005c)
Base address for reference picture index 9

Bit	Attr	Reset Value	Description
31:2	RW	0x00000000	sw_refer9_base Base address for reference picture index 9. See picture index definition from toplevel_sp
1	RW	0x0	sw_refer9_field_e Refer picture consist of single fields or frame: '0' = reference picture consists of frame '1' = reference picture consists of fields
0	RW	0x0	sw_refer9_topc_e Which field of reference picture is closer to current picture: '0' = bottom field is closer to current picture '1' = top field is closer to current picture

VDPU_SWREG23_VP8
Address: Operational Base + offset (0x005c)
Base address for reference picture index 9

Bit	Attr	Reset Value	Description
31:2	RW	0x00000000	sw_dct_strm2_base base address for VP7/VP8 DCT stream MB row 2,2n+2 0 30 R/
1:0	RO	0x0	reserved

VDPU_SWREG24
Address: Operational Base + offset (0x0060)
Base address for reference picture index 10

Bit	Attr	Reset Value	Description
31:2	RW	0x00000000	sw_refer10_base Base address for reference picture index 10. See picture index definition from toplevel_sp
1	RW	0x0	sw_refer10_field_e Refer picture consist of single fields or frame: '0' = reference picture consists of frame '1' = reference picture consists of fields
0	RW	0x0	sw_refer10_top_e Which field of reference picture is closer to current picture: '0' = bottom field is closer to current picture '1' = top field is closer to current picture

VDPU_SWREG24_VP8
Address: Operational Base + offset (0x0060)
Base address for reference picture index 10

Bit	Attr	Reset Value	Description
31:2	RW	0x00000000	sw_dct_strm3_base base address for VP7/VP8 DCT stream MB row 3,2n+3 0 30 R/W
1:0	RO	0x0	reserved

VDPU_SWREG25_VP8
Address: Operational Base + offset (0x0064)
Base address for reference picture index 10

Bit	Attr	Reset Value	Description
31:2	RW	0x00000000	sw_dct_strm4_base base address for VP7/VP8 DCT stream MB row 4,2n+4
1:0	RO	0x0	reserved

VDPU_SWREG25
Address: Operational Base + offset (0x0064)
Base address for reference picture index 11

Bit	Attr	Reset Value	Description
31:3	RO	0x0	reserved

Bit	Attr	Reset Value	Description
2	RW	0x0	sw_refer11_base Base address for reference picture index 11. See picture index definition from toplevel_sp
1	RW	0x0	sw_refer11_field_e Refer picture consist of single fields or frame: '0' = reference picture consists of frame '1' = reference picture consists of fields
0	RW	0x0	sw_refer11_topc_e Which field of reference picture is closer to current picture: '0' = bottom field is closer to current picture '1' = top field is closer to current picture

VDPU_SWREG26_VP8

Address: Operational Base + offset (0x0068)

Base address for reference picture index 10

Bit	Attr	Reset Value	Description
31:2	RW	0x00000000	sw_dct_strm5_base base address for VP7/VP8 DCT stream MB row 5,2n+5
1:0	RO	0x0	reserved

VDPU_SWREG26

Address: Operational Base + offset (0x0068)

Base address for reference picture index 12

Bit	Attr	Reset Value	Description
31:2	RW	0x00000000	sw_refer12_base Base address for reference picture index 12. See picture index definition from toplevel_sp
1	RW	0x0	sw_refer12_field_e Refer picture consist of single fields or frame: '0' = reference picture consists of frame '1' = reference picture consists of fields
0	RW	0x0	sw_refer12_topc_e Which field of reference picture is closer to current picture: '0' = bottom field is closer to current picture '1' = top field is closer to current picture

VDPU_SWREG27_VP8

Address: Operational Base + offset (0x006c)

Base address for reference picture index 13

Bit	Attr	Reset Value	Description
31:2	RW	0x00000000	sw_bitpl_ctrl_base Base address for ctrl data stream. Used if multistream is enabled
1:0	RO	0x0	reserved

VDPU_SWREG27
Address: Operational Base + offset (0x006c)
Base address for reference picture index 13

Bit	Attr	Reset Value	Description
31:2	RW	0x00000000	sw_refer13_base Base address for reference picture index 13. See picture index definition from toplevel_sp
1	RW	0x0	sw_refer13_field_e Refer picture consist of single fields or frame: '0' = reference picture consists of frame '1' = reference picture consists of fields
0	RW	0x0	sw_refer13_topc_e Which field of reference picture is closer to current picture: '0' = bottom field is closer to current picture '1' = top field is closer to current picture

VDPU_SWREG28_VP8
Address: Operational Base + offset (0x0070)
Base address for reference picture index 10

Bit	Attr	Reset Value	Description
31:2	RW	0x00000000	sw_dct_strm6_base base address for VP7/VP8 DCT stream MB row 6,2n+6
1:0	RO	0x0	reserved

VDPU_SWREG28
Address: Operational Base + offset (0x0070)
Base address for reference picture index14

Bit	Attr	Reset Value	Description
31:2	RW	0x00000000	sw_refer14_base Base address for reference picture index 14. See picture index definition from toplevel_sp
1	RW	0x0	sw_refer14_field_e Refer picture consist of single fields or frame: '0' = reference picture consists of frame '1' = reference picture consists of fields
0	RW	0x0	sw_refer14_topc_e Which field of reference picture is closer to current picture: '0' = bottom field is closer to current picture '1' = top field is closer to current picture

VDPU_SWREG29_VP8
Address: Operational Base + offset (0x0074)
Base address for reference picture index 10

Bit	Attr	Reset Value	Description
31:2	RW	0x00000000	sw_dct_strm7_base base address for VP7/VP8 DCT stream MB row 7,2n+7

Bit	Attr	Reset Value	Description
1:0	RO	0x0	reserved

VDPU_SWREG29
Address: Operational Base + offset (0x0074)
Base address for reference picture index15

Bit	Attr	Reset Value	Description
31:2	RW	0x00000000	sw_refer15_base Base address for reference picture index 15. See picture index definition from toplevel_sp
1	RW	0x0	sw_refer15_field_e Refer picture consist of single fields or frame: '0' = reference picture consists of frame '1' = reference picture consists of fields
0	RW	0x0	sw_refer15_topc_e Which field of reference picture is closer to current picture: '0' = bottom field is closer to current picture '1' = top field is closer to current picture

VDPU_SWREG30
Address: Operational Base + offset (0x0078)
Reference picture numbers for index 0 and 1 (H264 VLC

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	sw_refer1_nbr Number for reference picture index 1
15:0	RW	0x0000	sw_refer0_nbr Number for reference picture index 0

VDPU_SWREG30_VP8
Address: Operational Base + offset (0x0078)
Reference picture numbers for index 0 and 1 (H264 VLC

Bit	Attr	Reset Value	Description
31:28	RO	0x0	reserved
27:21	RW	0x00	sw_filt_mb_adj_0 filter level adjustment for MB type 0
20:14	RW	0x00	sw_filt_mb_adj_1 filter level adjustment for MB type 1
13:7	RW	0x00	sw_filt_mb_adj_2 filter level adjustment for MB type 2
6:0	RW	0x00	sw_filt_mb_adj_3 filter level adjustment for MB type 3

VDPU_SWREG31
Address: Operational Base + offset (0x007c)
Reference picture numbers for index 2 and 3 (H264 VLC) /

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	sw_refer3_nbr Number for reference picture index 3
15:0	RW	0x0000	sw_refer2_nbr Number for reference picture index 2

VDPU_SWREG31_VP8

Address: Operational Base + offset (0x007c)

Reference picture numbers for index 2 and 3 (H264 VLC) /

Bit	Attr	Reset Value	Description
31:28	RO	0x0	reserved
27:21	RW	0x00	sw_filt_ref_adj_0 filter level adjustment for reference frame type 0
20:14	RW	0x00	sw_filt_ref_adj_1 filter level adjustment for reference frame type 1
13:7	RW	0x00	sw_filt_ref_adj_2 filter level adjustment for reference frame type 2
6:0	RW	0x00	sw_filt_ref_adj_3 filter level adjustment for reference frame type 3

VDPU_SWREG32

Address: Operational Base + offset (0x0080)

Reference picture numbers for index 4 and 5 (H264 VLC)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	sw_refer5_nbr Number for reference picture index 5
15:0	RW	0x0000	sw_refer4_nbr Number for reference picture index 4

VDPU_SWREG32_VP8

Address: Operational Base + offset (0x0080)

Reference picture numbers for index 4 and 5 (H264 VLC)

Bit	Attr	Reset Value	Description
31:24	RO	0x0	reserved
23:18	RW	0x00	sw_filt_level_0 filter level value for reference frame type 0
17:12	RW	0x00	sw_filt_level_1 filter level value for reference frame type 1
11:6	RW	0x00	sw_filt_level_2 filter level value for reference frame type 2
5:0	RW	0x00	sw_filt_level_3 filter level value for reference frame type 3

VDPU_SWREG33

Address: Operational Base + offset (0x0084)

Reference picture numbers for index 6 and 7 (H264 VLC)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	sw_refer7_nbr Number for reference picture index 7
15:0	RW	0x0000	sw_refer6_nbr Number for reference picture index 6

VDPU_SWREG33_VP8

Address: Operational Base + offset (0x0084)

Reference picture numbers for index 6 and 7 (H264 VLC)

Bit	Attr	Reset Value	Description
31:27	RW	0x00	sw_quant_delta_0 quantisizer delta 0 0
26:22	RW	0x00	sw_quant_delta_1 quantisizer delta 1 0
21:11	RW	0x000	sw_quant_0 quantisizer value for LUT (7 bit)
10:0	RW	0x000	sw_quant_1 quantisizer value for LUT (7 bit)

VDPU_SWREG34

Address: Operational Base + offset (0x0088)

Reference picture numbers for index 8 and 9 (H264 VLC)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	sw_refer9_nbr Number for reference picture index 9
15:0	RW	0x0000	sw_refer8_nbr Number for reference picture index 8

VDPU_SWREG34_VP8

Address: Operational Base + offset (0x0088)

Reference picture numbers for index 8 and 9 (H264 VLC)

Bit	Attr	Reset Value	Description
31:22	RW	0x000	sw_pred_bc_tap_0_3 Field0000 Abstract Prediction filter set 0, tap 3
21:12	RW	0x000	sw_pred_bc_tap_1_0 Prediction filter set 1, tap 0
11:2	RW	0x000	sw_pred_bc_tap_1_1 Prediction filter set 1, tap 1
1:0	RO	0x0	reserved

VDPU_SWREG35_JPEG_ROI

Address: Operational Base + offset (0x008c)

JPEG roi offest/dc base address

Bit	Attr	Reset Value	Description
31:2	RW	0x00000000	sw_jpegdcoff_base JPEG roi offset/dc base address JPEG roi offest/dc base address
1:0	RO	0x0	reserved

VDPU_SWREG35

Address: Operational Base + offset (0x008c)

Reference picture numbers for index 10 and 11 (H264 VLC)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	sw_refer11_nbr Number for reference picture index 11
15:0	RW	0x0000	sw_refer10_nbr Number for reference picture index 10

VDPU_SWREG35_VP8

Address: Operational Base + offset (0x008c)

Reference picture numbers for index 10 and 11 (H264 VLC)

Bit	Attr	Reset Value	Description
31:22	RW	0x000	sw_pred_bc_tap_1_2 prection filter set 1, tap 2
21:12	RW	0x000	sw_pred_bc_tap_1_3 prection filter set 1, tap 3
11:2	RW	0x000	sw_pred_bc_tap_2_0 prection filter set 2, tap 0
1:0	RO	0x0	reserved

VDPU_SWREG36

Address: Operational Base + offset (0x0090)

Reference picture numbers for index 12 and 13 (H264 VLC)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	sw_refer13_nbr Number for reference picture index 13
15:0	RW	0x0000	sw_refer12_nbr Number for reference picture index 12

VDPU_SWREG36_JPEG_ROI

Address: Operational Base + offset (0x0090)

JPEG roi offset/dc length

Bit	Attr	Reset Value	Description
31:17	RO	0x0	reserved
16:0	RW	0x00000	sw_jpegdcoff_len sw_jpegdcoff_len The number of 64bit jpegdcoff, it can be used both when sw_roi_decode is 1'b0 or 1'b1

VDPU_SWREG36_VP8

Address: Operational Base + offset (0x0090)

Reference picture numbers for index 12 and 13 (H264 VLC)

Bit	Attr	Reset Value	Description
31:22	RW	0x000	sw_pred_bc_tap_2_1 predction filter set 2, tap 1
21:12	RW	0x000	sw_pred_bc_tap_2_2 predction filter set 2, tap 2
11:2	RW	0x000	sw_pred_bc_tap_2_3 predction filter set 2, tap 3
1:0	RO	0x0	reserved

VDPU_SWREG37_VP8

Address: Operational Base + offset (0x0094)

Reference picture numbers for index 12 and 13 (H264 VLC)

Bit	Attr	Reset Value	Description
31:22	RW	0x000	sw_pred_bc_tap_3_2 predction filter set 3, tap 2
21:12	RW	0x000	sw_pred_bc_tap_3_1 predction filter set 3, tap 1
11:2	RW	0x000	sw_pred_bc_tap_3_0 predction filter set 3, tap 0
1:0	RO	0x0	reserved

VDPU_SWREG37

Address: Operational Base + offset (0x0094)

Reference picture numbers for index 14 and 15 (H264 VLC)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	sw_refer15_nbr Number for reference picture index 15
15:0	RW	0x0000	sw_refer14_nbr Number for reference picture index 14

VDPU_SWREG38

Address: Operational Base + offset (0x0098)

Reference picture long term flags (H264 VLC) / VPx prediction filt

Bit	Attr	Reset Value	Description
31:22	RW	0x000	sw_pred_bc_tap_3_3 Prediction filter set 3, tap 3
21:12	RW	0x000	sw_pred_bc_tap_4_0 Prediction filter set 4, tap 0
11:2	RW	0x000	sw_perd_bc_tap_4_1 Prediction filter set 4, tap 1
1:0	RO	0x0	reserved

VDPU_SWREG38_H264
Address: Operational Base + offset (0x0098)
Reference picture numbers for index 12 and 13 (H264 VLC)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	sw_refer_lterm_e long term flag for reference picture index [31:0]

VDPU_SWREG39
Address: Operational Base + offset (0x009c)
Reference picture valid flags (H264 VLC) /VPx prediction filter ta

Bit	Attr	Reset Value	Description
31:22	RW	0x000	sw_pred_bc_tap_4_2 Prediction filter set 4, tap 2
21:12	RW	0x000	sw_pred_bc_tap_4_3 Prediction filter set 4, tap 3
11:2	RW	0x000	sw_pred_bc_tap_5_0 Prediction filter set 5, tap 0
1:0	RO	0x0	reserved

VDPU_SWREG39_H264
Address: Operational Base + offset (0x009c)
Reference picture numbers for index 12 and 13 (H264 VLC)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	sw_refer_valid_e valid flag for reference picture index [31:0]

VDPU_SWREG40
Address: Operational Base + offset (0x00a0)
Base address for standard dependent tables

Bit	Attr	Reset Value	Description
31:2	RW	0x00000000	sw_qtable_base Base address for standard dependent tables: JPEG= AC,DC, QP tables MPEG4=QP table base address if type 1 quantization is used MPEG2=QP table base address H.264=base address for various tables VP7,VP8=base address for stream decoding tables RV=base address for picture slice sizes [note]:the h264 and vp8 decoder will use these bits.
1:0	RO	0x0	reserved

VDPU_SWREG41
Address: Operational Base + offset (0x00a4)
Base address for direct mode motion vectors

Bit	Attr	Reset Value	Description
31:2	RW	0x00000000	sw_dir_mv_base Direct mode motion vector write/read base address. For H264 this is used only for direct mode motion vector write base. Progressive JPEG: ACDC coefficient read/write base address. If current round is for DC components this base address is pointing to luminance (separate base adderesses for chrominances), for AC component rounds this base is used for current type
1:0	RO	0x0	reserved

VDPU_SWREG42_VP8
Address: Operational Base + offset (0x00a8)

Bit	Attr	Reset Value	Description
31:22	RW	0x000	sw_pred_bc_tap_5_1 predction filter set 5, tap 1
21:12	RW	0x000	sw_pred_bc_tap_5_2 predction filter set 5, tap 2
11:2	RW	0x000	sw_pred_bc_tap_5_3 predction filter set 5, tap 3
1:0	RO	0x0	reserved

VDPU_SWREG42
Address: Operational Base + offset (0x00a8)
bi_dir initial ref pic list register (0-2)

Bit	Attr	Reset Value	Description
31:30	RO	0x0	reserved
29:25	RW	0x00	sw_binit_rlist_b2 Initial reference picture list for bi- direct backward picid 2
24:20	RW	0x00	sw_binit_rlist_f2 Initial reference picture list for bi- direct forward picid 2
19:15	RW	0x00	sw_binit_rlist_b1 Initial reference picture list for bi- direct backward picid 1
14:10	RW	0x00	sw_binit_rlist_f1 Initial reference picture list for bi- direct forward picid 1
9:5	RW	0x00	sw_binit_rlist_b0 Initial reference picture list for bi- direct backward picid 0
4:0	RW	0x00	sw_binit_rlist_f0 Initial reference picture list for bi- direct forward picid 0

VDPU_SWREG43_VP8
Address: Operational Base + offset (0x00ac)

Bit	Attr	Reset Value	Description
31:22	RW	0x000	sw_pred_bc_tap_6_0 predction filter set 6, tap 0

Bit	Attr	Reset Value	Description
21:12	RW	0x000	sw_pred_bc_tap_6_1 predction filter set 6, tap 1
11:2	RW	0x000	sw_pred_bc_tap_6_2 predction filter set 6, tap 2
1:0	RO	0x0	reserved

VDPU_SWREG43

Address: Operational Base + offset (0x00ac)

bi-dir initial ref pic list register (3-5)

Bit	Attr	Reset Value	Description
31:30	RO	0x0	reserved
29:25	RW	0x00	sw_binit_rlist_b5 Initial reference picture list for bi- direct backward picid 5
24:20	RW	0x00	sw_binit_rlist_f5 Initial reference picture list for bi- direct forward picid 5
19:15	RW	0x00	sw_binit_rlist_b4 Initial reference picture list for bi- direct backward picid 4
14:10	RW	0x00	sw_binit_rlist_f4 Initial reference picture list for bi- direct forward picid 4
9:5	RW	0x00	sw_binit_rlist_b3 Initial reference picture list for bi- direct backward picid 3
4:0	RW	0x00	sw_binit_rlist_f3 Initial reference picture list for bi- direct forward picid 3

VDPU_SWREG44_VP8

Address: Operational Base + offset (0x00b0)

Bit	Attr	Reset Value	Description
31:22	RW	0x000	sw_pred_bc_tap_6_3 predction filter set 6, tap 3
21:12	RW	0x000	sw_pred_bc_tap_7_0 predction filter set 7, tap 0
11:2	RW	0x000	sw_pred_bc_tap_7_1 predction filter set 7, tap 1
1:0	RO	0x0	reserved

VDPU_SWREG44

Address: Operational Base + offset (0x00b0)

bi-dir initial ref pic list register (6-8)

Bit	Attr	Reset Value	Description
31:30	RO	0x0	reserved
29:25	RW	0x00	sw_binit_rlist_b8 Initial reference picture list for bi- direct backward picid 8

Bit	Attr	Reset Value	Description
24:20	RW	0x00	sw_binit_rlist_f8 Initial reference picture list for bi- direct forward picid 8
19:15	RW	0x00	sw_binit_rlist_b7 Initial reference picture list for bi- direct backward picid 7
14:10	RW	0x00	sw_binit_rlist_f7 Initial reference picture list for bi- direct forward picid 7
9:5	RW	0x00	sw_binit_rlist_b6 Initial reference picture list for bi- direct backward picid 6
4:0	RW	0x00	sw_binit_rlist_f6 Initial reference picture list for bi- direct forward picid 6

VDPU_SWREG45_VP8
Address: Operational Base + offset (0x00b4)

Bit	Attr	Reset Value	Description
31:22	RW	0x000	sw_pred_bc_tap_7_2 predction filter set 7, tap 2
21:12	RW	0x000	sw_pred_bc_tap_7_3 predction filter set 7, tap 3
11:10	RW	0x0	sw_pred_tap_2_m1 additional Prediction filter tap -1 for set 2
9:8	RW	0x0	sw_pred_tap_2_4 Field0000 Abstract additional Prediction filter tap 4 for set 2
7:6	RW	0x0	sw_pred_tap_4_m1 additional Prediction filter tap -1 for set 4
5:4	RW	0x0	sw_pred_tap_4_4 Field0000 Abstract additional Prediction filter tap 4 for set 4
3:2	RW	0x0	sw_pred_tap_6_m1 additional Prediction filter tap -1 for set 6
1:0	RW	0x0	sw_pred_tap_6_4 Field0000 Abstract additional Prediction filter tap 4 for set 6

VDPU_SWREG45
Address: Operational Base + offset (0x00b4)
bi-dir initial ref pic list register (9- 11)

Bit	Attr	Reset Value	Description
31:30	RO	0x0	reserved
29:25	RW	0x00	sw_binit_rlist_b11 Initial reference picture list for bi- direct backward picid 11
24:20	RW	0x00	sw_binit_rlist_f11 Initial reference picture list for bi- direct forward picid 11

Bit	Attr	Reset Value	Description
19:15	RW	0x00	sw_binit_rlist_b10 Initial reference picture list for bi- direct backward picid 10
14:10	RW	0x00	sw_binit_rlist_f10 Initial reference picture list for bi- direct forward picid 10
9:5	RW	0x00	sw_binit_rlist_b9 Initial reference picture list for bi- direct backward picid 9
4:0	RW	0x00	sw_binit_rlist_f9 Initial reference picture list for bi- direct forward picid 9

VDPU_SWREG46
Address: Operational Base + offset (0x00b8)
bi-dir initial ref pic list register (12- 14) / VP7,VP8 quantization v

Bit	Attr	Reset Value	Description
31:30	RO	0x0	reserved
29:25	RW	0x00	sw_binit_rlist_b14 Initial reference picture list for bi- direct backward picid 14
24:20	RW	0x00	sw_binit_rlist_f14 Initial reference picture list for bi- direct forward picid 14
19:15	RW	0x00	sw_binit_rlist_b13 Initial reference picture list for bi- direct backward picid 13
14:10	RW	0x00	sw_binit_rlist_f13 Initial reference picture list for bi- direct forward picid 13
9:5	RW	0x00	sw_binit_rlist_b12 Initial reference picture list for bi- direct backward picid 12
4:0	RW	0x00	sw_binit_rlist_f12 Initial reference picture list for bi- direct forward picid 12

VDPU_SWREG46_VP8
Address: Operational Base + offset (0x00b8)
bi-dir initial ref pic list register (12- 14) / VP7,VP8 quantization v

Bit	Attr	Reset Value	Description
31:27	RW	0x00	sw_quant_delta_2 quantisizer delta 2
26:22	RW	0x00	sw_quant_delta_3 quantisizer delta 3
21:11	RW	0x000	sw_quant_2 quantisizer value for LUT (7 bit)
10:0	RW	0x000	sw_quant_3 quantisizer value for LUT (7 bit)

VDPU_SWREG47
Address: Operational Base + offset (0x00bc)
bi-dir and P fwd initial ref pic list register (15 and P 0-3) / VP7,V

Bit	Attr	Reset Value	Description
31:30	RO	0x0	reserved
29:25	RW	0x00	sw_pinit_rlist_f3 Initial reference picture list for P forward picid 3
24:20	RW	0x00	sw_pinit_rlist_f2 Initial reference picture list for P forward picid 2
19:15	RW	0x00	sw_pinit_rlist_f1 Initial reference picture list for P forward picid 1
14:10	RW	0x00	sw_pinit_rlist_f0 Initial reference picture list for P forward picid 0
9:5	RW	0x00	sw_binit_rlist_b15 Initial reference picture list for bi- direct backward picid 15
4:0	RW	0x00	sw_binit_rlist_f15 Initial reference picture list for bi- direct forward picid 15

VDPU_SWREG47_VP8

Address: Operational Base + offset (0x00bc)

bi-dir and P fwd initial ref pic list register (15 and P 0-3) / VP7,V

Bit	Attr	Reset Value	Description
31:27	RW	0x00	sw_quant_delta_4 quantisizer delta 4
26:0	RO	0x0	reserved

VDPU_SWREG48

Address: Operational Base + offset (0x00c0)

Error concealment register

Bit	Attr	Reset Value	Description
31:23	RW	0x000	sw_startmb_x Start MB from SW for X dimension. Used in error concealment case [note]:the h264 and vp8 decoder will use these bits.
22:15	RW	0x00	sw_startmb_y Start MB from SW for Y dimension. Used in error concealment case [note]:the h264 and vp8 decoder will use these bits.
14:0	RO	0x0	reserved

VDPU_SWREG49

Address: Operational Base + offset (0x00c4)

Prediction filter tap register for H264, MPEG4, VC1

Bit	Attr	Reset Value	Description
31:22	RW	0x000	sw_pred_bc_tap_0_0 Prediction filter set 0, tap 0 [note]:the h264 decoder will use these bits.

Bit	Attr	Reset Value	Description
21:12	RW	0x000	sw_pred_bc_tap_0_1 Prediction filter set 0, tap 1 [note]:the h264 decoder will use these bits.
11:2	RW	0x000	sw_pred_bc_tap_0_2 Prediction filter set 0, tap 2 [note]:the h264 decoder will use these bits.
1:0	RO	0x0	reserved

VDPU_SWREG50

Address: Operational Base + offset (0x00c8)

Synthesis configuration register decoder 0 (read only)

Bit	Attr	Reset Value	Description
31	RO	0x1	SW_DEC_MPEG2_PROF Decoding format support, MPEG-2 / MPEG-1 '0' = not supported '1' = supported
30:29	RO	0x3	SW_DEC_VC1_PROF Decoding format support, VC-1 2'd0 = not supported 2'd1 = supported up to simple profile 2'd2 = supported up to main profile 2'd3 = supported up to advanced profile
28	RO	0x1	SW_DEC_JPEG_PROF Decoding format support, JPEG 0 = not supported 1 = supported
27:26	RO	0x2	SW_DEC_MPEG4_PROF Decoding format support, MPEG-4 / H.263 2'd0 = not supported 2'd1 = supported up to simple profile 2'd2 = supported up to advanced simple profile
25:24	RO	0x3	SW_DEC_H264_PROF Decoding format support, H.264 2'd0 = not supported 2'd1 = supported up to baseline profile 2'd2 = supported up to high profile labeled stream with restricted high profile tools [note]:the h264 decoder will use these bits.
22	RO	0x0	SW_DEC_PJEPEG_EXIT Progressive JPEG support: '0' = Not supported '1' = supported

Bit	Attr	Reset Value	Description
21	RO	0x1	SW_DEC_OBUFF_LEVEL Decoder output buffer level: '0' = 1 MB buffering is used '1' = 4 MB buffering is used [note]:the h264 and vp8 decoder will use these bits.
20	RO	0x1	SW_REF_BUFF_EXIST [note]:the h264 and vp8 decoder will use these bits.
19:16	RO	0x5	SW_DEC_BUS_STRD [note]:the h264 and vp8 decoder will use these bits.
15:14	RO	0x1	SW_DEC_SYNTH_LAN [note]:the h264 and vp8 decoder will use these bits.
13:12	RO	0x2	SW_DEC_BUS_WIDTH 2'd0 = error 2'd1 = 32 bit bus 2'd2 = 64 bit bus 2'd3 = 128 bit bus [note]:the h264 and vp8 decoder will use these bits.
10:0	RO	0x780	SW_DEC_MAX_OWIDTH Max configured decoder video resolution that can be decoded. Informed as width of the picture in pixels [note]:the h264 and vp8 decoder will use these bits.

VDPU_SWREG51
Address: Operational Base + offset (0x00cc)
Reference picture buffer control register

Bit	Attr	Reset Value	Description
31	RW	0x0	sw_refbu_e Refer picture buffer enable: '0' = refer picture buffer disabled '1' = refer picture buffer enabled. Valid if picture size is QVGA or more [note]:the h264 and vp8 decoder will use these bits.
30:19	RW	0x000	sw_refbu_thr Reference buffer disable threshold value (cache miss amount). Used to buffer shut down (if more misses than allowed) [note]:the h264 and vp8 decoder will use these bits.
18:14	RW	0x00	sw_refbu_picid The used reference picture ID for reference buffer usage [note]:the h264 and vp8 decoder will use these bits.

Bit	Attr	Reset Value	Description
13	RW	0x0	sw_refbu_eval_e Enable for HW internal reference ID calculation. If given threshold level is reached by any picture_id after first MB row, that picture_id is used for reference buffer fill for rest of the picture [note]:the h264 and vp8 decoder will use these bits.
12	RW	0x0	sw_refbu_fparmod_e Field parity mode enable. Used in refbufferd evaluation mode '0' = use the result field of the evaluation '1' = use the parity mode field [note]:the h264 and vp8 decoder will use these bits.
11:9	RO	0x0	reserved
8:0	RW	0x000	sw_refbu_y_offset Y offset for refbufferd. This coordinate is used to compensate the global motion of the video for better buffer hit rate [note]:the h264 and vp8 decoder will use these bits.

VDPU_SWREG52
Address: Operational Base + offset (0x00d0)
Reference picture buffer information register 1 (read only)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	sw_refbu_hit_sum The sum of the refbufferd hits of the picture. Determined for each 8x8 luminance partition of the picture. The proceeding of the HW calculation can be read during HW decoding [note]:the h264 and vp8 decoder will use these bits.
15:0	RW	0x0000	sw_refbu_intra_sum The sum of the luminance 8x8 intra partitons of the picture. The proceeding of the HW calculation can be read during HW decoding [note]:the h264 and vp8 decoder will use these bits.

VDPU_SWREG53
Address: Operational Base + offset (0x00d4)
Reference picture buffer information register 2 (read only)

Bit	Attr	Reset Value	Description
31:22	RO	0x0	reserved
21:0	RW	0x000000	sw_refbu_y_mv_sum The sum of the decoded motion vector y-components of the picture. The first luminance motion vector of each MB is used in calculation. Other motion vectors of the MB are discarded. Each motion vector is saturated between -256 - 255 before calculation. The proceeding of the HW calculation can be read during HW decoding [note]:the h264 and vp8 decoder will use these bits.

VDPU_SWREG54
Address: Operational Base + offset (0x00d8)
Synthesis configuration register decoder 1 (read only)

Bit	Attr	Reset Value	Description
31	RO	0x1	SW_DEC_JPEG_EXTENS JPEG sampling support extension for 411 and 444 samplings and support for bigger max resolution than 16 Mpix (up to 67Mpixels): '0' = not supported '1' = supported
30	RO	0x1	SW_DEC_REFBU_ILACE Refbufferd support for interlaced content: '0' = not supported '1' = supported [note]:the h264 decoder will use these bits.
28	RO	0x0	SW_REF_BUFF2_EXIST Reference picture buffer 2 usage: '0' = not supported '1' = reference buffer 2 is used [note]:the h264 and vp8 decoder will use these bits.
27:26	RO	0x1	SW_DEC_RV_PROF Decoding format support, RV 2'd0 = not supported 2'd1 = supported up to 2'd2 = NA
25	RO	0x0	SW_DEC_RTL_ROM ROM implementation type (If design includes ROMs) '0': ROMs are implemented from actual ROM units '1': ROMs are impelemted from RTL
24	RO	0x1	SW_DEC_VP7_PROF Decoding format support, VP7 0 = not supported 1 = supported
23	RO	0x1	SW_DEC_VP8_PROF Decoding format support, VP8 0 = not supported 1 = supported [note]:the vp8 decoder will use these bits.
22	RO	0x1	SW_DEC_AVS_PROF Decoding format support, AVS 0 = not supported 1 = supported

Bit	Attr	Reset Value	Description
21:20	RO	0x1	SW_DEC_MVC_PROF Decoding format support, MVC 0 = not supported 1 = supported
19	RO	0x1	SW_DEC_VP8SNAP_E Decoding format support, VP8 snapshot 0 = not supported bigger than 1080p resolution 1 = supported upto 16kx16k pixel resolution (defined max) [note]:the vp8 decoder will use these bits.
18:17	RO	0x1	SW_DEC_TILED_L Tiled mode support level 2'd0 = not supported 2'd1 = supported with 8x4 tile size 2'd2, 2'd3 = reserved [note]:the h264 and vp8 decoder will use these bits.
16:0	RO	0x0	reserved

VDPU_SWREG55
Address: Operational Base + offset (0x00dc)
Reference picture buffer 2 / Advanced prefetch control register

Bit	Attr	Reset Value	Description
31	RW	0x0	sw_refbu2_buf_e Refer picture buffer 2 enable: '0' = refer picture buffer disabled '1' = refer picture buffer enabled. Valid if picture size is QVGA or more (can be turned of by HW if threshold value reached) [note]:the h264 and vp8 decoder will use these bits.
30:19	RW	0x000	sw_refbu2_thr Reference buffer disable threshold value (buffer miss amount). Used to buffer shut down (if more misses than allowed) [note]:the h264 and vp8 decoder will use these bits.
18:14	RW	0x00	sw_refbu2_picid The used reference picture ID for reference buffer usage [note]:the h264 and vp8 decoder will use these bits.
13:0	RW	0x0000	sw_apf_threshold Advanced prefetch threshold value. If current MB exceeds the threshold the advanced mode is not used. Value 0 disables threshold usage and advanced prefetch usage is restricted by internal memory limitation only [note]:the h264 and vp8 decoder will use these bits.

VDPU_SWREG56
Address: Operational Base + offset (0x00e0)
Reference buffer information register 3 (read only)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	sw_refbu_top_sum The sum of the top partitions of the picture [note]:the h264 and vp8 decoder will use these bits.
15:0	RW	0x0000	sw_refbu_bot_sum The sum of the bottom partitions of the picture [note]:the h264 and vp8 decoder will use these bits.

VDPU_SWREG57_INTRA_INTER

Address: Operational Base + offset (0x00e4)

intra_dll3t,intra_dblspeed,inter_dblspeed,stream_len_hi

Bit	Attr	Reset Value	Description
31:15	RO	0x0	reserved
14:8	RO	0x00	debug_service debug_service signals service_wr[2:0], service_rd[3:0]
7	RW	0x0	sw_cache_en cache enable 1'b1: cache enable 1'b0: cache disable when sw_cache_en is 1'b1, sw_pref_sigchan should also be 1'b1
6	RW	0x0	sw_pref_sigchan prefetch single channel enable 1'b1: prefetch single channel enable
5	RW	0x0	sw_axiwr_sel axi write master select 1'b0: auto sel encoder axi signals and decoder axi signals 1'b1: sel decoder axi signals (it only use to set bu_dec_e to 1'b0 in the middle of a frame)
4	RW	0x0	sw_paral_bus paral_bus enable when it is set to 1'b1, the axi support read and write service parallel; when it is set to 1'b0, the axi only support read and write serial
3	RW	0x0	sw_intra_dbl3t sw_intra_dbl3t In chroma dc intra prediction, when this bit is enable, there will 3 cycle enhance for every block
2	RW	0x0	sw_intra_dblspeed intra double speed enable Intra double speed enable
1	RW	0x0	sw_inter_dblspeed inter double speed enable Inter double speed enable

Bit	Attr	Reset Value	Description
0	RW	0x0	sw_stream_len_hi stream length high bit The extension bit of sw_stream_len

VDPU_SWREG57

Address: Operational Base + offset (0x00e4)

intra_dll3t,intra_dblspeed,inter_dblspeed,stream_len_hi

Bit	Attr	Reset Value	Description
31	RW	0x0	fuse_dec_h264 1 = H.264 enabled
30	RW	0x0	fuse_dec_mpeg4 1= MPEG-4/H.263 enabled
29	RW	0x0	fuse_dec_mpeg2 1 = MPEG-2/MPEG-1 enabled N
27	RW	0x0	fuse_dec_jpeg Field0000 Abstract 1= JPEG enabled
25	RW	0x0	fuse_dec_vc1 1 = VC1 enabled
24	RW	0x0	fuse_dec_pjpeg 1 = Progressive JPEG enabled (Requires also JPEG to be enabled
22	RW	0x0	fuse_dec_rv 1= RV enabled
21	RW	0x0	fuse_dec_vp7 1= VP7 enabled
20	RW	0x0	fuse_dec_vp8 1= VP8 enabled
19	RW	0x0	fuse_dec_avs 1 = AVS eanbled
18	RW	0x0	fuse_dec_mvc enabled (requires also H264 to be enabled)
17:16	RO	0x0	reserved
15	RW	0x0	fuse_dec_maxw_1920 1 = Max video width up to 1920 pixels enabled. Priority coded with priority 1.
14	RW	0x0	fuse_dec_maxw_1280 1 = Max video width up to 1280 pixels enabled. Priority coded with priority 2.
13	RW	0x0	fuse_dec_maxw_720 1 = Max video width up to 720 pixels enabled. Priority coded with priority 3.
12	RW	0x0	fuse_dec_maxw_352 1 = Max video width up to 352 pixels enabled. Priority coded with priority 4

Bit	Attr	Reset Value	Description
11:8	RO	0x0	reserved
7	RW	0x0	fuse_dec_refbuffer 1 = reference buffer used
6:0	RO	0x0	reserved

VDPU_SWREG58
Address: Operational Base + offset (0x00e8)
Decoder debug register 0 (read only)

Bit	Attr	Reset Value	Description
31	RO	0x0	reserved
30	RO	0x0	debug_mv_req mvst_mv_req signal value
29	RO	0x0	debug_rlc_req prtr_res_y_req signal value
28	RO	0x0	debug_res_y_req prtr_res_y_req signal value
27	RO	0x0	debug_res_c_req prtr_res_c_req signal value
26	RO	0x0	debug_strm_da_e strm_da_e signal value
25	RO	0x0	debug_framerdy dfbu_framerdy signal value
24	RO	0x0	debug_filter_req dfbu_req_e signal value
23	RO	0x0	debug_referreq0 prbu_referreq0 signal value
22	RO	0x0	debug_referreq1 prbu_referreq1 signal value
21	RO	0x0	reserved
20:0	RO	0x000000	debug_dec_mb_count HW internal MB counter value

VDPU_SWREG59
Address: Operational Base + offset (0x00ec)
H264 Chrominance 8 pixel interleaved data base

Bit	Attr	Reset Value	Description
31:2	RW	0x00000000	sw_dec_ch8pix_base Base address for additional chrominance data format where chrominance is interleaved in group of 8 pixels. The usage is enabled by sw_ch_8pix_ileav_e [note]:the h264 decoder will use these bits.
1:0	RO	0x0	reserved

VDPU_SWREG60
Address: Operational Base + offset (0x00f0)
Interrupt register post-processor

Bit	Attr	Reset Value	Description
31:14	RO	0x0	reserved
13	RW	0x0	sw_pp_bus_int Interrupt status bit bus. Error response from bus. In pipeline mode this bit is not used
12	RW	0x0	sw_pp_rdy_int Interrupt status bit pp. When this bit is high post processor has processed a picture in external mode. In pipeline mode this bit is not used.
11:9	RO	0x0	reserved
8	RW	0x0	sw_pp_irq Post-processor IRQ. SW will reset this after interrupt is handled. HINTpp is not used for pp if IRQ disable pp is high (sw_pp_irq_n_e = 1). In pipeline mode this bit is not used
7:5	RO	0x0	reserved
4	RW	0x0	sw_pp_irq_dis Post-processor IRQ disable. When high, there are no interrupts from HW concerning post processing. Polling must be used to see the interrupt
3:2	RO	0x0	reserved
1	RW	0x0	sw_pp_pipeline_e Decoder –post-processing pipeline enable: 0 = Post-processing is processing different picture than decoder or is disabled 1 = Post-processing is performed in pipeline with decoder
0	RW	0x0	sw_pp_e External mode post-processing enable. This bit will start the post-processing operation. Not to be used if PP is in pipeline with decoder (sw_pp_pipeline_e = 1). HW will reset this when picture is post-processed.

VDPU_SWREG61
Address: Operational Base + offset (0x00f4)
Device configuration register post-processor

Bit	Attr	Reset Value	Description
31:24	RW	0x01	sw_pp_axi_rd_id Read ID used for AXI PP read services (if connected to AXI)
23:16	RW	0x01	sw_pp_axi_wr_id Write ID used for AXI PP write services (if connected to AXI)
15	RO	0x0	reserved
14	RW	0x0	sw_pp_scmd_dis AXI Single Command Multiple Data disable.

Bit	Attr	Reset Value	Description
13	RW	0x0	sw_pp_in_a2_endsel Endian/swap select for Alpha blend input source 2: '0' = Use PP in endian/swap definitions (sw_pp_in_endian, sw_pp_in_swap) '1' = Use Ablend source 1 endian/swap definitions
12	RW	0x0	sw_pp_in_a1_swap32 Alpha blend source 1 input 32bit data swap (may be used for 64 bit environment): 0 = no swapping of 32 bit words 1 = 32 bit data words are swapped (needed in 64 bit environment to achieve 7-6-5-4-3-2-1-0 byte order(also little endian should be enabled))
11	RW	0x0	sw_pp_in_a1_endian Alpha blend source 1 input data byte endian mode. 0 = Big endian (0-1-2-3 order) 1 = Little endian (3-2-1-0 order)
10	RW	0x0	sw_pp_in_swap32_e PP input 32bit data swap (may be used for 64 bit environment): 0 = no swapping of 32 bit words 1 = 32 bit data words are swapped (needed in 64 bit environment to achieve 7-6-5-4-3-2-1-0 byte order(also little endian should be enabled))
9	RW	0x0	sw_pp_data_disc_e PP data discard enable. Precise burst lengths are used with reading services. Extra data is discarded internally.
8	RW	0x1	sw_pp_clkgate_e PP dynamic clock gating enable: 1 = Clock is gated from PP structures that are not used 0 = Clock is running for all PP structures Note: Clock gating value can be changed only when PP is not enabled
7	RW	0x0	sw_pp_in_endian PP input picture byte endian mode. Used only if PP is in standalone mode. If PP is running pipelined with the decoder, this bit has no effect. 0 = Big endian (0-1-2-3 order) 1 = Little endian (3-2-1-0 order)

Bit	Attr	Reset Value	Description
6	RW	0x0	sw_pp_out_endian PP output picture endian mode for YCbCr data or for any data if config value SW_PP_OEN_VERSION=1 0 = Big endian (0-1-2-3 order) 1 = Little endian (3-2-1-0 order) (NOTE: For SW_PP_OEN_VERSION=0 16 bit RGB data this bit works as pixel swapping bit. For 32 bit RGB this bit has no meaning)
5	RW	0x0	sw_pp_out_swap32_e PP output data word swap (may be used for 64 bit environment): 0 = no swapping of 32 bit words 1 = 32 bit data words are swapped (needed in 64 bit environment to achieve 7-6-5-4-3-2-1-0 byte order (also little endian should be enabled))
4:0	RW	0x00	sw_pp_max_burst Maximum burst length for PP bus transactions. 1-16

VDPU_SWREG62
Address: Operational Base + offset (0x00f8)
Deinterlace control register

Bit	Attr	Reset Value	Description
31	RW	0x0	sw_deint_e De-interlace enable. Input data is in interlaced format and deinterlacing needs to be performed
30	RO	0x0	reserved
29:16	RW	0x0000	sw_deint_threshold Threshold value used in deinterlacing
15	RW	0x0	sw_deint_blend_e Blend enable for de-interlacing
14:0	RW	0x0000	sw_deint_edge_det Edge detect value used for deinterlacing

VDPU_SWREG63
Address: Operational Base + offset (0x00fc)
base address for reading post-processing input picture uminan

Bit	Attr	Reset Value	Description
31:2	RW	0x00000000	sw_pp_in_lu_base Base address for post-processing input luminance picture. If PP input picture is fetched from fields this base address is used to point to topfield of the picture. Used in external mode only.
1:0	RO	0x0	reserved

VDPU_SWREG64
Address: Operational Base + offset (0x0100)
Base address for reading post-processing input picture Cb/Ch

Bit	Attr	Reset Value	Description
31:2	RW	0x00000000	sw_pp_in_cb_base Base address for post-processing input Cb picture or for both chrominance pictures (if chrominances interleaved). If PP input picture is fetched from fields this base address is used to point to topfield of the picture. Used in external mode only
1:0	RO	0x0	reserved

VDPU_SWREG65
Address: Operational Base + offset (0x0104)
Base address for reading post-processing input picture Cr

Bit	Attr	Reset Value	Description
31:2	RW	0x00000000	sw_pp_in_cr_base Base address for post-processing input cr picture. Used in external mode only
1:0	RO	0x0	reserved

VDPU_SWREG66
Address: Operational Base + offset (0x0108)
Base address for writing post-processed picture luminance/RGB

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	sw_pp_out_lu_base Base address for post-processing output picture (luminance/YUYV/RGB).

VDPU_SWREG67
Address: Operational Base + offset (0x010c)
Base address for writing post-processed picture Ch

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	sw_pp_out_ch_base Base address for post-processing output chrominance picture (interleaved chrominance).

VDPU_SWREG68
Address: Operational Base + offset (0x0110)
Register for contrast adjusting

Bit	Attr	Reset Value	Description
31:24	RW	0x00	sw_contrast_thr1 Threshold value 1, used with contrast adjusting
23:20	RO	0x0	reserved
19:10	RW	0x000	sw_contrast_off2 Offset value 2, used with contrast adjusting
9:0	RW	0x000	sw_contrast_off1 Offset value 1, used with contrast adjusting

VDPU_SWREG69
Address: Operational Base + offset (0x0114)
Register for colour conversion and contrast adjusting

Bit	Attr	Reset Value	Description
31	RW	0x0	sw_pp_in_start_ch For YUYV 422 input format. Enable for start_with_chrominance. '0' =the order is Y0CbY0Cr or Y0CrY0Cb '1'= the order is CbY0CrY0 or CrY0CbY0
30	RW	0x0	sw_pp_in_cr_first For YUYV 422 input format and YCbCr 420 semiplanar format. Enable for Cr first (before Cb) '0' =the order is Y0CbY0Cr or CbY0CrY0 (if 420 semiplanar chrominance: CbCrCbCr) '1'= the order is Y0CrY0Cb or CrY0CbY0 (if 420 semiplanar chrominance: CrCbCrCb)
29	RW	0x0	sw_pp_out_start_ch For YUYV 422 output format. Enable for start_with_chrominance. '0' =the order is Y0CbY0Cr or Y0CrY0Cb '1'= the order is CbY0CrY0 or CrY0CbY0
28	RW	0x0	sw_pp_out_cr_first For YUYV 422 output format. Enable for Cr first (beforeCb) '0' =the order is Y0CbY0Cr or CbY0CrY0 '1'= the order is Y0CrY0Cb or CrY0CbY0
27:18	RW	0x000	sw_color_coeffa2 Coefficient a2, used with Y pixel to calculate all color components
17:8	RW	0x000	sw_color_coeffa1 Coefficient a1, used with Y pixel to calculate all color components
7:0	RW	0x00	sw_contrast_thr2 Threshold value 2, used with contrast adjusting

VDPU_SWREG70
Address: Operational Base + offset (0x0118)
Register for colour conversion 0

Bit	Attr	Reset Value	Description
31:30	RO	0x0	reserved
29:20	RW	0x000	sw_color_coeffd Coefficient d, used with Cb to calculate green component value
19:10	RW	0x000	sw_color_coeffc Coefficient c, used with Cr to calculate green component value
9:0	RW	0x000	sw_color_coeffb Coefficient b, used with Cr to calculate red component value

VDPU_SWREG71
Address: Operational Base + offset (0x011c)
Register for colour conversion 1 + rotation mode

Bit	Attr	Reset Value	Description
31:30	RO	0x0	reserved
29:21	RW	0x000	sw_crop_startx Start coordinate x for the cropped area in macroblocks.
20:18	RW	0x0	sw_rotation_mode Rotation mode: 3'b000 = rotation disabled 3'b001 = rotate + 90 3'b010 = rotate –90 3'b011 = horizontal flip (mirror) 3'b100 = vertical flip 3'b101 = rotate 180
17:10	RW	0x00	sw_color_coefff Coefficient f, used with Y to adjust brightness
9:0	RW	0x000	sw_color_coeffe Coefficient e, used with Cb to calculate blue component value

VDPU_SWREG72
Address: Operational Base + offset (0x0120)
PP input size and -cropping register

Bit	Attr	Reset Value	Description
31:24	RW	0x00	sw_crop_starty Start coordinate y for the cropped area in macroblocks.
23	RO	0x0	reserved
22:18	RW	0x00	sw_rangemap_coef_y Range map value for Y component (RANGE_MAPY+9 in VC-1 standard)
17	RO	0x0	reserved
16:9	RW	0x00	sw_pp_in_height PP input picture height in MBs. Can be cropped from a bigger input picture in external mode
8:0	RW	0x000	sw_pp_in_width PP input picture width in MBs. Can be cropped from a bigger input picture in external mode

VDPU_SWREG73
Address: Operational Base + offset (0x0124)
PP input picture base address for Y bottom field

Bit	Attr	Reset Value	Description
31:2	RW	0x00000000	sw_pp_bot_yin_base PP input Y base for bottom field
1:0	RO	0x0	reserved

VDPU_SWREG74
Address: Operational Base + offset (0x0128)
PP input picture base for Ch bottom field

Bit	Attr	Reset Value	Description
31:2	RW	0x00000000	sw_pp_bot_cin_base PP input C base for bottom field (mixed chrominance)
1:0	RO	0x0	reserved

VDPU_SWREG79
Address: Operational Base + offset (0x013c)
Scaling ratio register 1 & padding for B

Bit	Attr	Reset Value	Description
31	RW	0x0	sw_rangemap_y_e Range map enable for Y component (RANGE_MAPY_FLAG in VC-1 standard). For VC1 main profile this bit is used as range expansion enable
30	RW	0x0	sw_rangemap_c_e Range map enable for chrominance component RANGE_MAPUV_FLAG in VC-1 standard)
29	RW	0x0	sw_ycbcr_range Defines the YCbCr range in RGB conversion: 0 =16 --> 235 for Y, 16 --> 240 for Chrominance 1 = 0 --> 255 for all components
28	RW	0x0	sw_rgb_pix_in32 RGB pixel amount/ 32 bit word 0 = 1 RGB pixel/32 bit 1 = 2 RGB pixels/32 bit
27:23	RW	0x00	sw_rgb_r_padd Amount of ones that will be padded in front of the R-component
22:18	RW	0x00	sw_rgb_g_padd Amount of ones that will be padded in front of the G-component
17:0	RW	0x00000	sw_scale_wratio Scaling ratio for width (outputw-1/inputw-1)

VDPU_SWREG80
Address: Operational Base + offset (0x0140)
Scaling register 0 ratio & padding for R and G

Bit	Attr	Reset Value	Description
31	RO	0x0	reserved
30	RW	0x0	sw_pp_fast_scale_e 0 = fast downscaling is not enabled 1 = fast downscaling is enabled. The quality of the picture is decreased but performance is improved

Bit	Attr	Reset Value	Description
29:27	RW	0x0	sw_pp_in_struct PP input data picture structure: 3'd0 = Top field / progressive frame structure: Read input data from top field base address /frame base address and read every line 3'd1 = Bottom field structure: Read input data from bottom field base address and read every line. 3'd2 = Interlaced field structure: Read input data from both top and bottom field base address and take every line from each field. 3'd3 = Interlaced frame structure: Read input data from both top and bottom field base address and take every second line from each field. 3'd4 = Ripped top field structure: Read input data from top field base address and read every second line. 3'd5 = Ripped bottom field structure: Read input data from bottom field base address and read every second line
26:25	RW	0x0	sw_hor_scale_mode Horizontal scaling mode: 2'b00 = Off 2'b01 = Upscale 2'b10 = Downscale
24:23	RW	0x0	sw_ver_scale_mode Vertical scaling mode: 2'b00 = Off 2'b01 = Upscale 2'b10 = Downscale
22:18	RW	0x00	sw_rgb_b_padd Amount of ones that will be padded in front of the B-component
17:0	RW	0x00000	sw_scale_hratio Scaling ratio for height (outputh-1/inputh-1)

VDPU_SWREG81
Address: Operational Base + offset (0x0144)
Scaling ratio register 2

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	sw_wscale_invra Inverse scaling ratio for width, or ch (inputw-1 / outputw-1)
15:0	RW	0x0000	sw_hscale_invra Inverse scaling ratio for height or cv (inputh-1 / outputh-1)

VDPU_SWREG82
Address: Operational Base + offset (0x0148)
Rmask register

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	sw_r_mask Bit mask for R component (and alpha channel)

VDPU_SWREG83
Address: Operational Base + offset (0x014c)
Gmask register

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	sw_g_mask Bit mask for G component (and alpha channel)

VDPU_SWREG84
Address: Operational Base + offset (0x0150)
Bmask register

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	sw_b_mask Bit mask for B component (and alpha channel)

VDPU_SWREG85
Address: Operational Base + offset (0x0154)
Post-processor control register

Bit	Attr	Reset Value	Description
31:29	RW	0x0	sw_pp_in_format PP input picture data format 3'd0 = YUYV 4:2:2 interleaved (supported only in external mode) 3'd1 = YCbCr 4:2:0 Semi-planar in linear raster-scan format 3'd2 = YCbCr 4:2:0 planar (supported only in external mode) 3'd3 = YCbCr 4:0:0 (supported only in pipelined mode) 3'd4 = YCbCr 4:2:2 Semi-planar (supported only in pipelined mode) 3'd5 = YCbCr 4:2:0 Semi-planar in tiled format (supported only in external mode (8170 decoder only) 3'd6 = YCbCr 4:4:0 Semi-planar (supported only in pipelined mode, possible for jpeg only) 3'd7 = Escape pp input data format. Defined in swreg86
28:26	RW	0x0	sw_pp_out_format PP output picture data format: 3'd0 = RGB 3'd1 = YCbCr 4:2:0 planar (Not supported) 3'd2 = YCbCr 4:2:2 planar (Not supported) 3'd3 = YUYV 4:2:2 interleaved 3'd4 = YCbCr 4:4:4 planar (Not supported) 3'd5 = YCh 4:2:0 chrominance interleaved 3'd6 = YCh 4:2:2 (Not supported) 3'd7 = YCh 4:4:4 (Not supported)

Bit	Attr	Reset Value	Description
25:15	RW	0x000	sw_pp_out_height Scaled picture height in pixels (Must be dividable by 2 or by any if Pixel Accurate PP output configuration is enabled) Max scaled picture height is 1920 pixels or maximum three times the input source height minus 8 pixels
14:4	RW	0x000	sw_pp_out_width Scaled picture width in pixels. Must be dividable by 8 or by any if Pixel Accurate PP output configuration is enabled. Max scaled picture width is 1920 pixels or maximum three times the input source width minus 8 pixels
3	RW	0x0	sw_pp_out_tiled_e Tiled mode enable for PP output. Can be used only for YCbYCr 422 output format. Can be used only if correpongind configuration supports this feature. Tile size is 4x4 pixels.
2	RW	0x0	sw_pp_out_swap16_e PP output swap 16 swaps 16 bit halfs inside of 32 bit word. Can be used for 16 bit RGB to change pixel orders but is valid also for any output format NOTE: requires that configuration of SW_PPD_OEN_VERSION=1
1	RW	0x0	sw_pp_crop8_r_e PP input picture width is not 16 pixels multiple. Only 8 pixels of the most right MB of the unrotated input picture is used for PP input.
0	RW	0x0	sw_pp_crop8_d_e PP input picture height is not 16 pixels multiple. Only 8 pixel rows of the most down MB of the unrotated input picture is used for PP input.

VDPU_SWREG86
Address: Operational Base + offset (0x0158)
Mask 1 start coordinate register

Bit	Attr	Reset Value	Description
31:29	RW	0x0	sw_pp_in_format_es Escape PP in format. Used if sw_pp_in_format is defined to 7: 0 2'd0 = YCbCr 4:4:4 2'd1 = YCbCr 4:1:1
28	RO	0x0	reserved
27:23	RW	0x00	sw_rangemap_coef_c Range map value for chrominance component (RANGE_MAPUV+9 in VC-1 standard)

Bit	Attr	Reset Value	Description
22	RW	0x0	sw_mask1_ablend_e Mask 1 alpha blending enable. Instead of masking the output picture the alpha blending is performed. Alpha blending source can be found from alpha blend 1 base address. Alpha blending can be enabled only for RGB/ YUYV 422 data.
21:11	RW	0x000	sw_mask1_starty Vertical start pixel for mask area 1. Defines the y coordinate. Coordinate 0,0 means the up-left corner in PP output luminance picture. See Table 47 for restrictions
10:0	RW	0x000	sw_mask1_startix Horizontal start pixel for mask area 1. Defines the x coordinate. Coordinate 0,0 means the up-left corner in PP output luminance picture. See Table 47 for restrictions

VDPU_SWREG87
Address: Operational Base + offset (0x015c)
Mask 2 start coordinate register

Bit	Attr	Reset Value	Description
31:23	RO	0x0	reserved
22	RW	0x0	sw_mask2_ablend_e Mask 2 alpha blending enable. Instead of masking the output picture the alpha blending is performed. Alpha blending source can be found from alpha blend 2 base address. Alpha blending can be enabled only for RGB/YUYV 422 data.
21:11	RW	0x000	sw_mask_starty Verical start pixel for mask area 2. Defines the y coordinate. Coordinate 0,0 means the up-left corner in PP ouput Y picture. See Table 47 for restrictions
10:0	RW	0x000	sw_mask2_startx Horizontal start pixel for mask area 2. Defines the x coordinate. Coordinate 0,0 means the up-left corner in PP output Y picture. See Table 47 for restrictions

VDPU_SWREG88
Address: Operational Base + offset (0x0160)
Mask 1 size and PP original width register

Bit	Attr	Reset Value	Description
31:23	RW	0x000	sw_ext_orig_width PP input picture original width in macro blocks.
22	RW	0x0	sw_mask1_e Mask 1 enable. If mask 1 is used this bit is high
21:11	RW	0x000	sw_mask1_endy Mask 1 end coordinate y in pixels (inside of PPD output picture). Range must be between [Mask1StartCoordinateY, ScaledHeight].

Bit	Attr	Reset Value	Description
10:0	RW	0x000	sw_mask1_endx Mask 1 end coordinate x in pixels (inside of PPD output picture). Range must be between [Mask1StartCoordinateX, ScaledWidth]

VDPU_SWREG89
Address: Operational Base + offset (0x0164)
Mask 2 size register

Bit	Attr	Reset Value	Description
31:23	RO	0x0	reserved
22	RW	0x0	sw_mask2_e Mask 2 enable. If mask 1 is used this bit is high
21:11	RW	0x000	sw_mask2_endy Mask 2 end coordinate y in pixels (inside of PP output picture). Range must be between [Mask2StartCoordinateY, ScaledHeight].
10:0	RW	0x000	sw_mask2_endx Mask 2 end coordinate x in pixels (inside of PP output picture). Range must be between [Mask2StartCoordinateX, ScaledWidth].

VDPU_SWREG90
Address: Operational Base + offset (0x0168)
PiP register 0

Bit	Attr	Reset Value	Description
31:30	RO	0x0	reserved
29	RW	0x0	sw_right_cross_e Right side overcross enable. 0 = No right side overcross, 1 = right side overcross
28	RW	0x0	sw_left_cross_e Left side overcross enable. 0 = No left side overcross, 1 = left side overcross
27	RW	0x0	sw_up_cross_e Upward overcross enable. 0 = No upward overcross, 1 = upward overcross
26	RW	0x0	sw_down_cross_e Downward overcross enable. 0 = No downward overcross, 1 = downward overcross
25:15	RW	0x000	sw_up_cross Amount of upward overcross (vertical pixels outside of display from the upper side). Range must be between [0, ScaledHeight].
14:11	RO	0x0	reserved

Bit	Attr	Reset Value	Description
10:0	RW	0x000	sw_down_cross Amount of downward overcross (vertical pixels outside of displayfrom the down side). Range must be between [0,ScaledHeight].

VDPU_SWREG91

Address: Operational Base + offset (0x016c)

PiP register 1 and dithering control

Bit	Attr	Reset Value	Description
31:30	RW	0x0	sw_dither_select_r Dithering control for R channel: 2'b00 = dithering disabled 2'b01 = use four-bit dither matrix 2'b10 = use five-bit dither matrix 2'b11 = use six-bit dither matrix
29:28	RW	0x0	sw_dither_select_g Dithering control for G channel: 2'b00 = dithering disabled 2'b01 = use four-bit dither matrix 2'b10 = use five-bit dither matrix 2'b11 = use six-bit dither matrix
27:26	RW	0x0	sw_dither_select_b Dithering control for B channel: 2'b00 = dithering disabled 2'b01 = use four-bit dither matrix 2'b10 = use five-bit dither matrix 2'b11 = use six-bit dither matrix
25:24	RO	0x0	reserved
23:22	RW	0x0	sw_pp_tiled_mode Input data is in tiled mode (at the moment valid only for YCbCr 420 data, pipeline or external mode): 2'd0 = Tiled mode not used 2'd1 = Tiled mode enabled for 8x4 sized tiles 2'd2, 2'd3 = reserved
21:11	RW	0x000	sw_right_cross Amount of right side overcross (Horizontal pixels outside o display from the right side). Range must be between [0, ScaledWidth].
10:0	RW	0x000	sw_left_cross Amount of left side overcross (Horizontal pixels outside of display from the left side). Range must be between [0, ScaledWidth].

VDPU_SWREG92

Address: Operational Base + offset (0x0170)
Display width and PP input size extension register

Bit	Attr	Reset Value	Description
31:29	RW	0x0	sw_pp_in_h_ext Extended PP input height. Used with JPEG
28:26	RW	0x0	sw_pp_in_w_ext Extended PP input width. Used with JPEG
25:23	RW	0x0	sw_crop_starty_ext Extended PP input crop start coordinate x. Used with JPEG
22:20	RW	0x0	sw_crop_start_ext Extended PP input crop start coordinate y. Used with JPEG
19:12	RO	0x0	reserved
11:0	RW	0x000	sw_display_width Width of the display in pixels. Max HDTV (1920)

VDPU_SWREG93
Address: Operational Base + offset (0x0174)
Display width and PP input size extension register

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	sw_abledn1_base Base address for alpha blending input 1 (if mask1 is used in alpha blending mode). Format of data is 24 bit RGB/ YCbCr and endian/swap -mode is as in PP input. Amount of data is informed with mask 1 size or with ablend1_scanline if ablend cropping is supported in configuration.

VDPU_SWREG94
Address: Operational Base + offset (0x0178)
Base address for alpha blend 2 gui component

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	sw_ablend2_base Base address for alpha blending input 2 (if mask2 is used in alpha blending mode). Format of data is 24 bit RGB/ YCbCr and endian/swap -mode is as in PP input. Amount of data is informed with mask 2 size or with ablend2_scanline if ablend cropping is supported in configuration.

VDPU_SWREG95
Address: Operational Base + offset (0x017c)
Base address for alpha blend 2 gui component

Bit	Attr	Reset Value	Description
31:26	RO	0x0	reserved
25:13	RW	0x0000	sw_ablend2_scan Scanline width in pixels for Ablend 2. Usage enabled if corresponding configuration bit is enabled

Bit	Attr	Reset Value	Description
12:0	RW	0x0000	sw_ablend1_scan Scanline width in pixels for Ablend 1. Usage enabled if corresponding configuration bit is enabled

VDPU_SWREG98
Address: Operational Base + offset (0x0188)
PP outupt width/height extension

Bit	Attr	Reset Value	Description
31:2	RO	0x0	reserved
1	RW	0x0	sw_pp_out_h_ext sw_pp_out_h_ext PP output heightextension
0	RW	0x0	sw_pp_out_w_ext sw_pp_out_w_ext PP output widthextension

VDPU_SWREG99
Address: Operational Base + offset (0x018c)
PP fuse register (read only)

Bit	Attr	Reset Value	Description
31	RO	0x1	fuse_pp_pp 1 = PP enabled
30	RO	0x1	fuse_pp_deint 1 = Deinterlacing enabled
29	RO	0x1	fuse_pp_ablend 1 = Alpha Blending enabled
28:16	RO	0x0	reserved
15	RO	0x1	fuse_pp_maxw_1920 1 = Max PP output width up to 1920 pixels enabled. Priority coded with priority 1
14	RO	0x1	fuse_pp_maxw_1280 1 = Max PP output width up to 1280 pixels enabled. Priority coded with priority 2
13	RO	0x1	fuse_pp_maxw_720 1 = Max PP output width up to 720 pixels enabled. Priority coded with priority 3
12	RO	0x1	fuse_pp_maxw_352 1 = Max PP output width up to 352 pixels enabled. Priority coded with priority 4
11:0	RO	0x0	reserved

VDPU_SWREG100
Address: Operational Base + offset (0x0190)
Synthesis configuration register post-processor (read only)

Bit	Attr	Reset Value	Description
31	RO	0x1	SW_ABLEND_CROP_E Alpha blending support for input cropping: '0': Not supported. External memory must include the exact image of the area being alpha blended '1' Supported. External memory can include a picture from blended area can be cropped. Requires usage of swreg95
30	RO	0x1	SW_PPD_PIXAC_E Pixel Accurate PP output mode exists: '0'= PIP, Scaling and masks can be adjusted by steps of 8 pixels (width) or 2 pixels (height) '1' = PIP, Scaling and masks can be adjusted by steps of 1 pixel for RGB and 2 pixels for subsampled chroma formats (by using bus specific write strobe functionality)
29	RO	0x1	SW_PPD_TILED_EXIST PP output YCbYCr 422 tiled support (4x4 pixel tiles) '0'=Not supported '1'=Supported
28	RO	0x1	SW_PPD_DITH_EXIST Dithering exists: '0' = no '1' = yes
27:26	RO	0x3	SW_PPD_SCALE_LEVEL Scaling support: 2'b00 = No scaling 2'b01 = Scaling with lo perfomance architecture 2'b10 = Scaling with high performance architecture 2'b11 = Scaling with high performance architecture + fast
25	RO	0x1	SW_PPD_DEINT_EXIST De-interlacing exits: '0' = no '1' = yes
24	RO	0x1	SW_PPD_BLEND_EXIST Alpha blending exists: '0' = no '1' = yes
23	RO	0x1	SW_PPD_IBUFF_LEVEL PP input buffering level: '0' = 1 MB input buffering is used '1' = 4 MB input buffering is used
22:19	RO	0x0	reserved
18	RO	0x1	SW_PPD_OEN_VERSION PP output endian version: '0' = Endian mode supported for other than RGB '1' = Endian mode supported for any output format

Bit	Attr	Reset Value	Description
17	RO	0x1	SW_PPD_OBUFF_LEVEL PP output buffering level: '0' = 1 unit output buffering is used '1' = 4 unit output buffering is used
16	RO	0x1	SW_PPD_PP_EXIST PPD exists: '0'=no '1'=yes
15:14	RO	0x1	SW_PPD_IN_TILED_L PPD input tiled mode support level 2'b0 = not supported 2'b1 = 8x4 tile size supported
13:11	RO	0x0	reserved
10:0	RO	0x780	SW_PPD_MAX_OWIDTH Max supported PP output width in pixels

VDPU_SWREG101
Address: Operational Base + offset (0x0194)
soft reset signals

Bit	Attr	Reset Value	Description
31:1	RO	0x0	reserved
0	RW	0x0	sw_soft_reset softreset pulse signal write to 1'b1, valid; write to 1'b0, invalid;

2.4.11 VPU MMU Register Summary

Name	Offset	Size	Reset Value	Description
VCODEC_MMU_DTE_ADDR	0x0000	W	0x00000000	MMU current page Table address
VCODEC_MMU_STATUS	0x0004	W	0x00000018	MMU status register
VCODEC_MMU_COMMAND	0x0008	W	0x00000000	MMU command register
VCODEC_MMU_PAGE_FAULT_ADDR	0x000c	W	0x00000000	MMU logical address of last page fault
VCODEC_MMU_ZAP_ONE_LINE	0x0010	W	0x00000000	MMU Zap cache line register
VCODEC_MMU_INT_RAWSTAT	0x0014	W	0x00000000	MMU raw interrupt status register
VCODEC_MMU_INT_CLEAR	0x0018	W	0x00000000	MMU raw interrupt status register
VCODEC_MMU_INT_MASK	0x001c	W	0x00000000	MMU raw interrupt status register
VCODEC_MMU_INT_STATUS	0x0020	W	0x00000000	MMU raw interrupt status register

Name	Offset	Size	Reset Value	Description
VCODEC_MMU_AUTO_GATING	0x0024	W	0x00000001	mmu auto gating

*Notes:Size:**B**- Byte (8 bits) access, **HW**- Half WORD (16 bits) access, **W**-WORD (32 bits) access*

2.4.12 VPU MMU Detail Register Description

VCODEC_MMU_DTE_ADDR

Address: Operational Base + offset (0x0000)

MMU current page Table address

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	MMU_DTE_ADDR MMU_DTE_ADDR MMU current page Table address

VCODEC_MMU_STATUS

Address: Operational Base + offset (0x0004)

MMU status register

Bit	Attr	Reset Value	Description
31:11	RO	0x0	reserved
10:6	RO	0x00	PAGE_FAULT_BUS_ID page fault bus id Index of master reponsible for last page fault
5	RO	0x0	PAGE_FAULT_IS_WRITE page fault acess The direction of access for last page fault: 0 = Read 1 = Write
4	RO	0x1	REPLAY_BUFFER_EMPTY replay buffer empty status 1'b1:The MMU replay buffer is empty
3	RO	0x1	MMU_IDLE mmu idle status The MMU is idle when accesses are being translated and there are no unfinished translated accesses. 1'b1: MMU is idle
2	RO	0x0	STAIL_ACTIVE stall active status MMU stall mode currently enabled. The mode is enabled by command 1'b1: MMU is in stall active status
1	RO	0x0	PAGE_FAULT_ACTIVE page fault active status MMU page fault mode currently enabled . The mode is enabled by command. 1'b1: page fault is active

Bit	Attr	Reset Value	Description
0	RO	0x0	PAGING_ENABLED Paging enabled status 1'b0: paging is disabled 1'b1: Paging is enabled

VCODEC_MMU_COMMAND
Address: Operational Base + offset (0x0008)
MMU command register

Bit	Attr	Reset Value	Description
31:3	RO	0x0	reserved
2:0	WO	0x0	MMU_CMD mmu cmd MMU_CMD. This can be: 0: MMU_ENABLE_PAGING 1: MMU_DISABLE_PAGING 2: MMU_ENABLE_STALL 3: MMU_DISABLE_STALL 4: MMU_ZAP_CACHE 5: MMU_PAGE_FAULT_DONE 6: MMU_FORCE_RESET

VCODEC_MMU_PAGE_FAULT_ADDR
Address: Operational Base + offset (0x000c)
MMU logical address of last page fault

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	PAGE_FAULT_ADDR page fault addr address of last page fault

VCODEC_MMU_ZAP_ONE_LINE
Address: Operational Base + offset (0x0010)
MMU Zap cache line register

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	MMU_ZAP_ONE_LINE zap one line address to be invalidated from the page table cache

VCODEC_MMU_INT_RAWSTAT
Address: Operational Base + offset (0x0014)
MMU raw interrupt status register

Bit	Attr	Reset Value	Description
31:2	RO	0x0	reserved
1	RW	0x0	READ_BUS_ERROR read bus error read bus error status

Bit	Attr	Reset Value	Description
0	RW	0x0	PAGE_FAULT page fault page fault status

VCODEC_MMU_INT_CLEAR
Address: Operational Base + offset (0x0018)
MMU raw interrupt status register

Bit	Attr	Reset Value	Description
31:2	RO	0x0	reserved
1	WO	0x0	READ_BUS_ERROR read bus error write 1 to clear read bus error
0	WO	0x0	PAGE_FAULT page fault clear write 1 to page fault clear

VCODEC_MMU_INT_MASK
Address: Operational Base + offset (0x001c)
MMU raw interrupt status register

Bit	Attr	Reset Value	Description
31:2	RO	0x0	reserved
1	RW	0x0	READ_BUS_ERROR read bus error enable the read bus interrupt souce when this bit is set to 1'b1
0	RW	0x0	PAGE_FAULT page fault mask enable the page fault interrupt souce when this bit is set to 1'b1

VCODEC_MMU_INT_STATUS
Address: Operational Base + offset (0x0020)
MMU raw interrupt status register

Bit	Attr	Reset Value	Description
31:2	RO	0x0	reserved
1	RO	0x0	READ_BUS_ERROR read bus error status 1'b1:read bus error status
0	RO	0x0	PAGE_FAULT page fault status 1'b1:page fault

VCODEC_MMU_AUTO_GATING
Address: Operational Base + offset (0x0024)
mmu auto gating

Bit	Attr	Reset Value	Description
31:1	RO	0x0	reserved

Bit	Attr	Reset Value	Description
0	RW	0x1	mmu_auto_clkgating mmu_auto_clkgating when it is 1'b1, the mmu will auto gating it self

2.4.13 VDPU Pref_cacheRegister Summary

Name	Offset	Size	Reset Value	Description
pref_cache_VERSION	0x0000	W	0xcac20101	VERSION register
pref_cache_SIZE	0x0004	W	0x06110206	L2 cache SIZE
pref_cache_STATUS	0x0008	W	0x00000000	Status register
pref_cache_COMMAND	0x0010	W	0x00000000	Command setting register
pref_cache_CLEAR_PAGE	0x0014	W	0x00000000	clear page register
pref_cache_MAX_READS	0x0018	W	0x0000001c	maximum read register
pref_cache_PERFCNT_SRC0	0x0020	W	0x00000000	performance counter 0 source register
pref_cache_PERFCNT_VAL0	0x0024	W	0x00000000	performance counter 0 value register
pref_cache_PERFCNT_SRC1	0x0028	W	0x00000000	performance counter 0 source register
pref_cache_PERFCNT_VAL1	0x002c	W	0x00000000	performance counter 1 value register

*Notes:Size:**B**- Byte (8 bits) access, **HW**- Half WORD (16 bits) access, **W**-WORD (32 bits) access*

2.4.14 VDPU_Pref_cache Detail Register Description

pref_cache_VERSION

Address: Operational Base + offset (0x0000)

VERSION register

Bit	Attr	Reset Value	Description
31:16	RO	0xcac2	PRODUCT_ID Field0000 Abstract Field0000 Description
15:8	RO	0x01	VERSION_MAJOR Field0000 Abstract Field0000 Description
7:0	RO	0x01	VERSION_MINOR Field0000 Abstract Field0000 Description

pref_cache_SIZE

Address: Operational Base + offset (0x0004)

L2 cache SIZE

Bit	Attr	Reset Value	Description
31:24	RO	0x06	External_bus_width Field0000 Abstract Log2 external bus width in bits

Bit	Attr	Reset Value	Description
23:16	RO	0x11	CACHE_SIZE Field0000 Abstract Log2 cache size in bytes
15:8	RO	0x02	ASSOCIATIVITY Field0000 Abstract Log2 associativity
7:0	RO	0x06	LINE_SIZE Field0000 Abstract Log2 line size in bytes

pref_cache_STATUS
Address: Operational Base + offset (0x0008)
Status register

Bit	Attr	Reset Value	Description
31:2	RO	0x0	reserved
1	RO	0x0	DATA_BUSY Field0000 Abstract set when the cache is busy handling data
0	RO	0x0	CMD_BUSY Field0000 Abstract set when the cache is busy handling commands

pref_cache_COMMAND
Address: Operational Base + offset (0x0010)
Command setting register

Bit	Attr	Reset Value	Description
31:6	RO	0x0	reserved
5:4	RW	0x0	sw_addrb_sel Field0000 Abstract 2'b00: to sel b[14:6] 2'b01: to sel b[15:9], b[7:6] 2'b10: to sel b[16:10], b[7:6] 2'b11: to sel b[17:11], b[7:6]
3	RO	0x0	reserved
2:0	WO	0x0	COMMAND Field0000 Abstract The possible command is 1 = Clear entire cache

pref_cache_CLEAR_PAGE
Address: Operational Base + offset (0x0014)
clear page register

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	CLEAR_PAGE Field0000 Abstract writing an address, invlidates all lines in that page from the cache

pref_cache_MAX_READS
Address: Operational Base + offset (0x0018)
maximum read register

Bit	Attr	Reset Value	Description
31:5	RO	0x0	reserved
4:0	RW	0x1c	MAX_READS Field0000 Abstract Limit the number of outstanding read transactions to this amount

pref_cache_PERFCNT_SRC0
Address: Operational Base + offset (0x0020)
performance counter 0 source register

Bit	Attr	Reset Value	Description
31:4	RO	0x0	reserved
3:0	RW	0x0	PERFCNT_SRC0 Field0000 Abstract This register holds all the possible source values for Performance Counter 0 4'd0: total clock cycles 4'd1: active clock cycles 4'd2: read transactions, master 4'd3: word reads, master 4'd4: read transactions, slave 4'd5: word reads, slave 4'd6: read hit, slave 4'd7: read misses, slave 4'd8: read invalidates, slave 4'd9: cacheable read transactions, slave 4'd10: bad hit nmber, slave

pref_cache_PERFCNT_VAL0
Address: Operational Base + offset (0x0024)
performance counter 0 value register

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	PERFCNT_VAL0 Field0000 Abstract Performance counter 0 value

pref_cache_PERFCNT_SRC1
Address: Operational Base + offset (0x0028)
performance counter 0 source register

Bit	Attr	Reset Value	Description
31:4	RO	0x0	reserved
3:0	RW	0x0	PERFCNT_SRC1 Field0000 Abstract This register holds all the possible source values for Performance Counter 1 4'd0: total clock cycles 4'd1: active clock cycles 4'd2: read transactions, master 4'd3: word reads, master 4'd4: read transactions, slave 4'd5: word reads, slave 4'd6: read hit, slave 4'd7: read misses, slave 4'd8: read invalidates, slave 4'd9: cacheable read transactions, slave 4'd10: bad hit nmber, slave

pref_cache_PERFCNT_VAL1
Address: Operational Base + offset (0x002c)
performance counter 1 value register

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	PERFCNT_VAL1 Field0000 Abstract Performance counter 1 value

2.5 Application Notes

2.5.1 HEVC Configuration flow

1. Write the hevc_vpu_sel bit in the GRF_SOC_CONx register with "1" to choose HEVC to work.
2. Prepare the data in the DDR.
3. Set the HEVC general system configuration in HEVC.swreg2, such as working mode, in/out endian.
4. Set the picture parameters with HEVC.swreg3.
5. Set the input and output data base address and HEVC reference configuration with HEVC.swreg4~HEVC.swreg43.
6. If CABAC error detection is desired, set the HEVC.swreg44 to enable the corresponding error detection.
7. Set the interrupt configuration and start the HEVC with HEVC.swreg1.
8. Wait for the frame interrupt, and then get the processed results in the target DDR
9. Clear all the interrupts, and repeat Process2~Process8 to start a new frame decoding if the decoding is not finished yet.

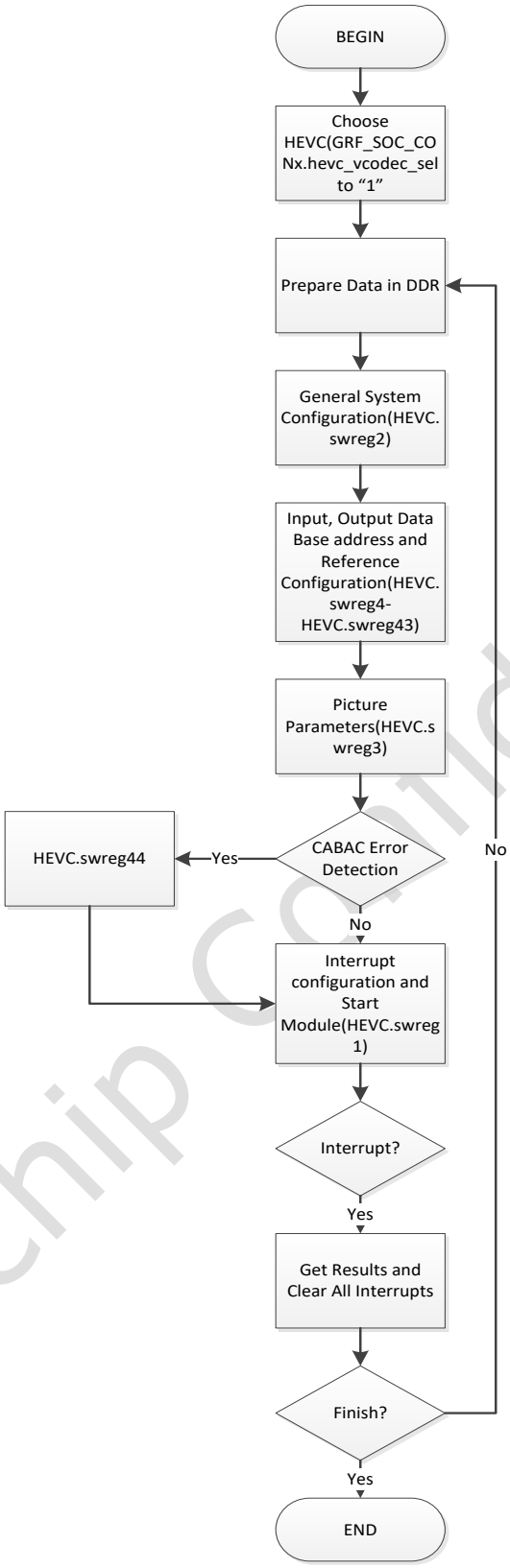


Fig. 2-6 HEVC Common Configuration Flow

2.5.2 VPU Configuration flow

1.Because HEVC and VPU share the same memory and ahb bus and some of axi bus, we should set the hevc_vpu_sel bit in GRF_SOC_CONx to 1’b0 to select VPU to work.

2.Prepare the decoder data in the DDR memory, And in decoder other than JPEG decoder, the input stream buffer should at least contain a slice or a frame data, otherwise the

decoder will produce an interrupt and show error and then reset itself.

3.Config all the registers will be used. and please notice that which be list as follows:
In encoder----- We can configure the registers to control the input picture data format (such as endian and swap), but some input data format are fixed, such as cabac_table data. And the register VEPU_SWREG64~95 are JPEG quantization registers. They are write only registers. When you want to write these registers, you should first set VEPU_SWREG14[0] to 1'b0 and VEPU_SWREG14[2:1] to 2'b10(select JPEG mode).
In decoder---- The decoder can support ref buffer mode or cacheable mode, but they can't be both enabled. We can config the swreg57[6],swreg57[7] to enable cache and config the swreg51 to control the ref buffer.

4.You should config VDPU_SWREG1[0] as 1'b1 to enable video decoder. And config VDPU_SWREG60[0] as 1'b1 to enable pp. If pp performed in pipeline with decoder, you should config VDPU_SWREG60[1] as 1'b1 and then config VDPU_SWREG1[0] as 1'b1 to enable decoder and pp. VEPU_SWREG14[0] set to 1'b1 to enable encoder.

5.Wait for the frame interrupt,and then check if the frame decoder ready interrupt is right or not, after that,you can get the processed results in the target DDR

6.Clear all the interrupts,repeat step 2~5 to start a new frame decoder or encoder.

Chapter 3 RGA

3.1 Overview

RGA is a separate 2D raster graphic acceleration unit. It accelerates 2D graphics operations, such as point/line drawing, image scaling, rotation, BitBLT, alpha blending and image blur/sharpness.

3.1.1 Features

- **Data format**
  - Input data: ARGB/RGB888/RGB565/YUV420/YUV422
  - Output data: ARGB/RGB888/RGB565 (YUV420/YUV422 for blur/sharpness);
  - Pixel Format conversion, BT.601/BT.709
  - Dither operation
  - Max resolution: 8192x8192 source image, 2048x2048 frame buffer
- **Scaling**
  - Down-scaling and up-scaling
  - Three sampling modes: Nearest sampling (Stretched BitBLT), Bi-linear filter or Bi-cubic filter
  - Arbitrary non-integer scaling ratio, from 1/2 to 8
  - Average filter pre-scaling (2's Down-scaling bypass path, not available with other 2D operation)
- **Rotation**
  - Arbitrary rotation, minimum 1 degree step
  - No per-pixel alpha in arbitrary rotation (without 90, 180, 270)
  - x-mirror, y-mirror
- **BitBLT**
  - Block transfer
  - Color palette (with transparency mode)/Color fill
  - Transparency mode (color keying/stencil test, specified value/range)
- **Alpha Blending**
  - Per-pixel/user-specified alpha blending (Porter-duff alpha support)
  - Fading
  - Anti-aliasing (for rotation)
- **Raster operation**
  - ROP2/ROP3/ROP4
  - No ROP in arbitrary rotation (except 90/180/270 degree)
- **YUV_Output**
  - Output data: YUV422SP/YUV420SP, not used in Pre_scaling Line/point drawing and Blur/sharp, Alpha and ROP.
  - ColorPalette alpha 1/8bit mode
- **RW_Align**
  - AXI read/write can be DDR-Align in Line scan mode, high performance.

3.2 Block Diagram

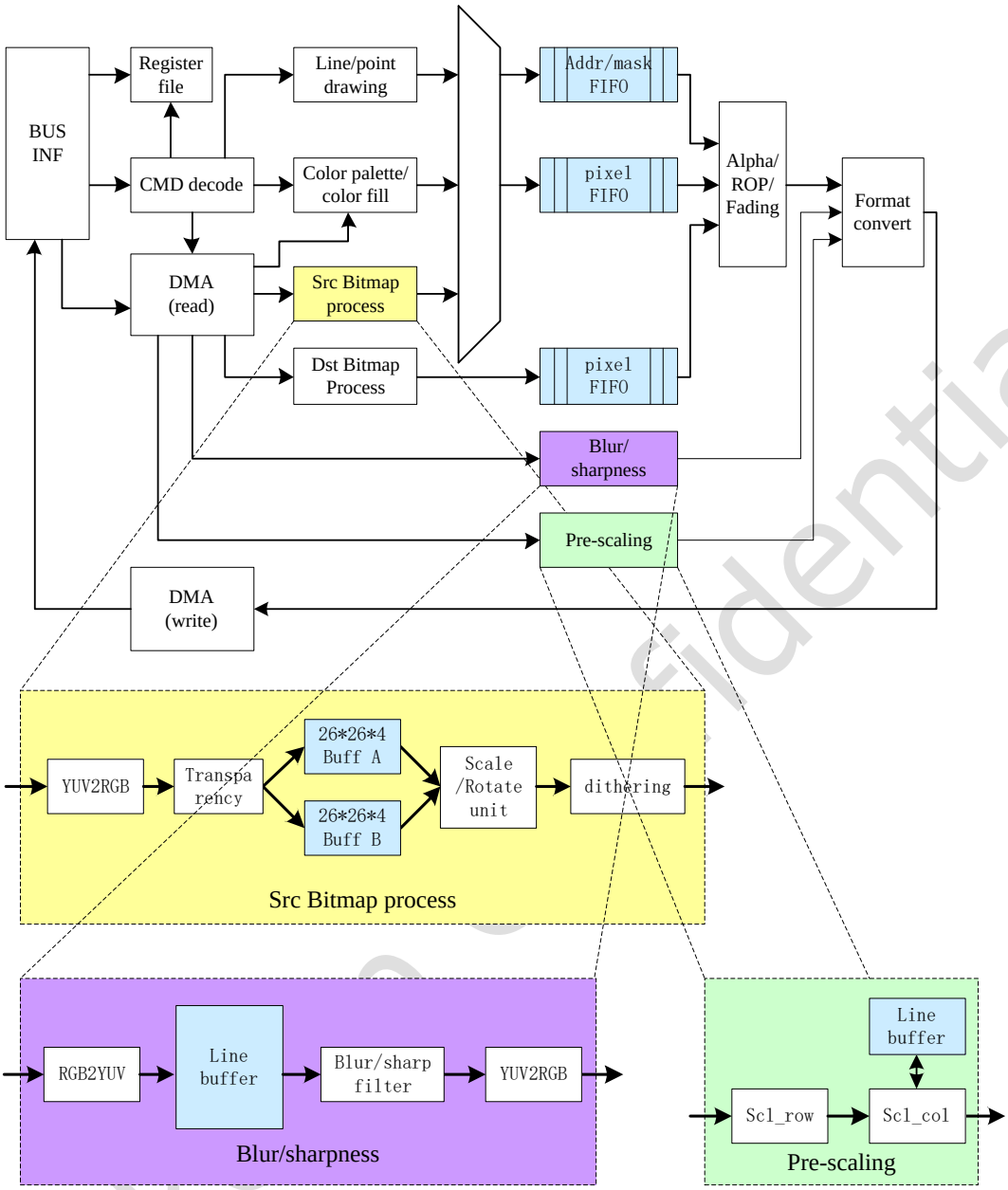


Fig. 3-1 RGA Block Diagram

3.3 Function Description

3.3.1 Data Format

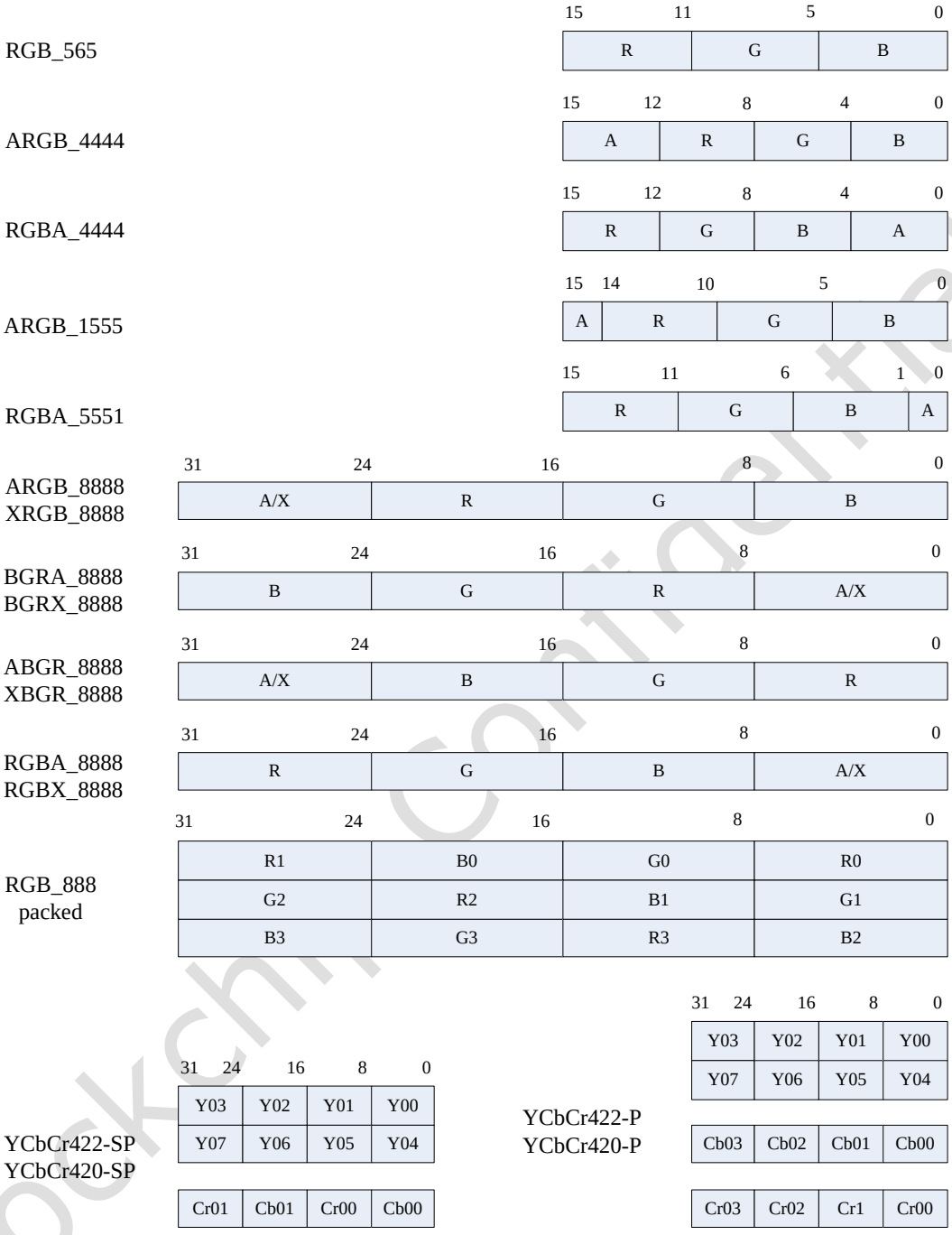


Fig. 3-2 RGA Input Data Format

All input datas (defined by SRC_IN_FMT/DST_IN_FMT) are converted to ABGR8888. The results are converted to the output data format (defined by DST_OUT_FMT).

3.3.2 Dithering

There could have dithering operation for source image when the source image format is not RGB565 and the destination format is RGB565.

The down-dithering is done using Dither Matrix.

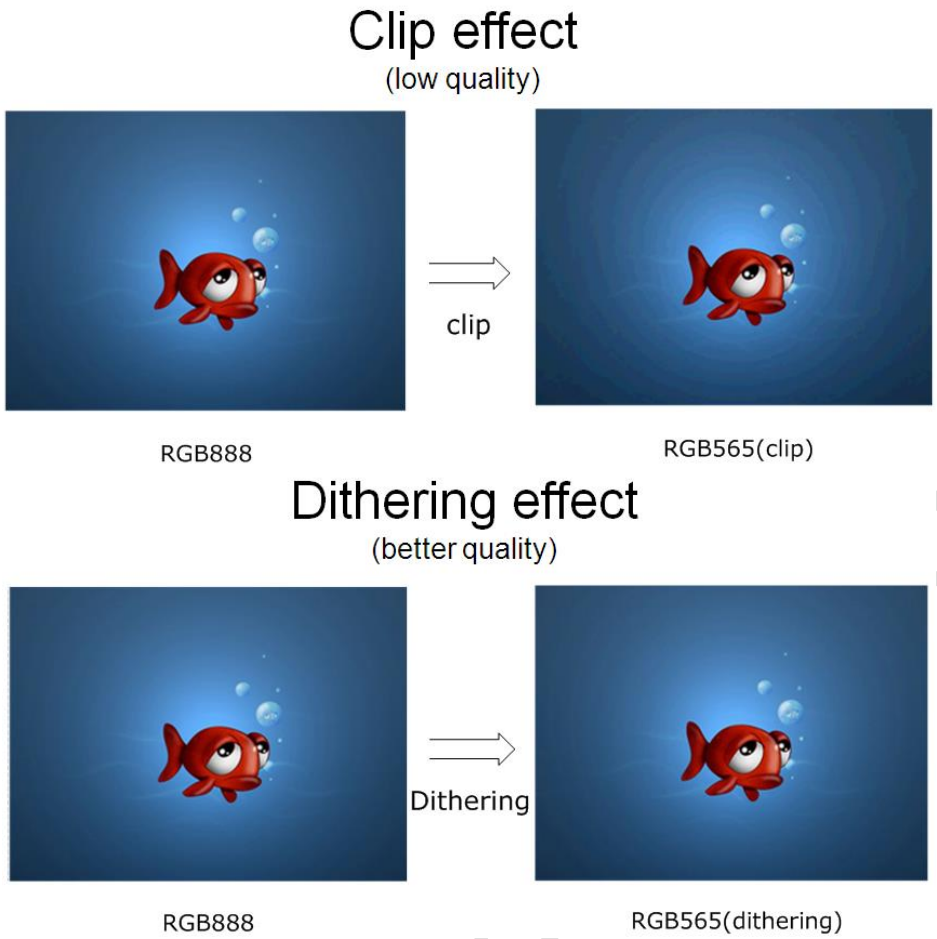


Fig. 3-3 RGA Dither effect

3.3.3 Scaling

The scaling operation is the image resizing processing of source image. Scaling is done base on ARGB8888 format.

There are three sampling modes: Nearest sampling (Stretched BitBLT), Bi-linear filter or Bi-cubic filter.

3.3.4 Rotation

Arbitrary rotation and x-mirror, y-mirror operation is supported in RGA. The rotation operation is combined with scaling operation.

Alpha is available only if there is no rotation or 90-degree/180-degree /270-degree rotation or x-mirror/y-mirror. Anti-aliasing is done by the alpha blending of the boundary pixels.

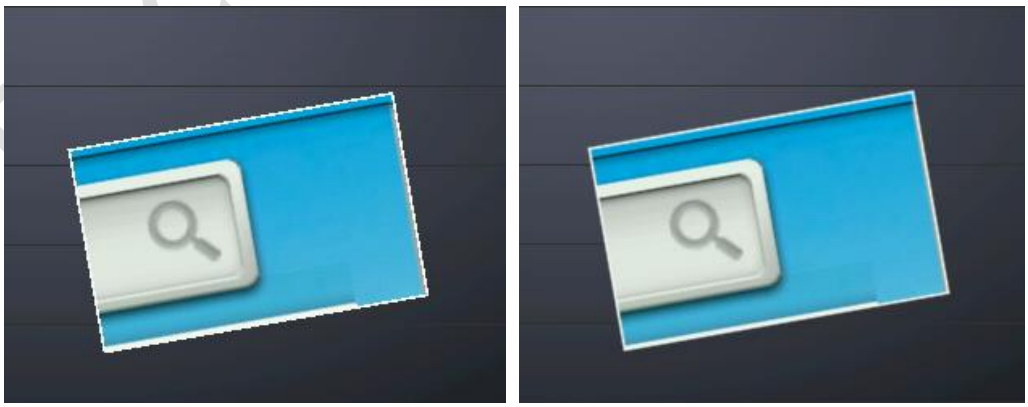


Fig. 3-4 RGA Rotation AA effect

3.3.5 Bitmap Block Transfer

BitBlt is a computer graphics operation in which several bitmaps are combined into one using a raster operator. The bitmap rectangular block is transfered from source frame buffer to destination frame buffer. There are three bitmap block transfer type: bitmap block transfer (RGB/YCbCr), color expansion, and solid fill.
RGA also supports transparency mode in BitBLT. There are two transparency modes (stencil test): normal mode and inverted mode.
There are 4 enable control bits for ARGB color channel for stencil test, which can be set independently.

- Transparency mode
  - Normal Stencil test (Color keying), Pixels with the same color or in the range of user-specified colors are discarded.
  - Inverted stencil test, Pixels with the different color or out the range of user-specified colors are discarded.
- Color palette
  - 1bpp/2bpp/4bpp/8bpp palette data formats are support in RGA source layer. 1bpp color expansion can be BG color and FG color, transparency and FG color according the alpha enable bit. There is a 256x25bit LUT in RGA for 2bpp/4bpp/8bpp color palette. The following is the table of 8bpp with alpha enable bit in the MSB.

Table 3-1 RGA 8bpp color palette LUT

INDEX\ Bit Pos.	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
00H	AEN	R7	R6	R5	R4	R3	R2	R1	R0	G7	G6	G5	G4	G3	G2	G1	G0	B7	B6	B5	B4	B3	B2	B1	B0
01H	AEN	R7	R6	R5	R4	R3	R2	R1	R0	G7	G6	G5	G4	G3	G2	G1	G0	B7	B6	B5	B4	B3	B2	B1	B0
.....	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…
FFH	AEN	R7	R6	R5	R4	R3	R2	R1	R0	G7	G6	G5	G4	G3	G2	G1	G0	B7	B6	B5	B4	B3	B2	B1	B0

- Color fill
  - Two modes of color fill can be done by RGA: solid fill and gradient fill.

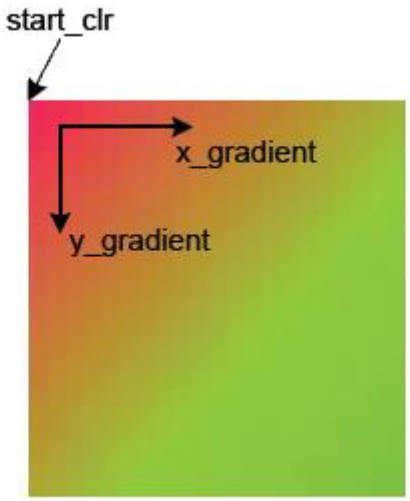


Fig. 3-5 RGA Gradient Fill

Gradient fill using following equations for ARGB calculation of every pixel in different coordinary.

A_cur = (A_start + x*x_A_gradient) +y*y_A_gradient;
R_cur = (R_start + x*x_R_gradient) +y*y_R_gradient;
G_cur = (G_start + x*x_G_gradient) +y*y_G_gradient;
B_cur = (B_start + x*x_B_gradient) +y*y_B_gradient;

A_start, R_start, G_start, B_start is the ARGB value of start point. There are four pairs of values for horizontal and vertical gradient. Saturation operation could be enabled or disabled if the color overflows 255 or underflows 0.

3.3.6 Alpha Blending

Alpha blending is devided to two stages. The first stage is mix alpha (per-pixel/user-specified), where Porter-Duff (pre-multiplied) alpha is supported. The second stage is fading.

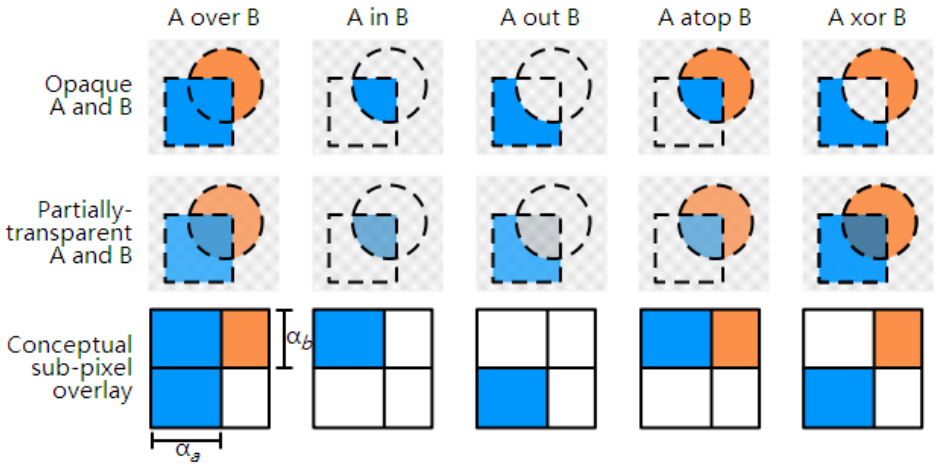


Fig. 3-6 RGA Alpha blanding

1. Mix (per-pixel/user-specified) alpha

$$data = (source \times (ALPHA + 1) + destination \times (255 - ALPHA)) >> 8$$

2. Porter-Duff

Porter-Duff alpha is a premultiplied alpha between two layers。

Porter-Duff formula:

$$C_r = C_s * F_s + C_d * F_d$$

$$A_r = A_s * F_s + A_d * F_d$$

(C – color, A – alpha, s – source, d – destination, r – result, F – factor)

There are 12 diferent mix types for Fs and Fd factor.

Table 3-2 RGA Porter-Duff alpha factor

NO.	type	Source factor	Destination factor
1	CLEAR	0	0
2	SRC	1	0
3	DST	0	1
4	SRC OVER	1	(1-As)
5	DST OVER	(1-Ad)	1
6	SRC IN	Ad	0
7	DST IN	0	As
8	SRC OUT	(1-Ad)	0
9	DST OUT	0	(1-As)
10	SRC ATOP	Ad	(1-As)

11	DST ATOP	(1-Ad)	As
12	XOR	(1-Ad)	(1-As)

3. Fading

$$data = ((source \times (ALPHA + 1) >> 8) + fading\_offset$$

3.3.7 Raster Operation (ROP)

Raster operation (ROP) is a Boolean operation between operands, which involve AND, OR, XOR, and NOT operations. For ROP2, operands are P (select pan) and D (Destination bitmap). For ROP3, operands are P (pattern), S (source bitmap) and D (Destination bitmap). For ROP4, operands are P (pattern), S (source bitmap), D (Destination bitmap) and MASK.

Table 3-3 RGA ROP Boolean operations

Operator	Meaning
a	Bitwise AND
n	Bitwise NOT (inverse)
o	Bitwise OR
x	Bitwise exclusive OR (XOR)

3.4 Register Description

3.4.1 Internal Address Mapping

Slave address can be divided into different length for different usage, which is shown as follows.

3.4.2 Registers Summary

Name	Offset	Size	Reset Value	Description
RGA_SYS_CTRL	0x0000	W	0x00000000	RGA system control register
RGA_CMD_CTRL	0x0004	W	0x00000000	RGA command code control
RGA_CMD_ADDR	0x0008	W	0x00000000	RGA command codes start address register
RGA_STATUS	0x000c	W	0x00000000	RGA status register
RGA_INT	0x0010	W	0x00000000	RGA interrupt register
RGA_AXI_ID	0x0014	W	0x49854210	RGA AXI ID setting register
RGA_MMU_STA_CTRL	0x0018	W	0x00000000	RGA MMU statistic ctrl
RGA_MMU_STA	0x001c	W	0x00000000	RGA MMU statistic data
RGA_WORK_CNT	0x0020	W	0x00000000	RGA work cycle counter
RGA_VERSION	0x0028	W	0x02018632	RGA version num
RGA_MODE_CTRL	0x0100	W	0x00000000	RGA mode control register
RGA_SRC_Y_MST	0x0104	W	0x00000000	Source image Y/RGB/line drawing start addr
RGA_SRC_CB_MST	0x0108	W	0x00000000	Source image Cb/Cbr start addr
RGA_SRC_CR_MST	0x010c	W	0x00000000	Source image Cr/color palette start addr
RGA_SRC_VIR_INFO	0x0110	W	0x00000000	Source image virtual width

Name	Offset	Size	Reset Value	Description
RGA_SRC_ACT_INFO	0x0114	W	0x00000000	Source image active width/height
RGA_SRC_X_PARA	0x0118	W	0x00000000	Source image horizontal scaling/rotation parameter
RGA_SRC_Y_PARA	0x011c	W	0x00000000	Source image vertical scaling/rotation parameter
RGA_SRC_TILE_XINFO	0x0120	W	0x00000000	Source tile start point coordinate,Source tile width
RGA_SRC_TILE_YINFO	0x0124	W	0x00000000	Source tile start point coordinate,Source tile height
RGA_SRC_TILE_H_INCR	0x0128	W	0x00000000	Source tile horizontal X/Y increment value
RGA_SRC_TILE_V_INCR	0x012c	W	0x00000000	Source tile vertical X/Y increment value
RGA_SRC_TILE_OFFSETX	0x0130	W	0x00000000	Source tile start point x for DST tile start point remap
RGA_SRC_TILE_OFFSETY	0x0134	W	0x00000000	Source tile start point y for DST tile start point remap
RGA_SRC_BG_COLOR	0x0138	W	0x00000000	Source image background color
RGA_SRC_FG_COLOR	0x013c	W	0x00000000	Source image foreground color
RGA_SRC_TR_COLOR0	0x0140	W	0x00000000	Source image transparency color min value
RGA_CP_GR_A	0x0140	W	0x00000000	RGA color gradient fill step register (color fill mode)
RGA_SRC_TR_COLOR1	0x0144	W	0x00000000	Source image transparency color max value,Color gradient fill st
RGA_CP_GR_B	0x0144	W	0x00000000	RGA color gradient fill step register (color fill mode)
RGA_LINE_DRAW	0x0148	W	0x00000000	Point/line drawing setting
RGA_PAT_ST_POINT	0x0148	W	0x00000000	RGA pattern start point
RGA_DST_MST	0x014c	W	0x00000000	Destination image start addr
RGA_DST_VIR_INFO	0x0150	W	0x00000000	Destination image virtual width
RGA_DST_CTR_INFO	0x0154	W	0x00000000	Destination image control window active width/height
RGA_ALPHA_CON	0x0158	W	0x00000000	Alpha blending/ROP mode register
RGA_PAT_CON	0x015c	W	0x00000000	Pattern size/offset
RGA_DST_VIR_WIDTH	0x015c	W	0x00000000	Register0000 Abstract
RGA_ROP_CON0	0x0160	W	0x00000000	Raster operation code0 control register
RGA_CP_GR_G	0x0160	W	0x00000000	Color gradient fill step of green
RGA_PRESCL_CB_MST	0x0160	W	0x00000000	RGA pre-scale Cb destination start addr
RGA_ROP_CON1	0x0164	W	0x00000000	Raster operation code1 control register

Name	Offset	Size	Reset Value	Description
RGA_CP_GR_R	0x0164	W	0x00000000	Color gradient fill step of red
RGA_PRESCL_CR_MST	0x0164	W	0x00000000	RGA pre-scale Cr destination start addr
RGA_MMU_CTRL	0x0168	W	0x00000000	MMU control register
RGA_MMU_TLB	0x016c	W	0x00000000	RGA MMU TLB base address
RGA_RGA_YUV_OUT_CFG	0x0170	W	0x00000000	RGA Destination output configuration
RGA_RGA_DST_UV_MST	0x0174	W	0x00000000	RGA Destination image UV start addr

*Notes:Size:**B**- Byte (8 bits) access, **HW**- Half WORD (16 bits) access, **W**-WORD (32 bits) access*

3.4.3 Detail Register Description

RGA_SYS_CTRL
Address: Operational Base + offset (0x0000)
RGA system control register

Bit	Attr	Reset Value	Description
31:4	RO	0x0	reserved
3	RW	0x0	acg_en RGA auto clock gating enable bit 1'b0：disable 1'b1：enable
2	RW	0x0	cmd_mode RGA command mode 1'b0：slave mode 1'b1：master mode
1	W1C	0x0	op_st RGA operation start bit Only used in passive (slave) control mode
0	W1C	0x0	soft_reset RGA soft reset write '1' to this would reset the RGA engine except config registers.

RGA_CMD_CTRL
Address: Operational Base + offset (0x0004)
RGA command code control

Bit	Attr	Reset Value	Description
31:13	RO	0x0	reserved
12:3	RW	0x000	cmd_incr_num RGA command increment number
2	RW	0x0	cmd_stop RGA command stop mode Command execution would stop after the current graphic operation finish if set this bit to '1'.

Bit	Attr	Reset Value	Description
1	W1C	0x0	cmd_incr_valid RGA command increment valid (Auto cleared) When setting this bit, 1. The total cmd number would increase by the RGA_INCR_CMD_NUM. 2. RGA would continue running if idle.
0	W1C	0x0	cmd_line_fet_st RGA command line fetch start (command line reset) (Auto cleared) When fetch start, the total cmd number would reset to RGA_INCR_CMD_NUM.

RGA_CMD_ADDR
Address: Operational Base + offset (0x0008)
RGA command codes start address register

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	cmd_addr RGA command codes start address

RGA_STATUS
Address: Operational Base + offset (0x000c)
RGA status register

Bit	Attr	Reset Value	Description
31:20	RO	0x000	cmd_total_num RGA command total number
19:8	RO	0x000	cur_cmd_num RGA current command number
7:1	RO	0x0	reserved
0	RO	0x0	engine_status RGA engine status 1'b0: idle 1'b1: working

RGA_INT
Address: Operational Base + offset (0x0010)
RGA interrupt register

Bit	Attr	Reset Value	Description
31:11	RO	0x0	reserved
10	RW	0x0	all_cmd_finish_int_en All command finished interrupt enable
9	RW	0x0	mmu_int_en MMU interrupt enable
8	RW	0x0	error_int_en Error interrupt enable

Bit	Attr	Reset Value	Description
7	W1C	0x0	cur_cmd_finish_int_clr Current command finished interrupt clear(auto clear)
6	W1C	0x0	all_cmd_finish_int_clr All command finished interrupt clear(auto clear)
5	W1C	0x0	mmu_int_clr MMU interrupt clear(auto clear)
4	W1C	0x0	error_int_clr Error interrupt clear(auto clear)
3	RO	0x0	cur_cmd_finish_int_flag Current command finished interrupt flag
2	RO	0x0	all_cmd_finish_int_flag All command finished interrupt flag
1	RO	0x0	mmu_int_flag MMU interrupt flag
0	RO	0x0	error_int_flag Error interrupt flag

RGA_AXI_ID
Address: Operational Base + offset (0x0014)
RGA AXI ID setting register

Bit	Attr	Reset Value	Description
31:30	RW	0x1	mmu_rid MMU read channel address maping axi bus ID Note: Don't use the same ID with RGA axi bus ID.2'b11, [31:30].
29:28	RW	0x0	mmu_wid MMU write channel address maping axi bus ID Note: Don't use the same ID with RGA axi bus ID.2'b11, [29:28].
27:24	RW	0x9	mask_rid mask read AXI ID
23:20	RW	0x8	cmd_rid CMD fetch AXI ID
19:16	RW	0x5	dst_wid DST write AXI ID
15:12	RW	0x4	dst_rid DST/LUT/PAT read AXI ID
11:8	RW	0x2	src_cr_rid SRC Cr read AXI ID
7:4	RW	0x1	src_cb_rid SRC Cb read AXI ID
3:0	RW	0x0	src_yrgb_rid SRC YRGB read AXI ID

RGA_MMU_STA_CTRL
Address: Operational Base + offset (0x0018)
RGA MMU statistic ctrl

Bit	Attr	Reset Value	Description
31:4	RO	0x0	reserved
3	W1C	0x0	mmu_sta_cnt_clr TLB statistic counter clear(auto clear) 1'b0: no clear 1'b1: clear After be set to 1, this bit will clear by itself 1 cycle later
2	W1C	0x0	mmu_sta_resume TLB statistic resume(auto clear) After be set to 1, this bit will clear by itself 1 cycle later.
1	W1C	0x0	mmu_sta_pause TLB statistic pause Note: before reading MMU_TLB_STATISTIC, this bit must be set to 1.
0	RW	0x0	mmu_sta_en TLB statistic enable 1'b0: disable 1'b1: enable

RGA_MMU_STA
Address: Operational Base + offset (0x001c)
RGA MMU statistic data

Bit	Attr	Reset Value	Description
31:16	RO	0x0000	tlb_miss_sta_cnt TLB miss statistic counter
15:0	RO	0x0000	tlb_hit_sta_cnt TLB hit statistic counter

RGA_MODE_CTRL
Address: Operational Base + offset (0x0100)
RGA mode control register

Bit	Attr	Reset Value	Description
31	RW	0x0	alpha_source_sel alpha select for alpha mix mode 1'b0: SRC alpha: SRC*As+DST*(1-As) 1'b1: DST alpha: SRC*(1-Ad)+DST*Ad
30	RW	0x0	alpha_zero_key_mode ARGB888 alpha zero key mode 0x000000 would be changed to 0x000100(RGB888)/0x0020(RGB565)for ARGB888 to RGBX/RGB565 color key 1'b0: disable 1'b1: enable
29	RW	0x0	cur_cmd_finish_int_en Current command finished interrupt enable

Bit	Attr	Reset Value	Description
28	RW	0x0	endian_swap Color palette endian swap 1'b0: big endian 1'b1: little endian
27	RW	0x0	dst_alpha_swap Destination bitmap data alpha swap 1'b0: ABGR 1'b1: BGRA
26	RW	0x0	dst_rb_swap Destination bitmap data RB swap 1'b0: BGR 1'b1: RGB
25	RW	0x0	dst_rgb_pack Destination bitmap BGR packed 1'b0: ABGR 1'b1: BGR packed
24:23	RW	0x0	dst_data_fmt Destination bitmap data format(Collor fill/ROP pattern data format) 2'b00: XBGR888/ABGR888 2'b01: RGB565 2'b10: ARGB1555 2'b11: ARGB4444
22	RW	0x0	pat_mode Color fill/ROP4 pattern 1'b0: solid color 1'b1: pattern color
21:20	RW	0x0	src_filter_type SRC rotation/mirror mode[3:2]: filter type 2'b00: nearest neighbor 2'b01: bi-linear 2'b10: bi-cubic
19:18	RW	0x0	src_rotate_mode SRC rotation/mirror mode[1:0] 2'b00: bypass 2'b01: rotation 2'b10: x mirror 2'b11: y mirror
17:14	RW	0x0	src_trans_en Source transparency enable bits [3]: A value stencil test enable bit [2]: B value stencil test enable bit [1]: G value stencil test enable bit [0]: R value stencil test enable bit

Bit	Attr	Reset Value	Description
13	RW	0x0	src_trans_mode Source color key mode 1'b0: normal stencil test 1'b1: inverted stencil test
12:11	RW	0x0	src_yuv2rgb_mode Source bitmap YUV2RGB conversion mode 2'b00: BT.601-MPEG 2'b01: BT.601-JPEG 2'b10: BT.709 2'b11: BT.601-MPEG
10	RW	0x0	src_uv_swap Source Cb-Cr swap 1'b0: CrCb 1'b1: CbCr
9	RW	0x0	src_alpha_swap Source bitmap data alpha swap 1'b0: ABGR 1'b1: BGRA
8	RW	0x0	src_rb_swap Source bitmap data RB swap 1'b0: BGR 1'b1: RGB
7:4	RW	0x0	src_data_fmt Source bitmap data format 4'b0000: XBGR888/ABGR888 4'b0001: RGB565 4'b0010: ARGB1555 4'b0011: ARGB4444 4'b0100: YUV422SP 4'b0101: YUV422P 4'b0110: YUV420SP 4'b0111: YUV420P 4'b1000: 1BPP (color palette) 4'b1001: 2BPP (color palette) 4'b1010: 4BPP (color palette) 4'b1011: 8BPP (color palette)
3	RW	0x0	src_rga_pack Source bitmap RGB packed 1'b0: ABGR 1'b1: BGR packed

Bit	Attr	Reset Value	Description
2:0	RW	0x0	render_mode RGA 2D render mode 3'b000: Bitblt 3'b001: Color palette 3'b010: Color fill（pattern fill） 3'b011: Line/point drawing 3'b100: Blur/sharp filter 3'b101: Pre-scaling 3'b110: Update palette LUT 111: Update pattern buffer

RGA_SRC_Y_MST
Address: Operational Base + offset (0x0104)
Source image Y/RGB/line drawing start addr

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	src_y_mst RGA source image Y/RGB start address register

RGA_SRC_CB_MST
Address: Operational Base + offset (0x0108)
Source image Cb/Cbr start addr

Bit	Attr	Reset Value	Description

31:0	RW	0x00000000	src_cb_mst
			RGA source image Cb/Cbr start address register
			source image Cb start address(YUV422/420-P);
			source image Cb/Cr start address(YUV422/420-SP);
			mask start address in ROP4 mode

RGA_SRC_CR_MST
Address: Operational Base + offset (0x010c)
Source image Cr/color palette start addr

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	src_cr_mst source image Cr start address(YUV422/420-P)

RGA_SRC_VIR_INFO
Address: Operational Base + offset (0x0110)
Source image virtual width

Bit	Attr	Reset Value	Description
31:14	RO	0x0	reserved
13:0	RW	0x0000	src_vir_stride source image virtual stride(words)

RGA_SRC_ACT_INFO
Address: Operational Base + offset (0x0114)

Source image active width/height

Bit	Attr	Reset Value	Description
31:29	RO	0x0	reserved
28:16	RW	0x0000	src_act_height source image active height
15:13	RO	0x0	reserved
12:0	RW	0x0000	src_act_width source image active width

RGA_SRC_X_PARA
Address: Operational Base + offset (0x0118)
Source image horizontal scaling/rotation parameter

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	src_h_para1 Source image horizontal scaling/rotation parameter1 sin(a)/ZoomX (signed 2.14)
15:0	RW	0x0000	src_h_para0 Source image horizontal scaling/rotation parameter0 cos(a)/ZoomX (signed 2.14)

RGA_SRC_Y_PARA
Address: Operational Base + offset (0x011c)
Source image vertical scaling/rotation parameter

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	src_v_para1 Source image vertical scaling/rotation parameter1 cos(a)/ZoomY (signed 2.14)
15:0	RW	0x0000	src_v_para0 Source image vertical scaling/rotation parameter0 -sin(a)/ZoomY (signed 2.14)

RGA_SRC_TILE_XINFO
Address: Operational Base + offset (0x0120)
Source tile start point coordinate,Source tile width

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	src_tile_width Source tile width (unsigned 5.11)
15:0	RW	0x0000	src_tile_xst Source tile start point x coordinate (signed13.3)

RGA_SRC_TILE_YINFO
Address: Operational Base + offset (0x0124)
Source tile start point coordinate,Source tile height

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	src_tile_height Source tile height (unsigned 5.11)

Bit	Attr	Reset Value	Description
15:0	RW	0x0000	src_tile_yst Source tile start point y coordinate (signed13.3)

RGA_SRC_TILE_H_INCR
Address: Operational Base + offset (0x0128)
Source tile horizontal X/Y increment value

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	src_h_tile_y_incr Source horizontal tile Y increment value (signed 6.10)
15:0	RW	0x0000	src_h_tile_x_incr Source horizontal tile X increment value (signed 6.10)

RGA_SRC_TILE_V_INCR
Address: Operational Base + offset (0x012c)
Source tile vertical X/Y increment value

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	src_v_tile_y_incr Source vertical tile Y increment value (signed 6.10)
15:0	RW	0x0000	src_v_tile_x_incr Source vertical tile X increment value (signed 6.10)

RGA_SRC_TILE_OFFSETX
Address: Operational Base + offset (0x0130)
Source tile start point x for DST tile start point remap

Bit	Attr	Reset Value	Description
31:19	RO	0x0	reserved
18:0	RW	0x00000	src_tile_xoff Source tile start point offset X for DST tile start point remap(unsigned 5.14)

RGA_SRC_TILE_OFFSETY
Address: Operational Base + offset (0x0134)
Source tile start point y for DST tile start point remap

Bit	Attr	Reset Value	Description
31:19	RO	0x0	reserved
18:0	RW	0x00000	src_tile_yoff Source tile start point offset Y for DST tile start point remap(unsigned 5.14)

RGA_SRC_BG_COLOR
Address: Operational Base + offset (0x0138)
Source image background color

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	src_bg_color Source image background color "0" bit color for mono expansion.

RGA_SRC_FG_COLOR
Address: Operational Base + offset (0x013c)
Source image foreground color

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	src_fg_color Source image foreground color "1" bit color for mono expansion. Line/point color, Color fill color, Pan color

RGA_SRC_TR_COLOR0
Address: Operational Base + offset (0x0140)
Source image transparency color min value

Bit	Attr	Reset Value	Description
31:24	RW	0x00	src_trans_a_min source image transparency color A min value
23:16	RW	0x00	src_trans_b_min source image transparency color B min value
15:8	RW	0x00	src_trans_g_min source image transparency color G min value
7:0	RW	0x00	src_trans_r_min source image transparency color R min value

RGA_CP_GR_A
Address: Operational Base + offset (0x0140)
RGA color gradient fill step register (color fill mode)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	a_gr_y Y gradient value of Alpha (signed 8.8)
15:0	RW	0x0000	a_gr_x X gradient value of Alpha (signed 8.8)

RGA_SRC_TR_COLOR1
Address: Operational Base + offset (0x0144)
Source image transparency color max value,Color gradient fill st

Bit	Attr	Reset Value	Description
31:24	RW	0x00	src_trans_a_max source image transparency color A max value
23:16	RW	0x00	src_trans_b_max source image transparency color B max value
15:8	RW	0x00	src_trans_g_max source image transparency color G max value

Bit	Attr	Reset Value	Description
7:0	RW	0x00	src_trans_r_max source image transparency color R max value

RGA_CP_GR_B
Address: Operational Base + offset (0x0144)
RGA color gradient fill step register (color fill mode)

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	b_gr_y Y gradient value of Blue (signed 8.8)
15:0	RW	0x0000	b_gr_x X gradient value of Blue (signed 8.8)

RGA_LINE_DRAW
Address: Operational Base + offset (0x0148)
Point/line drawing setting

Bit	Attr	Reset Value	Description
31	RW	0x0	line_draw_aa Line drawing Anti-alising operation 1'b0: disable 1'b1: enable
30	RW	0x0	line_draw_last_point_en Line drawing last point drawing 1'b0: Don't draw 1'b1: Draw
29	RW	0x0	line_draw_semi_dir Direction of semi-major axis 1'b0: Increase 1'b1: Decrease
28	RW	0x0	line_draw_major_dir Direction of major axis 1'b0: Increase 1'b1: Decrease
27:16	RW	0x000	line_draw_incr Line drawing X/Y delta step (unsigned 0.12) X delta step value if X is major axis; Y delta step value if Y is major axis;
15:12	RW	0x0	line_draw_width Line width (1~16 pixel)
11	RW	0x0	line_draw_dir Line drawing direction 1'b0: X is major axis 1'b1: Y is major axis

Bit	Attr	Reset Value	Description
10:0	RW	0x000	line_draw_length Line drawing X/Y length of line (unigned 11) X length if X is major axis Y length if Y is major axis

RGA_PAT_ST_POINT
Address: Operational Base + offset (0x0148)
RGA pattern start point

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:0	RW	0x00	pat_st_point Pattern start point in pattern ram

RGA_DST_MST
Address: Operational Base + offset (0x014c)
Destination image start addr

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	dst_mst destiniation image RGB start address source image color palette table start address(color palette mode)

RGA_DST_VIR_INFO
Address: Operational Base + offset (0x0150)
Destination image virtual width

Bit	Attr	Reset Value	Description
31:30	RO	0x0	reserved
29:28	RW	0x0	dst_alpha_sel DST Per pixel alpha or user set alpha 2'b00: user set alpha 2'b01: per pixel alpha 2'b10: per pixel alpha & user set alpha 2'b11: un-defined
27:16	RW	0x000	mask_vir_stride mask image virtual stride[6:0] (words) destination image virtual height[11:0] for line point drawing
15:12	RO	0x0	reserved
11:0	RW	0x000	dst_vir_stride destination image virtual stride(words)

RGA_DST_CTR_INFO
Address: Operational Base + offset (0x0154)
Destination image control window active width/height

Bit	Attr	Reset Value	Description
31	RO	0x0	reserved

Bit	Attr	Reset Value	Description
30:29	RW	0x0	alpha_out_sel mix alpha mode alpha output When mix alpha mode is select Control the alpha output: 2'b00: alphasrc 2'b01: alphadst 2'b10: alphasrc + alphadst - (alphasrc* alphadst >>8)
28	RW	0x0	dither_type_sel Dither down type sel 1'b0: select android dither down method 1'b1: select Allegro dither down method
27:16	RW	0x000	dst_ctrl_win_height destination image control window height (11bits) Start Y in line drawing mode (11bits) Pre_scaling active height in pre_scaling mode (12bits)
15:12	RO	0x0	reserved
11:0	RW	0x000	dst_ctrl_win_width destination image control window width (11bits) Start X in line drawing mode (11bits) Pre_scaling active width in pre_scaling mode (12bits)

RGA_ALPHA_CON
Address: Operational Base + offset (0x0158)
Alpha blending/ROP mode register

Bit	Attr	Reset Value	Description
31	RW	0x0	ratate_aa Rotation AA mode 1'b0:no AA 1'b1:AA
30	RW	0x0	gr_cal_mode Gradient calculation mode 1'b0:clip 1'b1:not-clip
29	RW	0x0	dither_down_en Dither_down_en 1'b0:disable 1'b1:enable
28	RW	0x0	alpah_cal_sel Alpha_cal_sel 1'b0:alpha' = alpha + (alpha>>7) 1'b1:alpha' = alpha

Bit	Attr	Reset Value	Description
27:26	RW	0x0	bs_filter_type Blur/sharp Filter type 2'b00:weakest 2'b01:weaker 2'b10:stronger 2'b11:strongest
25	RW	0x0	bs_mode Blur/sharp filter mode 1'b0: Blur 1'b1: sharp
24	RW	0x0	pre_scl_yuv_out_fmt Pre-scale YCbCr output format 1'b0: The same with source format 1'b1: all is semi-planar
23:22	RW	0x0	pre_scl_v_ratio Pre_scaler vertical scaling ratio: 2'b00: 1 2'b01: 1/2 2'b10: 1/4 2'b11: 1/8
21:20	RW	0x0	pre_scl_h_ratio Pre_scaler horizontal scaling ratio: 2'b00: 1 2'b01: 1/2 2'b10: 1/4 2'b11: 1/8
19:18	RW	0x0	rop_mode ROP mode select 2'b00: ROP 2 2'b01: ROP 3 2'b10: ROP 4
17	RW	0x0	fading_en Fading enable 1'b0: disable 1'b1: enable
16	RW	0x0	mix_alpha_mode Mix alpha mode or porter-duff alpha mode
15:8	RW	0x00	user_set_alpha User set alpha constant value/fading_alpha_value

Bit	Attr	Reset Value	Description
7:4	RW	0x0	port_duff_mode Porter-duff mode 4'd0: CLEAR 4'd1: SRC 4'd2: DST 4'd3: SRC OVER 4'd4: DST OVER 4'd5: SRC IN 4'd6: DST IN 4'd7: SRC OUT 4'd8: DST OUT 4'd9: SRC ATOP 4'd10: DST ATOP 4'd11: XOR
3:2	RW	0x0	src_alpah_sel Source per pixel alpha or user set alpha 2'b00: user set alpha 2'b01: per pixel alpha 2'b10: per pixel alpha & user set alpha 2'b11: un-defined
1	RW	0x0	alpha_rop_sel Alpha or ROP sel: 1'b0: alpha 1'b1: ROP
0	RW	0x0	alpha_rop_en Alpha or ROP enable 1'b0: disable 1'b1: enable

RGA_PAT_CON
Address: Operational Base + offset (0x015c)
Pattern size/offset

Bit	Attr	Reset Value	Description
31:24	RW	0x00	pat_yoff Pattern y offset
23:16	RW	0x00	pat_xoff Pattern x offset
15:8	RW	0x00	pat_height Pattern height
7:0	RW	0x00	pat_width Pattern width Pattern total number when doing pattern load

RGA_DST_VIR_WIDTH
Address: Operational Base + offset (0x015c)

Register0000 Abstract

Bit	Attr	Reset Value	Description
31:12	RO	0x0	reserved
11:0	RW	0x000	dst_vir_width_pixel destination image virtual width(pixel)

RGA_ROP_CON0

Address: Operational Base + offset (0x0160)

Raster operation code0 control register

Bit	Attr	Reset Value	Description
31:24	RO	0x0	reserved
23:0	RW	0x000000	rop3_code0 Rop3 code 0 control bits

RGA_CP_GR_G

Address: Operational Base + offset (0x0160)

Color gradient fill step of green

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	g_gr_y Y gradient value of Green (signed 8.8)
15:0	RW	0x0000	g_gr_x X gradient value of Green (signed 8.8)

RGA_PRESCL_CB_MST

Address: Operational Base + offset (0x0160)

RGA pre-scale Cb destination start addr

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	pre_scl_uv_dst_mst Pre-scale Cb/Cr destination start addr

RGA_ROP_CON1

Address: Operational Base + offset (0x0164)

Raster operation code1 control register

Bit	Attr	Reset Value	Description
31:24	RO	0x0	reserved
23:0	RW	0x000000	rop3_code1 Rop3 code 1 control bits

RGA_CP_GR_R

Address: Operational Base + offset (0x0164)

Color gradient fill step of red

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	r_gr_y X gradient value of Red (signed 8.8)
15:0	RW	0x0000	r_gr_x X gradient value of Red (signed 8.8)

RGA_PRESCL_CR_MST
Address: Operational Base + offset (0x0164)
RGA pre-scale Cr destination start addr

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	pre_scl_v_dst_mst Pre-scale Cr destination start addr

RGA_MMU_CTRL
Address: Operational Base + offset (0x0168)
MMU control register

Bit	Attr	Reset Value	Description
31:24	RW	0x00	fading_b Fading offset B value
23:16	RW	0x00	fading_g Fading offset G value
15:8	RW	0x00	fading_r Fading offset R value
7:6	RO	0x0	reserved
5:4	RW	0x0	mmu_page_table_size RGA MMU Page table size 2'b00: 1KB page 2'b01: 2KB page 2'b10: 4KB page 2'b11: 8KB page
3	W1C	0x0	cmd_flush_en MMU TLB CMD channel flush enable bit (auto clear) 1'b0: no flush 1'b1: flush
2	W1C	0x0	dst_flush_en MMU TLB DST channel flush enable bit (auto clear) 1'b0: no flush 1'b1: flush
1	W1C	0x0	src_flush_en MMU TLB SRC channel flush enable bit (auto clear) 1'b0: no flush 1'b1: flush
0	RW	0x0	mmu_en RGA MMU enable

RGA_MMU_TLB
Address: Operational Base + offset (0x016c)
RGA MMU TLB base address

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	mmu_tlb_addr RGA MMU TLB base address(word)

RGA_RGA_YUV_OUT_CFG
Address: Operational Base + offset (0x0170)
RGA Destination output configuration

Bit	Attr	Reset Value	Description
31:15	RW	0x00000	Reserved Reserved
14:8	RW	0x00	sw_axi_rw_align_e_n AXI WR/RD Align in line scan; Suggested value=7'b0, Align enable; [6:0] each bit value: 0, enable; 1: disable; [6]: DST BGR565/4444/1555 Read align disable [5]: DST ABGR888 Read align disable [4]: DST BGR565/4444/1555 Write align disable [3]: DST ABGR888 Write align disable [2]: SRC Y channel read Rlign disable [1]: SRC BGR565/4444/1555 Read align disable [0]: SRC ABGR888 Read align disable
7	RW	0x0	sw_cp_alpha_bit_sel alpha bits width select for ColorPalette; default=1'b0; 1'b1: 1bit alpha; 1'b0: 8bit alpha;
6	RW	0x0	sw_dst_csc_clip BGR2YUV Clip mode(from 0~255 clip to 36~235) 1'b1: clip enable; 1'b0: unclip
5:4	RW	0x0	sw_dst_csc_mode DST bitmap RGB2YUV conversion mode 2'b00: Bypass mode, only used in y2y mode 2'b01: BT.601-range0 2'b10: BT.601-range1 2'b11: BT.709-range0
3	RW	0x0	sw_dst_uv_swap Destination Cb-Cr swap 1'b0: CrCb 1'b1: CbCr
2	RW	0x0	reserved reserved
1	RW	0x0	sw_dst_yuv_fmt Destination bitmap data format 1'b0: YUV422SP 1'b1: YUV420SP

Bit	Attr	Reset Value	Description
0	RW	0x0	sw_dst_yuv_en Field0000 Abstract Destination YUV output enable, Could not used Pre_scaling/Line-point drawing and Blur/Sharp, Alpha and ROP. 1'b1: enable; 1'b0: disable;

RGA_RGA_DST_UV_MST
Address: Operational Base + offset (0x0174)
RGA Destination image UV start addr

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	sw_dst_uv_mst destiniation image UV start address

3.5 Programming Guide

3.5.1 Register Partition

There are two types of register in RGA. The first 8 registers (0x0 - 0x1C) are general registers for system configuration including command mode, command parameter, RGA status, general interrupts. The other registers (range from 0x100~0x178) are command registers for command codes.

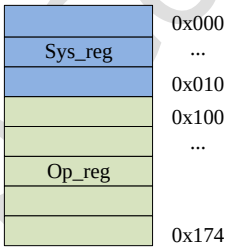


Fig. 3-7 HDMI TX Software Main Sequence Diagram

3.5.2 Command Modes

RGA has two command modes: slave mode and master mode. In salve mode (RGA_SYS_CTRL[2] = 1'b0), 2D graphic command only could be run one by one. CPU set all the command registers in RGA and then start RGA running by setting RGA_SYS_CTRL[2] to '1'. In master mode (RGA_SYS_CTRL[2] = 1'b1), 2D graphic commands could be run sequentially. After setting command's number to RGA_CMD_CTRL[12:3], writing '1' to RGA_CMD_CTRL[0] will start the command fetch, then Internal command DMA fetch commands from external command line.
Command line is a collection of several command codes with continuous address. At the first start, the command start address (RGA_CMD_ADDR) and command number (RGA_CMD_CTRL[12:3]) should be set, then write '1' to cmd_line_st (RGA_CMD_CTRL[0]) to start the command line fetch. Incremental command is supported by setting cmd_incr_num (RGA_CMD_CTRL[12:3]) and cmd_incr_valid (RGA_CMD_CTRL[1]=1'b1)

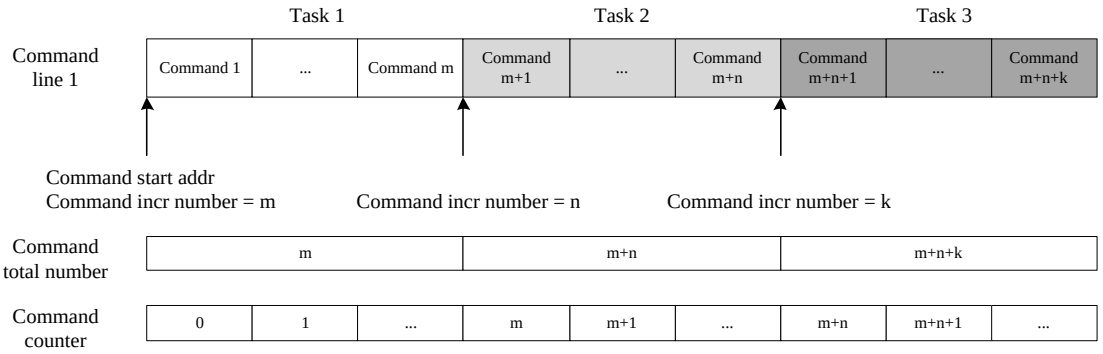


Fig. 3-8 RGA command line and command counter

3.5.3 Command Sync

In slave command mode, command sync is controlled by CPU.
In master command mode, user can enable the current_cmd_int (RGA_MODE_CTRL[25] = 1'b1) command by command to generate a interrupt at the end point of target command operation.

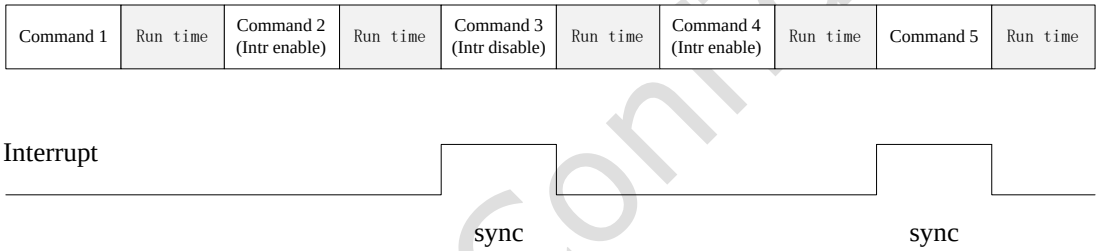


Fig. 3-9 RGA command sync generation

Chapter 4 Image Enhancement Processor（IEP）

4.1 Overview

The Image Enhancement Processor (IEP) receives data from system main memory and transmits data to system main memory by AXI bus.
The features of IEP are as follow:

- **Image format**
  - Input data: XRGB/RGB565/YUV420/YUV422
  - Output data: ARGB/RGB565/YUV420/YUV422
  - ARGB/XRGB/RGB565/YUV swap
  - YUV semi-planar/planar
  - BT601_l/BT601_f/BT709_l/BT709_f color space conversion
  - RGB dither up/down conversion
  - YUV up/down sampling conversion
  - Max resolution for static image up to 8192x8192
  - Max resolution for dynamic image up to 1920x1080
- **Enhancement**
  - Gamma adjustment with programmable mapping table
  - Hue/Saturation/Brightness/Contrast enhancement
  - Color enhancement with programmable coefficient
  - Detail enhancement with filter matrix up to 5x5
  - Edge enhancement with filter matrix up to 5x5
  - Programmable difference table for detail enhancement
  - Programmable distance table for detail and edge enhancement
- **Noise reduction**
  - Compression noise reduction with filter matrix up to 5x5
  - Programmable difference table for compression noise reduction
  - Programmable distance table for compression noise reduction
- **De-interlace**
  - Input 4 fields, output 2 frames mode
  - Input 4 fields, output 1 frames mode
  - Input 2 fields, output 1 frames mode
  - Programmable motion detection coefficient
  - Programmable high frequency factor
  - Programmable edge interpolation parameter
- **Interface**
  - Programmable direct path to VOP
  - 32bit AHB bus slave
  - 64bit AXI bus master
  - Combined interrupt output

4.2 Block Diagram

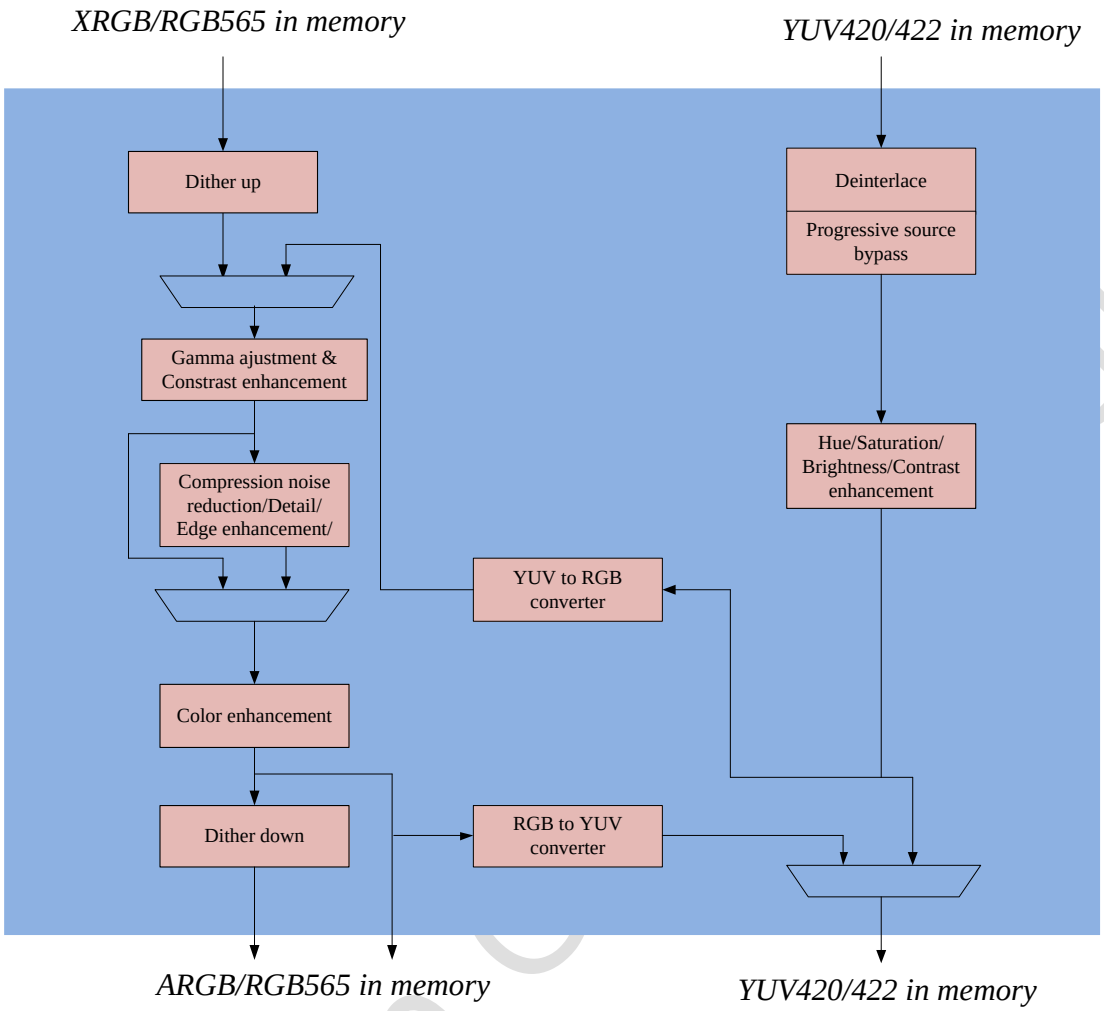


Fig. 4-1 IEP block diagram

The data path in IEP is in the previously diagram. The IEP comprises with:

- Deinterlace

There are five deinterlace mode including I4O2 (input 4 fields and output 2 frames once), I4O1B, I4O1T, I2O1B, I2O1T in the deinterlace block. YUV bypass is also supported. Pay attention if compression noise reduction, detail or edge enhancement work together with deinterlace is not allowed.

- Enhancement

Not only hue, saturation, brightness, contrast enhancement, but also blue screen, black screen and color bar are supported in YUV domain enhancement block. Gamma adjustment, edge enhancement, detail enhancement and color enhancement are supported in RGB domain enhancement block.

- Noise Reduction

Spatial and temporal sampling noise can be reduced in YUV domain noise reduction block. Compression noise can be reduced in RGB domain noise reduction block.

4.3 Function Description

4.3.1 Deinterlace

There are five deinterlace mode including I4O2, I4O1B, I4O1T, I2O1B and I2O1T in the deinterlace block. The I4O2 mode represents for 4 fields of input images and 2 frames of output images, so all of the two groups of source address registers and two groups of destination address registers need to be configured. For example, if source and destination format are both YUV420, the source address register IEP_SRC_ADDR_YRGB, IEP_SRC_ADDR_CBCR are used for source field0 and field 1, the source address register

IEP_SRC_ADDR_Y1, IEP_SRC_ADDR_CBCR1 are used for source field2 and field3. The I4O1B and I4O1T mode have the same input images as the I4O2 mode, but only one frame output is generated once. The I2O1B and I2O1T mode have the same output as I4O1B and I4O1T mode, but only two fields input are needed. If bypass mode is selected, there are not any deinterlace operations. The parameter dil_ei_sel, dil_ei_radius, dil_ei_smooth, dil_ei_mode, dil_hf_en and dil_hf_fct in register IEP_CONFIG0 and registers IEP_DIL_MTN_TAB0~7 may have different influence in deinterlace effect depend on the type of the image source.

4.3.2 Noise reduction

Compression noise reduction is used for reducing the noise after the decompression of picture or video. Before the compression noise reduction is enabled, the IEP_ENH_DDE_COE0/1 from address 0x400 to 0x5FC for difference and distance coefficients must be written firstly. The filter matrix can be selected from 3x3/5x5 and the filter weight can be programmed by configuring IEP_ENH_RGB_CNFG.

4.3.3 Enhancement

Not only hue, saturation, brightness, contrast enhancement, but also blue screen, black screen and color bar are supported in this block. IEP_ENH_YUV_CNFG_0/1/2 registers can be configured to modify the YUV enhance parameters to satisfied with the requirement. Before the gamma adjustment or contrast enhancement is enabled in RGB domain, the IEP_ENH_CG_TAB from address 0x100 to 0x3FC for B, G, R mapping must be written firstly. If the color enhancement is enabled, the IEP_ENH_C_COE must be written the required value.

Before the edge or detail enhancement is enabled, the IEP_ENH_DDE_COE0/1 from address 0x400 to 0x5FC for difference and distance coefficients must be written firstly. The filter matrix can be selected from 3x3/5x5 and the filter weight can be programmed by configuring IEP_ENH_RGB_CNFG.

4.3.4 Format conversion

The color space conversion either from RGB to YUV or from YUV to RGB has the selections including BT601/709_L/F mode, and the input can be clipped or not.

If the source format is RGB565, dither up must be enabled. In contrary to the destination format is RGB565, dither down must be enabled.

4.3.5 Shadow registers

The configuration registers can be configured at any time, but they cannot have any effect immediately unless config_done is available and a new frame_start is enabled. The registers IEP_RAW_CONFIG0/1, IEP_RAW_VIR_IMG_WIDTH, IEP_RAW_IMG_SCL_FCT, IEP_RAW_SRC_IMG_SIZE, IEP_RAW_ENH_YUV_CNFG_0/1/2 corresponding to the registers have the similar names but without letters _RAW. They are used for raw register value reading before the configurations really have effect on the new frame.

4.3.6 VOP direct path

The IEP_DST_ADDR for DMA writing is useless if vop_path_en bit is set, because all RGB or YUV data is supplied for VOP directly from local bus via VOP and IEP.

4.4 Register Description

Slave address can be divided into different length for different usage, which is shown as follows.

4.4.1 Registers Summary

Name	Offset	Size	Reset Value	Description
IEP_CONFIG0	0x0000	W	0x00000000	configuration register0
IEP_CONFIG1	0x0004	W	0x00000000	configuration register1
IEP_STATUS	0x0008	W	0x00000000	status register
IEP_INT	0x000c	W	0x00000000	interrupt register
IEP_FRM_START	0x0010	W	0x00000000	frame start
IEP_CONFIG_DONE	0x0018	W	0x00000000	configuration done

IEP_FRM_CNT	0x001c	W	0x00000000	frame counter
IEP_VIR_IMG_WIDTH	0x0020	W	0x01400140	Image virtual width
IEP_IMG_SCL_FCT	0x0024	W	0x20002000	scaling factor
IEP_SRC_IMG_SIZE	0x0028	W	0x00f00140	Source image width/height
IEP_DST_IMG_SIZE	0x002c	W	0x00f00140	Destination image width/height
IEP_DST_IMG_WIDTH_TILE0	0x0030	W	0x00000000	Destination image tile0 width
IEP_DST_IMG_WIDTH_TILE1	0x0034	W	0x00000000	Destination image tile1 width
IEP_DST_IMG_WIDTH_TILE2	0x0038	W	0x00000000	Destination image tile2 width
IEP_DST_IMG_WIDTH_TILE3	0x003c	W	0x00000000	Destination image tile3 width
IEP_ENH_YUV_CNFG_0	0x0040	W	0x00000000	brightness,contrast,saturation adjustment
IEP_ENH_YUV_CNFG_1	0x0044	W	0x00000000	Hue configuration
IEP_ENH_YUV_CNFG_2	0x0048	W	0x00000000	color bar configuration
IEP_ENH_RGB_CNFG	0x004c	W	0x00000000	enhancement RGB configuration
IEP_ENH_C_COE	0x0050	W	0x00000000	rgb color enhancement coefficient
IEP_SRC_ADDR_YRGB	0x0080	W	0x00000000	Start address of source image(Y/RGB)
IEP_SRC_ADDR_CBCR	0x0084	W	0x00000000	Start address of source image(Cb/Cr)
IEP_SRC_ADDR_CR	0x0088	W	0x00000000	Start address of source image(Cr)
IEP_SRC_ADDR_Y1	0x008c	W	0x00000000	Start address of source image(Y)
IEP_SRC_ADDR_CBCR1	0x0090	W	0x00000000	Start address of source image(Cb/Cr)
IEP_SRC_ADDR_CR1	0x0094	W	0x00000000	Start address of source image(Cr)
IEP_SRC_ADDR_Y_ITEMP	0x0098	W	0x00000000	Start address of source image(Y integer part)
IEP_SRC_ADDR_CBCR_ITEMP	0x009c	W	0x00000000	Start address of source image(CBCR integer part)
IEP_SRC_ADDR_CR_ITEMP	0x00a0	W	0x00000000	Start address of source image(CR integer part)
IEP_SRC_ADDR_Y_FTEMP	0x00a4	W	0x00000000	Start address of source image(Y fraction part)
IEP_SRC_ADDR_CBCR_FTEMP	0x00a8	W	0x00000000	Start address of source image(CBCR fraction part)
IEP_SRC_ADDR_CR_FTEMP	0x00ac	W	0x00000000	Start address of source image(CR fraction part)
IEP_DST_ADDR_YRGB	0x00b0	W	0x00000000	Start address of destination image(Y/RGB)
IEP_DST_ADDR_CBCR	0x00b4	W	0x00000000	Start address of destination image(Cb/Cr)

IEP_DST_ADDR_CR	0x00b8	W	0x00000000	Start address of destination image(Cr)
IEP_DST_ADDR_Y1	0x00bc	W	0x00000000	Start address of destination image(Y)
IEP_DST_ADDR_CBCR1	0x00c0	W	0x00000000	Start address of destination image(Cb/Cr)
IEP_DST_ADDR_CR1	0x00c4	W	0x00000000	Start address of destination image(Cr)
IEP_DST_ADDR_Y_ITEMP	0x00c8	W	0x00000000	Start address of destination image(Y integer part)
IEP_DST_ADDR_CBCR_ITEMP	0x00cc	W	0x00000000	Start address of destination image(CBCR integer part)
IEP_DST_ADDR_CR_ITEMP	0x00d0	W	0x00000000	Start address of destination image(CR integer part)
IEP_DST_ADDR_Y_FTEMP	0x00d4	W	0x00000000	Start address of destination image(Y fraction part)
IEP_DST_ADDR_CBCR_FTEMP	0x00d8	W	0x00000000	Start address of destination image(CBCR fraction part)
IEP_DST_ADDR_CR_FTEMP	0x00dc	W	0x00000000	Start address of destination image(CR fraction part)
IEP_DIL_MTN_TAB0	0x00e0	W	0x00000000	Deinterlace motion table0
IEP_DIL_MTN_TAB1	0x00e4	W	0x00000000	Deinterlace motion table1
IEP_DIL_MTN_TAB2	0x00e8	W	0x00000000	Deinterlace motion table2
IEP_DIL_MTN_TAB3	0x00ec	W	0x00000000	Deinterlace motion table3
IEP_DIL_MTN_TAB4	0x00f0	W	0x00000000	Deinterlace motion table4
IEP_DIL_MTN_TAB5	0x00f4	W	0x00000000	Deinterlace motion table5
IEP_DIL_MTN_TAB6	0x00f8	W	0x00000000	Deinterlace motion table6
IEP_DIL_MTN_TAB7	0x00fc	W	0x00000000	Deinterlace motion table7
IEP_ENH_CG_TAB	0x0100	W	0x00000000	contrast and gamma enhancement table
IEP_ENH_DDE_COE0	0x0400	W	0x00000000	denoise,detail and edge enhancement coefficient
IEP_ENH_DDE_COE1	0x0500	W	0x00000000	denoise,detail and edge enhancement coefficient
IEP_MMU_DTE_ADDR	0x0800	W	0x00000000	MMU current page table address
IEP_MMU_STATUS	0x0804	W	0x00000018	MMU status register
IEP_MMU_CMD	0x0808	W	0x00000000	MMU command register
IEP_MMU_PAGE_FAULT_ADDR	0x080c	W	0x00000000	MMU logic address of last page fault
IEP_MMU_ZAP_ONE_LINE	0x0810	W	0x00000000	MMU zap cache line register
IEP_MMU_INT_RAWSTAT	0x0814	W	0x00000000	MMU raw interrupt status register
IEP_MMU_INT_CLEAR	0x0818	W	0x00000000	MMU interrupt clear register
IEP_MMU_INT_MASK	0x081c	W	0x00000000	MMU interrupt mask register
IEP_MMU_INT_STATUS	0x0820	W	0x00000000	MMU interrupt status register
IEP_MMU_AUTO_GATING	0x0824	W	0x00000001	MMU clock auto gating register

*Notes:Size:**B**- Byte (8 bits) access, **HW**- Half WORD (16 bits) access, **W**-WORD (32 bits) access*

4.4.2 Detail Register Description

IEP_CONFIG0

Address: Operational Base + offset (0x0000)

configuration register0

Bit	Attr	Reset Value	Description
31:24	RO	0x0	reserved
23	RW	0x0	dil_ei_sel deinterlace edge interpolation select
22:21	RW	0x0	dil_ei_radius deinterlace edge interpolation radius
20	RW	0x0	rgb_con_gam_order RGB contrast enhancement and gamma adjustment operation order select. 1'b0:CG prior to DDE 1'b1:DDE prior to CG (CG represent for contrast & gamma operation, and DDE represent for denoise, detail or edge enhancement operation)
19:18	RW	0x0	rgb_enh_sel RGB enhancement select 2'b00: no operation 2'b01: denoise 2'b10: detail enhancement 2'b11: edge enhancement
17	RW	0x0	rgb_con_gam_en RGB contrast enhancement and gamma adjustment enable 1'b0:disable 1'b1:enable
16	RW	0x0	rgb_color_enh_en RGB color enhancement enable 1'b0:disable 1'b1:enable
15	RW	0x0	dil_ei_smooth deinterlace edge interpolation for smooth effect 1'b0:disable 1'b1:enable
14	RW	0x0	yuv_enh_en yuv enhancement enable 1'b0:disable 1'b1:enable
13	RW	0x0	yuv_dns_en YUV 3D denoise enable 1'b0:disable 1'b1:enable

12	RW	0x0	dil_ei_mode deinterlace edge interpolation 1'b0:disable 1'b1:enable
11	RW	0x0	dil_hf_en deinterlace high frequency calculation enable 1'b0:disable 1'b1:enable
10:8	RW	0x0	dil_mode Deinterlace mode select: 3'b000: YUV deinterlace and bypass path disable; 3'b001: I4O2 mode 3'b010: I4O1B mode 3'b011: I4O1T mode 3'b100: I2O1B mode 3'b101: I2O1T mode 3'b110: bypass mode
7:1	RW	0x00	dil_hf_fct deinterlace high frequency factor
0	RW	0x0	vop_path_en VOP direct path enable 1'b0:disable 1'b1:enable

IEP_CONFIG1

Address: Operational Base + offset (0x0004)

configuration register1

Bit	Attr	Reset Value	Description
31:24	RW	0x00	glb_alpha global alpha value only valid when destination format is ARGB
23	RW	0x0	rgb2yuv_input_clip RGB to YUV input range 1'b0:R/G/B=[0,255] 1'b1:R/G/B=[16,235]
22	RW	0x0	yuv2rgb_input_clip YUV to RGB input range 1'b0:Y/U/V=[0,255] 1'b1:Y=[16,235],U/V=[16,240]
21	RW	0x0	rgb_to_yuv_en RGB to YUV conversion enable 1'b0:disable 1'b1:enable

20	RW	0x0	yuv_to_rgb_en YUV to RGB conversion enable 1'b0:disable 1'b1:enable
19:18	RW	0x0	rgb2yuv_coe_sel rgb2yuv coefficient select 2'b00:bt601_1 2'b01:bt601_f 2'b10:bt709_1 2'b11:bt709_f
17:16	RW	0x0	yuv2rgb_coe_sel yuv2rgb coefficient select 2'b00:bt601_1 2'b01:bt601_f 2'b10:bt709_1 2'b11:bt709_f
15	RW	0x0	dthr_down_en dither down enable 1'b0:disable 1'b1:enable
14	RW	0x0	dthr_up_en dither up enable 1'b0:disable 1'b1:enable
13:12	RW	0x0	dst_yuv_swap destination YUV swap 2'b00:SP UV 2'b01:SP VU 2'b10, 2'b11:P
11:10	RW	0x0	dst_rgb_swap destination RGB swap ARGB destination 2'b00:ARGB 2'b01:ABGR 2'b10:RGBA 2'b11:BGRA RGB565 destination 2'b00,2'b10:RGB 2'b01, 2'b11:BGR
9:8	RW	0x0	dst_fmt Ouput image Format 2'b00 : ARGB 2'b01 : RGB565 2'b10 : YUV422 2'b11 : YUV420

7:6	RO	0x0	reserved
5:4	RW	0x0	src_yuv_swap source YUV swap 2'b00:SP UV 2'b01:SP VU 2'b10, 2'b11:P
3:2	RW	0x0	src_rgb_swap source RGB swap XRGB source 2'b00:XRGB 2'b01:XBGR 2'b10:RGBX 2'b11:BGRX RGB565 source 2'b00,2'b10:RGB 2'b01,2'b11:BGR
1:0	RW	0x0	src_fmt Input image Format 2'b00 : XRGB 2'b01 : RGB565 2'b10 : YUV422 2'b11 : YUV420

IEP_STATUS
Address: Operational Base + offset (0x0008)
status register

Bit	Attr	Reset Value	Description
31:20	RO	0x0	reserved
19	RO	0x0	rrgb_idle_ack RGB read DMA idle acknowlege
18	RO	0x0	wrgb_idle_ack RGB write DMA idle acknowlege
17	RO	0x0	ryuv_idle_ack YUV read DMA idle acknowlege
16	RO	0x0	wyuv_idle_ack YUV write DMA idle acknowlege
15:9	RO	0x0	reserved
8	RO	0x0	voi_sts vop direct path status
7	RO	0x0	rrgb_sts RGB DMA read status
6	RO	0x0	wrgb_sts RGB DMA write status
5	RO	0x0	ryuv_sts YUV DMA read status

4	RO	0x0	wyuv_sts YUV DMA write status
3	RO	0x0	dde_sts RGB denoise/enhancement status
2	RO	0x0	dil_sts de-interlace or yuv bypass status
1	RO	0x0	scl_sts scaling status
0	RO	0x0	dns_sts YUV 3D denoise status

IEP_INT
Address: Operational Base + offset (0x000c)
interrupt register

Bit	Attr	Reset Value	Description
31:17	RO	0x0	reserved
16	W1C	0x0	frm_done_int_clr Frame process done interrupt clear After be set to 1, this bit will be clear automatically.
15:9	RO	0x0	reserved
8	RW	0x0	frm_done_int_en Frame process done interrupt enable: 1'b0:disable 1'b1:enable
7:1	RO	0x0	reserved
0	RO	0x0	frm_done_int Frame process done interrupt 1'b0: inactive; 1'b1: active;

IEP_FRM_START
Address: Operational Base + offset (0x0010)
frame start

Bit	Attr	Reset Value	Description
31:1	RO	0x0	reserved
0	W1C	0x0	frm_start frame start Write 1, self clear.

IEP_CONFIG_DONE
Address: Operational Base + offset (0x0018)
configuration done

Bit	Attr	Reset Value	Description
31:1	RO	0x0	reserved

0	RW	0x0	config_done configuration done Wait for frame start to update raw register configuration to really used registers.

IEP_FRM_CNT
Address: Operational Base + offset (0x001c)
frame counter

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	frm_cnt frame counter Self increase one after a frame operation is finished. Write arbitrary value to clear to zero.

IEP_VIR_IMG_WIDTH
Address: Operational Base + offset (0x0020)
Image virtual width

Bit	Attr	Reset Value	Description
31:16	RW	0x0140	dst_vir_image_width Destination virtual image width
15:0	RW	0x0140	src_vir_image_width Source virtual image width

IEP_IMG_SCL_FCT
Address: Operational Base + offset (0x0024)
scaling factor

Bit	Attr	Reset Value	Description

31:16	RW	0x2000	vrt_scl_fct
			Vertical scale factor
			up scaling:
			vrt_scl_fct=floor(src_image_height/dst_image_height);
			down scaling:
			vrt_scl_fct=ceiling((dst_image_height+1)/(src_image_height+1));

15:0	RW	0x2000	hrz_scl_fct
			Horizontal scale factor
			up scaling:
			hrz_scl_fct=floor(src_image_width/dst_image_width);
			down scaling:
			hrz_scl_fct=ceiling((dst_image_width+1)/(src_image_width+1));

IEP_SRC_IMG_SIZE
Address: Operational Base + offset (0x0028)
Source image width/height

Bit	Attr	Reset Value	Description
31:29	RO	0x0	reserved

28:16	RW	0x00f0	src_image_height source image height
15:13	RO	0x0	reserved
12:0	RW	0x0140	src_image_width source image width

IEP_DST_IMG_SIZE
Address: Operational Base + offset (0x002c)
Destination image width/height

Bit	Attr	Reset Value	Description
31:29	RO	0x0	reserved
28:16	RW	0x00f0	dst_image_height Destination image height
15:13	RO	0x0	reserved
12:0	RW	0x0140	dst_image_width Destination image width

IEP_DST_IMG_WIDTH_TILE0
Address: Operational Base + offset (0x0030)
Destination image tile0 width

Bit	Attr	Reset Value	Description
31:12	RO	0x0	reserved
11:0	RW	0x000	dst_image_width_tile0 Destination image tile0 width

IEP_DST_IMG_WIDTH_TILE1
Address: Operational Base + offset (0x0034)
Destination image tile1 width

Bit	Attr	Reset Value	Description
31:11	RO	0x0	reserved
10:0	RW	0x000	dst_image_width_tile1 Destination image tile1 width

IEP_DST_IMG_WIDTH_TILE2
Address: Operational Base + offset (0x0038)
Destination image tile2 width

Bit	Attr	Reset Value	Description
31:11	RO	0x0	reserved
10:0	RW	0x000	dst_image_width_tile2 Destination image tile2 width

IEP_DST_IMG_WIDTH_TILE3
Address: Operational Base + offset (0x003c)
Destination image tile3 width

Bit	Attr	Reset Value	Description
31:11	RO	0x0	reserved

10:0	RW	0x000	dst_image_width_tile3 Destination image tile3 width

IEP_ENH_YUV_CNFG_0
Address: Operational Base + offset (0x0040)
brightness,contrast,saturation adjustment

Bit	Attr	Reset Value	Description
31:25	RO	0x0	reserved
24:16	RW	0x000	sat_con YUV saturation and contrast adjustment saturation * contrast range from 0 to 1.992*1.992, and this value is saturation* contrast * 128
15:8	RW	0x00	contrast YUV contrast adjustment contrast value range from 0 to 1.992, and this value is contrast*128.
7:6	RO	0x0	reserved
5:0	RW	0x00	brightness YUV brightness adjustment range from -32 to 31 6'b000000:0; 6'b000001:1; …… 6'b011111:31; 6'b100000:-32; 6'b100001:-31; …… 6'b111110:-2; 6'b111111:-1;

IEP_ENH_YUV_CNFG_1
Address: Operational Base + offset (0x0044)
Hue configuration

Bit	Attr	Reset Value	Description
31:16	RO	0x0	reserved
15:8	RW	0x00	cos_hue the cos function value for hue adjustment sin function value range from 0.866 to 1 ,and this value is cos * 128 ,no sign bit
7:0	RW	0x00	sin_hue the sin function value for hue adjustment sin function value range from -0.5 to 0.5 ,and this value is sin * 128 ,and the high bit is sign bit

IEP_ENH_YUV_CNFG_2
Address: Operational Base + offset (0x0048)

color bar configuration

Bit	Attr	Reset Value	Description
31:26	RO	0x0	reserved
25:24	RW	0x0	video_mode video mode 2'b00:black screen 2'b01:blue screen 2'b10:color bars 2'b11:normal video
23:16	RW	0x00	color_bar_v color bar v value
15:8	RW	0x00	color_bar_u color bar u value
7:0	RW	0x00	color_bar_y color bar y value

IEP_ENH_RGB_CNFG

Address: Operational Base + offset (0x004c)

enhancement RGB configuration

Bit	Attr	Reset Value	Description
31:30	RW	0x0	luma_spat_sel 3D denoise luma spatial coefficient select
29:28	RW	0x0	luma_temp_sel 3D denoise luma temporal coefficient select
27:26	RW	0x0	chroma_spat_sel 3D denoise chroma spatial coefficient select
25:24	RW	0x0	chroma_temp_sel 3D denoise chroma temporal coefficient select
23:16	RW	0x00	enh_threshold enhancement threshold In denoise and detail enhancement operation, more than the threshold, considering as detail; but if less than the threshold, considering as noise, need to to be filtered.
15	RO	0x0	reserved

14:8	RW	0x00	enh_alpha enhancement alpha value 7'b0000000:0 7'b0000001:1/16 7'b0000010:2/16 …… 7'b0001111:15/16 7'b0010000:1 7'b0010001:1+1/16; 7'b0010010:1+2/16; 7'b0010011:1+3/16; …… 7'b0100000:2; …… 7'b0110000:3; …… 7'b1000000:4; …… 7'b1010000:5; …… 7'b1100000:6; other : reserved
7:2	RO	0x0	reserved
1:0	RW	0x0	enh_radius enhancement radius 2'b00:R=1 2'b01:R=2 2'b10:R=3 2'b11:R=4

IEP_ENH_C_COE
Address: Operational Base + offset (0x0050)
rgb color enhancement coefficient

Bit	Attr	Reset Value	Description
31:7	RO	0x0	reserved
6:5	RW	0x0	c_int_coe color enhancement integer coefficient
4:0	RW	0x00	c_frac_coe color enhancement fraction coefficient

IEP_SRC_ADDR_YRGB
Address: Operational Base + offset (0x0080)
Start address of source image(Y/RGB)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	src_image_yrgb_mst Source image data YRGB start address in Memory

IEP_SRC_ADDR_CBCR
Address: Operational Base + offset (0x0084)
Start address of source image(Cb/Cr)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	src_image_cbcr_mst Source image data CbCr start address in Memory

IEP_SRC_ADDR_CR
Address: Operational Base + offset (0x0088)
Start address of source image(Cr)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	src_image_cr_mst Source image data Cr start address in Memory

IEP_SRC_ADDR_Y1
Address: Operational Base + offset (0x008c)
Start address of source image(Y)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	src_image_y_mst Source image data Y start address in Memory

IEP_SRC_ADDR_CBCR1
Address: Operational Base + offset (0x0090)
Start address of source image(Cb/Cr)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	src_image_cbcr_mst Source image data CbCr start address in Memory

IEP_SRC_ADDR_CR1
Address: Operational Base + offset (0x0094)
Start address of source image(Cr)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	src_image_cr_mst Source image data Cr start address in Memory

IEP_SRC_ADDR_Y_ITEMP
Address: Operational Base + offset (0x0098)
Start address of source image(Y integer part)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	src_image_y_mst_itemp Interger part source image data Y start address in Memory

IEP_SRC_ADDR_CBCR_ITEMP
Address: Operational Base + offset (0x009c)
Start address of source image(CBCR integer part)

Bit	Attr	Reset Value	Description

31:0	RW	0x00000000	src_image_cbcr_mst_cbcr_itemp Interger part source image data CBCR start address in Memory

IEP_SRC_ADDR_CR_ITEMP
Address: Operational Base + offset (0x00a0)
Start address of source image(CR integer part)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	src_image_cr_mst_cr_itemp Interger part source image data CR start address in Memory

IEP_SRC_ADDR_Y_FTEMP
Address: Operational Base + offset (0x00a4)
Start address of source image(Y fraction part)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	src_image_y_mst_ftemp Fraction part source image data Y start address in Memory

IEP_SRC_ADDR_CBCR_FTEMP
Address: Operational Base + offset (0x00a8)
Start address of source image(CBCR fraction part)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	src_image_cbcr_mst_ftemp Fraction part source image data CBCR start address in Memory

IEP_SRC_ADDR_CR_FTEMP
Address: Operational Base + offset (0x00ac)
Start address of source image(CR fraction part)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	src_image_cr_mst_ftemp Fraction part source image data CR start address in Memory

IEP_DST_ADDR_YRGB
Address: Operational Base + offset (0x00b0)
Start address of destination image(Y/RGB)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	dst_image_yrgb_mst Destination image data YRGB start address in Memory

IEP_DST_ADDR_CBCR
Address: Operational Base + offset (0x00b4)
Start address of destination image(Cb/Cr)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	dst_image_cbcr_mst Destination image data CBCR start address in Memory

IEP_DST_ADDR_CR

Address: Operational Base + offset (0x00b8)
Start address of destination image(Cr)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	dst_image_cr_mst Destination image data CR start address in Memory

IEP_DST_ADDR_Y1
Address: Operational Base + offset (0x00bc)
Start address of destination image(Y)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	dst_image_y_mst Destination image data Y start address in Memory

IEP_DST_ADDR_CBCR1
Address: Operational Base + offset (0x00c0)
Start address of destination image(Cb/Cr)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	dst_image_cbcr_mst Destination image data CbCr start address in Memory

IEP_DST_ADDR_CR1
Address: Operational Base + offset (0x00c4)
Start address of destination image(Cr)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	dst_image_cr_mst Destination image data Cr start address in Memory

IEP_DST_ADDR_Y_ITEMP
Address: Operational Base + offset (0x00c8)
Start address of destination image(Y integer part)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	dst_image_y_mst_itemp Interger part destination image data Y start address in Memory

IEP_DST_ADDR_CBCR_ITEMP
Address: Operational Base + offset (0x00cc)
Start address of destination image(CBCR integer part)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	dst_image_cbcr_mst_itemp Int part destination image data CBCR start address in Memory

IEP_DST_ADDR_CR_ITEMP
Address: Operational Base + offset (0x00d0)
Start address of destination image(CR integer part)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	dst_image_cr_mst_itemp Interger part destination image data CR start address in Memory

IEP_DST_ADDR_Y_FTEMP
Address: Operational Base + offset (0x00d4)
Start address of destination image(Y fraction part)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	dst_image_y_mst_ftemp Fraction part destination image data Y start address in Memory

IEP_DST_ADDR_CBCR_FTEMP
Address: Operational Base + offset (0x00d8)
Start address of destination image(CBCR fraction part)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	dst_image_cbcr_mst_ftemp Fraction part destination image data CBCR start address in Mem

IEP_DST_ADDR_CR_FTEMP
Address: Operational Base + offset (0x00dc)
Start address of destination image(CR fraction part)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	dst_image_cr_mst_ftemp Fraction part destination image data CR start address

IEP_DIL_MTN_TAB0
Address: Operational Base + offset (0x00e0)
Deinterlace motion table0

Bit	Attr	Reset Value	Description
31	RO	0x0	reserved
30:24	RW	0x00	mtn_sub_tab3 motion sub table3
23	RO	0x0	reserved
22:16	RW	0x00	mtn_sub_tab2 motion sub table2
15	RO	0x0	reserved
14:8	RW	0x00	mtn_sub_tab1 motion sub table1
7	RO	0x0	reserved
6:0	RW	0x00	mtn_sub_tab0 motion sub table0

IEP_DIL_MTN_TAB1
Address: Operational Base + offset (0x00e4)
Deinterlace motion table1

Bit	Attr	Reset Value	Description
31	RO	0x0	reserved
30:24	RW	0x00	mtn_sub_tab3 motion sub table3

23	RO	0x0	reserved
22:16	RW	0x00	mtn_sub_tab2 motion sub table2
15	RO	0x0	reserved
14:8	RW	0x00	mtn_sub_tab1 motion sub table1
7	RO	0x0	reserved
6:0	RW	0x00	mtn_sub_tab0 motion sub table0

IEP_DIL_MTN_TAB2
Address: Operational Base + offset (0x00e8)
Deinterlace motion table2

Bit	Attr	Reset Value	Description
31	RO	0x0	reserved
30:24	RW	0x00	mtn_sub_tab3 motion sub table3
23	RO	0x0	reserved
22:16	RW	0x00	mtn_sub_tab2 motion sub table2
15	RO	0x0	reserved
14:8	RW	0x00	mtn_sub_tab1 motion sub table1
7	RO	0x0	reserved
6:0	RW	0x00	mtn_sub_tab0 motion sub table0

IEP_DIL_MTN_TAB3
Address: Operational Base + offset (0x00ec)
Deinterlace motion table3

Bit	Attr	Reset Value	Description
31	RO	0x0	reserved
30:24	RW	0x00	mtn_sub_tab3 motion sub table3
23	RO	0x0	reserved
22:16	RW	0x00	mtn_sub_tab2 motion sub table2
15	RO	0x0	reserved
14:8	RW	0x00	mtn_sub_tab1 motion sub table1
7	RO	0x0	reserved
6:0	RW	0x00	mtn_sub_tab0 motion sub table0

IEP_DIL_MTN_TAB4
Address: Operational Base + offset (0x00f0)

Deinterlace motion table4

Bit	Attr	Reset Value	Description
31	RO	0x0	reserved
30:24	RW	0x00	mtn_sub_tab3 motion sub table3
23	RO	0x0	reserved
22:16	RW	0x00	mtn_sub_tab2 motion sub table2
15	RO	0x0	reserved
14:8	RW	0x00	mtn_sub_tab1 motion sub table1
7	RO	0x0	reserved
6:0	RW	0x00	mtn_sub_tab0 motion sub table0

IEP_DIL_MTN_TAB5

Address: Operational Base + offset (0x00f4)

Deinterlace motion table5

Bit	Attr	Reset Value	Description
31	RO	0x0	reserved
30:24	RW	0x00	mtn_sub_tab3 motion sub table3
23	RO	0x0	reserved
22:16	RW	0x00	mtn_sub_tab2 motion sub table2
15	RO	0x0	reserved
14:8	RW	0x00	mtn_sub_tab1 motion sub table1
7	RO	0x0	reserved
6:0	RW	0x00	mtn_sub_tab0 motion sub table0

IEP_DIL_MTN_TAB6

Address: Operational Base + offset (0x00f8)

Deinterlace motion table6

Bit	Attr	Reset Value	Description
31	RO	0x0	reserved
30:24	RW	0x00	mtn_sub_tab3 motion sub table3
23	RO	0x0	reserved
22:16	RW	0x00	mtn_sub_tab2 motion sub table2
15	RO	0x0	reserved
14:8	RW	0x00	mtn_sub_tab1 motion sub table1
7	RO	0x0	reserved

6:0	RW	0x00	mtn_sub_tab0 motion sub table0

IEP_DIL_MTN_TAB7
Address: Operational Base + offset (0x00fc)
Deinterlace motion table7

Bit	Attr	Reset Value	Description
31	RO	0x0	reserved
30:24	RW	0x00	mtn_sub_tab3 motion sub table3
23	RO	0x0	reserved
22:16	RW	0x00	mtn_sub_tab2 motion sub table2
15	RO	0x0	reserved
14:8	RW	0x00	mtn_sub_tab1 motion sub table1
7	RO	0x0	reserved
6:0	RW	0x00	mtn_sub_tab0 motion sub table0

IEP_ENH_CG_TAB
Address: Operational Base + offset (0x0100)
contrast and gamma enhancement table

Bit	Attr	Reset Value	Description
31:24	RW	0x00	cg_tab_3 cg table 3 pixel value 3,7,11,15,……mapping
23:16	RW	0x00	cg_tab_2 cg table 2 pixel value 2,6,10,14,……mapping
15:8	RW	0x00	cg_tab_1 cg table 1 pixel value 1,5,9,13,……mapping
7:0	RW	0x00	cg_tab_0 cg table 0 256x8bit contrast & gamma mapping table pixel value 0,4,8,12,……mapping

IEP_ENH_DDE_COE0
Address: Operational Base + offset (0x0400)
denoise,detail and edge enhancement coefficient

Bit	Attr	Reset Value	Description
31:30	RO	0x0	reserved
29:24	RW	0x00	dde_coe_3 dde coefficient 3 coefficient number 3,7,11,15,……

23:22	RO	0x0	reserved
21:16	RW	0x00	dde_coe_2 dde coefficient 2 coefficient number 2,6,10,14,……
15:14	RO	0x0	reserved
13:8	RW	0x00	dde_coe_1 dde coefficient 1 coefficient number 1,5,9,13,……
7:6	RO	0x0	reserved
5:0	RW	0x00	dde_coe_0 dde coefficient 0 256x6bit cofficient for denoise and detail enhancement coefficient number 0,4,8,12,……

IEP_ENH_DDE_COE1
Address: Operational Base + offset (0x0500)
denoise,detail and edge enhancement coefficient

Bit	Attr	Reset Value	Description
31:30	RO	0x0	reserved
29:24	RW	0x00	dde_coe_3 dde coefficient 3 coefficient number 3,7,11,15,……
23:22	RO	0x0	reserved
21:16	RW	0x00	dde_coe_2 dde coefficient 3 coefficient number 2,6,10,14,……
15:14	RO	0x0	reserved
13:8	RW	0x00	dde_coe_1 dde coefficient 1 coefficient number 1,5,9,13,……
7:6	RO	0x0	reserved
5:0	RW	0x00	dde_coe_0 dde coefficient 1 81x6bit cofficient for denoise and detail enhancement coefficient number 0,4,8,12,……

IEP_MMU_DTE_ADDR
Address: Operational Base + offset (0x0800)
MMU current page table address

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	mmu_dte_addr page table address

IEP_MMU_STATUS
Address: Operational Base + offset (0x0804)
MMU status register

Bit	Attr	Reset Value	Description
31:11	RO	0x0	reserved
10:6	RW	0x00	mmu_page_fault_bus_id Index of master responsible for the last page fault
5	RW	0x0	mmu_page_fault_is_write The direction of access for last page fault: 1'b0: read 1'b1:write
4	RW	0x1	mmu_replay_buffer_empty The MMU replay buffer is empty.
3	RW	0x1	mmu_idle the MMU is idle when accesses are being translated and there are no unfinished translated access. The MMU_IDLE signal only reports idle when the MMU processor is idle and accesses are active on the external bus. Note: the MMU can be idle in page fault mode.
2	RW	0x0	mmu_stall_active MMU stall mode currently enabled. The mode is enabled by command.
1	RW	0x0	mmu_page_fault_active MMU page fault mode currently enabled.The mode is enabled by command
0	RW	0x0	mmu_paging_enabled mmu paging is enabled

IEP_MMU_CMD
Address: Operational Base + offset (0x0808)
MMU command register

Bit	Attr	Reset Value	Description
31:3	RO	0x0	reserved
2:0	RW	0x0	mmu_cmd 3'd0: MMU_ENABLE_PAGING. enable paging. 3'd1: MMU_DISABLE_PAGING. disable paging. 3'd2: MMU_ENABLE_STALL. turn on stall mode. 3'd3: MMU_DISABLE_STALL. turn off stall mode. 3'd4: MMU_ZAP_CACHE. zap the entire page table cache. 3'd5: MMU_PAGE_FAULT_DONE. leave page fault mode. 3'd6: MMU_FORCE_RESET. reset the mmu. The MMU_ENABLE_STALL command can always be issued. Other commands are ignored unless the MMU is idle or stalled.

IEP_MMU_PAGE_FAULT_ADDR
Address: Operational Base + offset (0x080c)
MMU logic address of last page fault

Bit	Attr	Reset Value	Description

31:0	RW	0x00000000	mmu_page_fault_addr address of last page fault

IEP_MMU_ZAP_ONE_LINE
Address: Operational Base + offset (0x0810)
MMU zap cache line register

Bit	Attr	Reset Value	Description
31:1	RO	0x0	reserved
0	RW	0x0	mmu_zap_one_line address to be invalidated from the page table cache.

IEP_MMU_INT_RAWSTAT
Address: Operational Base + offset (0x0814)
MMU raw interrupt status register

Bit	Attr	Reset Value	Description
31:2	RO	0x0	reserved
1	RW	0x0	read_bus_error read bus error
0	RW	0x0	page_fault page fault

IEP_MMU_INT_CLEAR
Address: Operational Base + offset (0x0818)
MMU interrupt clear register

Bit	Attr	Reset Value	Description
31:2	RO	0x0	reserved
1	RW	0x0	read_bus_error_clear read bus error interrupt clear. write 1 to this register can clear read bus error interrupt.
0	RW	0x0	page_fault_clear page fault interrupt clear, write 1 to this register can clear page fault interrupt.

IEP_MMU_INT_MASK
Address: Operational Base + offset (0x081c)
MMU interrupt mask register

Bit	Attr	Reset Value	Description
31:2	RO	0x0	reserved
1	RW	0x0	read_bus_error_int_en read bus error interrupt enable
0	RW	0x0	page_fault_int_en page fault interrupt enable

IEP_MMU_INT_STATUS
Address: Operational Base + offset (0x0820)
MMU interrupt status register

Bit	Attr	Reset Value	Description
31:2	RO	0x0	reserved
1	RW	0x0	read_bus_error read bus error interrupt
0	RW	0x0	page_fault page fault interrupt

IEP_MMU_AUTO_GATING
Address: Operational Base + offset (0x0824)
MMU clock auto gating register

Bit	Attr	Reset Value	Description
31:1	RO	0x0	reserved
0	RW	0x1	mmu_auto_gating mmu clock auto gating when it is 1, the mmu will auto gating itself

4.5 Application Notes

4.5.1 VOP path disabled configure flow

1. Keep IEP direct path disabled.
2. Configure all registers which are needed at any time.
3. Configure IEP_CONFIG_DONE.
4. Configure IEP_FRM_START.

4.5.2 VOP path enabled configure flow

1. Keep IEP direct path enabled.
2. Configure all IEP registers which are needed.
3. Configure VOP related registers which are needed.
4. Configure CONFIG_DONE register in VOP only.
5. Wait for frame start from VOP and IEP direct path.

4.5.3 VOP path turn on flow

1. Configure all IEP registers which are needed.
2. Configure VOP related registers which are needed.
3. Enable IEP direct path.
4. Enable VOP direct path.
5. Configure CONFIG_DONE register in VOP only.
6. Wait for frame start from VOP and IEP direct path.

4.5.4 VOP path turn off flow

1. Disable VOP direct path.
2. Disable IEP direct path, so IEP do not receive any other CONFIG_DONE and frame start from VOP immediately.
3. Configure CONFIG_DONE register in VOP.
4. Wait for frame start from VOP and IEP direct path, so VOP quit direct path mode completely.
5. Configure IEP registers which are needed at any time.
6. Configure IEP_CONFIG_DONE.
7. Configure IEP_FRM_START, IEP is working at write back mode now.

Chapter 5 Camera Interface (CIF)

5.1 Overview

The Camera Interface, receives the data from Camera or CCIR656 encoder, and transfers the data into system main memory by AXI bus.
The features of camera interface are as follow:

- Support YCbCr422 input
- Support Raw8bit input
- Support JPEG input
- Support YCbCr422/420 output
- Support UYVY/VYUY/YUYV/YVYU configurable
- Support up to 8192x8192 resolution source(Exception:width of BT656 data is limited to 1024)
- Support picture in picture
- Support arbitrary size window crop
- Support error/terminate interrupt and combined interrupt output
- Support clk/vsync/href polarity configurable
- Support one frame stop/ping-pong mode

5.2 Block Diagram

CIF comprises with:

- AHB Slave

Host configure the registers via the AHB Slave

- AXI Master

Transmit the data to chip memory via the AXI Master

- INTERFACE

Translate the input video data into the requisite data format

- CROP

Bypass or crop the source video data to a smaller size destination

- DMA

Control the operation of AXI Master

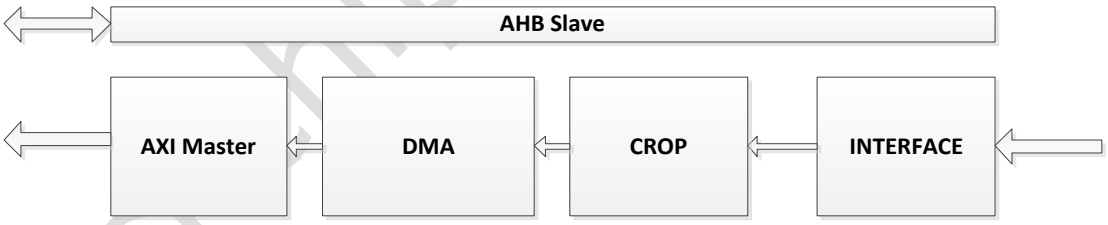


Fig. 5-1 CIF Block Diagram

5.3 Function Description

This chapter is used to illustrate the operational behavior of how CIF works. If YUV422 or ccir656 signal is received from external devices, CIF translate it into YUV422/420 data, and separate the data to Y and UV data, then store them to different memory via AXI bus separately. But if raw data is received, there are not any translations happened, the 8 data is considered as 16bit data and write directly to memory.

5.3.1 Support Vsync high active or low active

- Vsync Low active as below

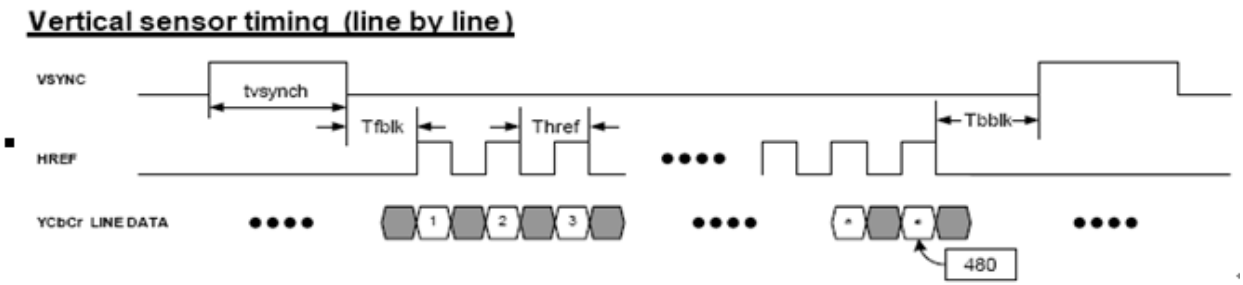


Fig. 21-2 Timing diagram for CIF when vsync low active

- Vsync High active

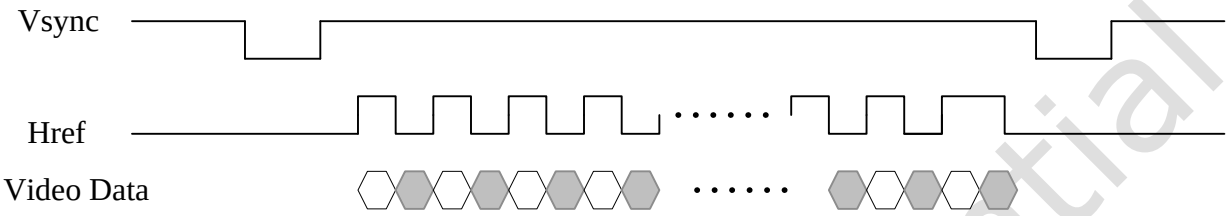


Fig. 21-3 Timing diagram for CIF when vsync high active

5.3.2 Support href high active or low active

- Href high active

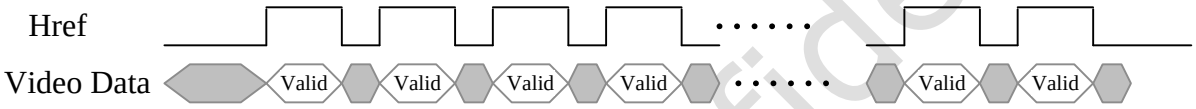


Fig. 21-4 Timing diagram for CIF when href high active

- Href Low active

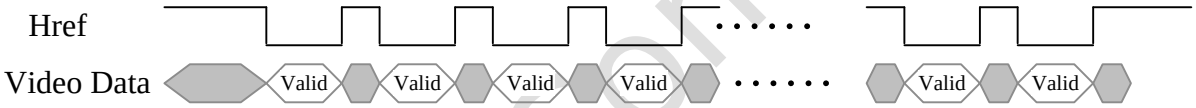


Fig. 21-5 Timing diagram for CIF when href low active

- Y first

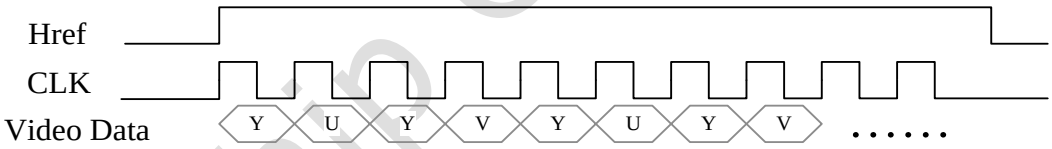


Fig. 21-6 Timing diagram for CIF when Y data first

- U first

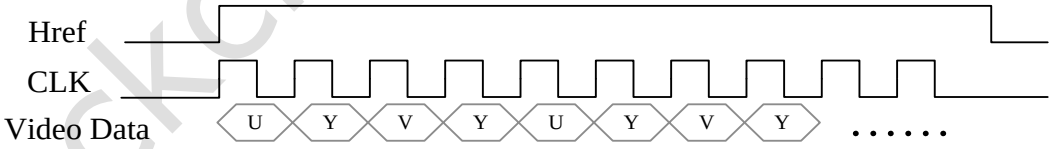


Fig. 21-7 Timing diagram for CIF when U data first

5.3.3 Support CCIR656(NTSC and PAL)

- ccir656 format

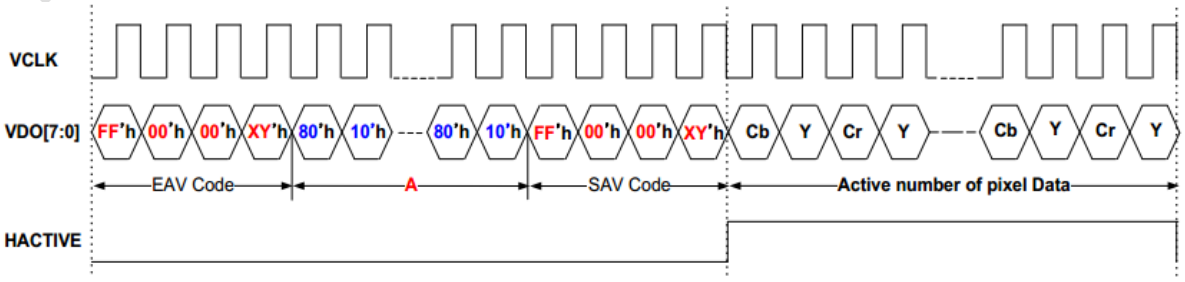


Fig. 21-8 ccir656 format

5.3.4 Support Raw data(8-bit) or JPEG

Pixel Data Timing Example

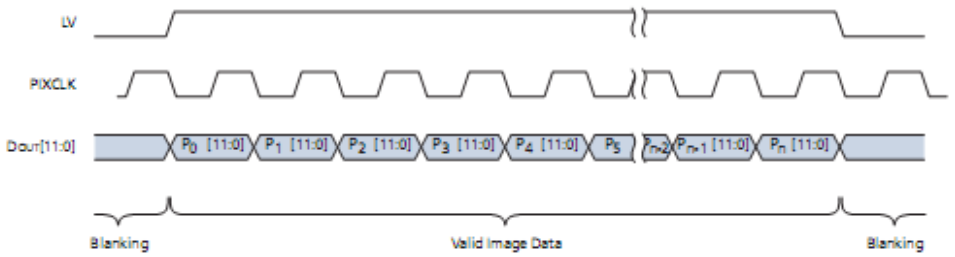


Fig. 21-9 Raw data or JPEG Timing

CIF module can work in two modes: one frame stop mode, ping-pong mode.

One frame stop mode

In this mode, configure the parameter WORK_MODE to one frame stop mode. After one frame captured, CIF will automatic stop. After capturing, the image Y, UV data will be stored at main memory location defined by CIF_FRM0_ADDR_Y, FRM0_ADDR_UV separately.

Ping-Pong mode

After one frame(F1) captured, CIF will start to capture the next frame(F2) automatically, and host must assign new address pointer of frame1 and clear the frame1 status, thus CIF will capture the third frame automatically(by new F1 address) without any stop and so on for the following frames. But if host did not update the frame buffer address, the CIF will cover the pre-frame data stored in the memory with the following frame data.

Storage

Difference between the YUV mode and raw mode is that in the YUV mode or ccir656 mode, data will be storage in the Y data buffer and UV data buffer; but in the raw or jpeg mode, RGB data will be storage in the same buffer. In addition, in the YUV mode, the width of Y, U or V data is a byte in memory; in Raw or JPEGE mode, the width is a half-word no matter the data source is 8 bit.

CROP

The parameter START_Y and START_X defines the coordinate of crop start point. And the frame size after cropping is following the value of SET_WIDTH and SET_HEIGHT.

The BCH Encoder is responsible for encoding data to be written into flash device. The max encoded length is 1133bytes, in which the data length is 1024bytes, system information is 4bytes, BCH code is 105bytes.

5.4 Register Description

5.4.1 Registers Summary

Name	Offset	Size	Reset Value	Description
CIF_CTRL	0x00000	W	0x00007000	CIF control
CIF_INTEN	0x00004	W	0x00000000	CIF interrupt status
CIF_INTSTAT	0x00008	W	0x00000000	CIF interrupt status
CIF_FOR	0x0000c	W	0x00000000	CIF format
CIF_FRM0_ADDR_Y	0x00014	W	0x00000000	CIF frame0 y address
CIF_FRM0_ADDR_UV	0x00018	W	0x00000000	CIF frame0 uv address
CIF_FRM1_ADDR_Y	0x0001c	W	0x00000000	CIF frame1 y address
CIF_FRM1_ADDR_UV	0x00020	W	0x00000000	CIF frame1 uv address
CIF_VIR_LINE_WIDTH	0x00024	W	0x00000000	CIF virtual line width
CIF_SET_SIZE	0x00028	W	0x01e002d0	CIF frame set size
CIF_CROP	0x00044	W	0x00000000	CIF crop start point
CIF_SCL_CTRL	0x00048	W	0x00000000	CIF scale control

Name	Offset	Size	Reset Value	Description
CIF_FIFO_ENTRY	0x00054	W	0x00000000	CIF FIFO entry
CIF_FRAME_STATUS	0x00060	W	0x00000000	CIF frame status
CIF_CUR_DST	0x00064	W	0x00000000	CIF current destination address
CIF_LAST_LINE	0x00068	W	0x00000000	CIF last frame line number
CIF_LAST_PIX	0x0006c	W	0x00000000	CIF last line pixel number

*Notes: Size : **B** - Byte (8 bits) access, **HW** - Half WORD (16 bits) access, **W** -WORD (32 bits) access*

5.4.2 Detail Register Description

CIF_CTRL

Address: Operational Base + offset (0x0000)

CIF control

Bit	Attr	Reset Value	Description
31:16	RO	0x0	reserved
15:12	RW	0x7	AXI_BURST_TYPE axi master burst type 0-15 : burst1~16
11:3	RO	0x0	reserved
2:1	RW	0x0	WORK_MODE Working Mode 00-one frame stop mode 01-ping-pong mode 02-line loop mode 03-reserved
0	RW	0x0	CAP_EN capture enable 0-disable 1-enable

CIF_INTEN

Address: Operational Base + offset (0x0004)

CIF interrupt enable

Bit	Attr	Reset Value	Description
31:7	RO	0x0	reserved
6	RW	0x0	BUS_ERR_EN bus error axi master or ahb slave response error 0-disable 1-enable
5:4	RO	0x0	reserved
3	RW	0x0	PIX_ERR_EN pixel err interrupt enable the pixel number of last line not equal to the set height 0-disable 1-enable

Bit	Attr	Reset Value	Description
2	RW	0x0	LINE_ERR_EN line err interrupt enable the line number of last frame not equal to the set height 0-disable 1-enable
1	RW	0x0	LINE_END_EN line end interrupt enable 0-disable 1-enable
0	RW	0x0	FRAME_END_EN frame end interrupt enable after dma transfer the frame data 0-disable 1-enable

CIF_INTSTAT

Address: Operational Base + offset (0x0008)

CIF interrupt status

Bit	Attr	Reset Value	Description
31:7	RO	0x0	reserved
6	W1C	0x0	BUS_ERR bus error axi master or ahb slave response error 0-no interrupt 1-interrupt
5:4	RO	0x0	reserved
3	W1C	0x0	PIX_ERR pixel err interrupt the pixel number of last line not equal to the set height 0-no interrupt 1-interrupt
2	W1C	0x0	LINE_ERR line err interrupt the line number of last frame not equal to the set height 0-no interrupt 1-interrupt
1	W1C	0x0	LINE_END line end interrupt enable 0-no interrupt 1-interrupt
0	W1C	0x0	FRAME_END frame end interrupt after dma transfer the frame data 0-no interrupt 1-interrupt

CIF_FOR

Address: Operational Base + offset (0x000c)

CIF format

Bit	Attr	Reset Value	Description
31:20	RO	0x0	reserved
19	RW	0x0	UV_STORE_ORDER UV storage order 0 - UVUV 1 - VUVU
18	RW	0x0	RAW_END raw data endian 0 - little end 1 - big end
17	RW	0x0	OUT_420_ORDER output 420 order 00 - UV in the even line 01 - UV in the odd line Note: The first line is even line(line 0).
16	RW	0x0	OUTPUT_420 output 420 or 422 0 - output is 422 1 - output is 420
15:13	RO	0x0	reserved
12:11	RW	0x0	RAW_WIDTH raw data width must be 2'b00.
10	RW	0x0	JPEG_MODE JPEG mode 0 - other mode 1 - mode1
9	RW	0x0	FIELD_ORDER ccir input order 0-odd field first 1-even field first
8	RW	0x0	IN_420_ORDER 420 input order 00 - UV in the even line 01 - UV in the odd line Note: The first line is even line(line 0).
7	RW	0x0	INPUT_420 input 420 or 422 0 - 422 1 - 420

Bit	Attr	Reset Value	Description
6:5	RW	0x0	YUV_IN_ORDER YUV input order 00 - UYVY 01 - YVYU 10 - VYUY 11 - YUYV
4:2	RW	0x0	INPUT_MODE input mode 000 - YUV 010 - PAL 011 - NTSC 100 - RAW 101 - JPEG 110 - MIPI Other - invalid
1	RW	0x0	HREF_POL href input polarity 0-high active 1-low active
0	RW	0x0	VSYNC_POL vsync input polarity 0-;ow active 1-high active

CIF_FRM0_ADDR_Y
Address: Operational Base + offset (0x0014)
CIF frame0 y address

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	FRM0_ADDR_Y frame0 y address

CIF_FRM0_ADDR_UV
Address: Operational Base + offset (0x0018)
CIF frame0 uv address

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	FRM0_ADDR_UV frame0 uv address

CIF_FRM1_ADDR_Y
Address: Operational Base + offset (0x001c)
CIF frame1 y address

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	FRM1_ADDR_Y frame1 y address

CIF_FRM1_ADDR_UV
Address: Operational Base + offset (0x0020)
CIF frame1 uv address

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	FRM1_ADDR_UV frame1 uv address

CIF_VIR_LINE_WIDTH
Address: Operational Base + offset (0x0024)
CIF virtual line width

Bit	Attr	Reset Value	Description
31:15	RO	0x0	reserved
14:0	RW	0x0000	VIR_LINE_WIDTH virtual line width

CIF_SET_SIZE
Address: Operational Base + offset (0x0028)
CIF frame set size

Bit	Attr	Reset Value	Description
31:29	RO	0x0	reserved
28:16	RW	0x01e0	SET_HEIGHT set height
15:13	RO	0x0	reserved
12:0	RW	0x02d0	SET_WIDTH set width

CIF_CROP
Address: Operational Base + offset (0x0044)
CIF crop start point

Bit	Attr	Reset Value	Description
31:29	RO	0x0	reserved
28:16	RW	0x0000	START_Y start y point
15:13	RO	0x0	reserved
12:0	RW	0x0000	START_X start x point

CIF_SCL_CTRL
Address: Operational Base + offset (0x0048)
CIF scale control

Bit	Attr	Reset Value	Description
31:6	RO	0x0	reserved
5	RW	0x0	RAW_16B_BP raw 16 bit bypass 0-no bypass 1-bypass

Bit	Attr	Reset Value	Description
4	RW	0x0	YUV_16B_BP YUV 16 bit bypass 0-no bypass 1-bypass
3:0	RO	0x0	reserved

CIF_FIFO_ENTRY
Address: Operational Base + offset (0x0054)
CIF FIFO entry

Bit	Attr	Reset Value	Description
31:15	RO	0x0	reserved
14:8	RW	0x00	UV_FIFO_ENTRY valid UV double word in FIFO write 0 clear
7	RO	0x0	reserved
6:0	RO	0x00	Y_FIFO_ENTRY valid Y double word in FIFO write 0 clear

CIF_FRAME_STATUS
Address: Operational Base + offset (0x0060)
CIF frame status

Bit	Attr	Reset Value	Description
31:16	RO	0x0000	FRAME_NUM complete frame number write 0 to clear
15:2	RO	0x0	reserved
1	RO	0x0	F1_STS frame 0 status 0- frame 1 not ready 1- frame 1 ready write 0 clear
0	RO	0x0	F0_STS frame 0 status 0- frame 0 not ready 1- frame 0 ready write 0 clear

CIF_CUR_DST
Address: Operational Base + offset (0x0064)
CIF current destination address

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	CUR_DST current destination address maybe not the current, because the clock synchronization.

CIF_LAST_LINE

Address: Operational Base + offset (0x0068)
CIF last frame line number

Bit	Attr	Reset Value	Description
31:14	RO	0x0	reserved
13:0	RO	0x0000	LAST_LINE_NUM line number of last frame

CIF_LAST_PIX
Address: Operational Base + offset (0x006c)
CIF last line pixel number

Bit	Attr	Reset Value	Description
31:15	RO	0x0	reserved
14:0	RO	0x0000	LAST_PIX_NUM pixel number of last line

5.5 Interface Description

Table 5-1 CIF Interface Description

Module Pin	Direction	Pad Name
cif_clkout	O	IO_CIFclkout
cif_clkin	I	IO_CIFclkin
cif_href	I	IO_CIFhref
cif_vsync	I	IO_CIFvsync
cif_data0	I	IO_CIFd0
cif_data1	I	IO_CIFd1
cif_data2	I	IO_CIFd2
cif_data3	I	IO_CIFd3
cif_data4	I	IO_CIFd4
cif_data5	I	IO_CIFd5
cif_data6	I	IO_CIFd6
cif_data7	I	IO_CIFd7

5.6 Application Notes

The most important configuration requirement of all operations is the CAP_EN bit must be set after all the mode selection is ready. The configuration order of the input/output data format, YUV order, the address, frame size/width, AXI burst length and other options do not need to care.
There are many debug registers to make it easy to read the internal operation information of CIF. The valid pixel number of scale result in FIFO can be known by read CIF_SCL_VALID_NUM. The line number of last frame and the pixel number of last line can be also known by read the CIF_LAST_LINE and CIF_LAST_PIX.

Chapter 6 Interconnect

6.1 Overview

The chip-level interconnect consists of one cpu_sys interconnect and peri_sys interconnects. It enables communication among the modules and subsystems in the device. The cpu_sys interconnect handles many types of data transfers, especially exchanges with system-on-chip (SoC)/external memories. It transfers data with a maximum width of 128 bits from the initiator to the target. It is a little-endian platform.
The peri_sys interconnect belongs to peripheral system which is responsible for peripheral devices control such as USB device, flash device, UART, SPI etc.

6.2 cpu_sys Interconnect

The cpu_sys interconnect block diagram is shown below:

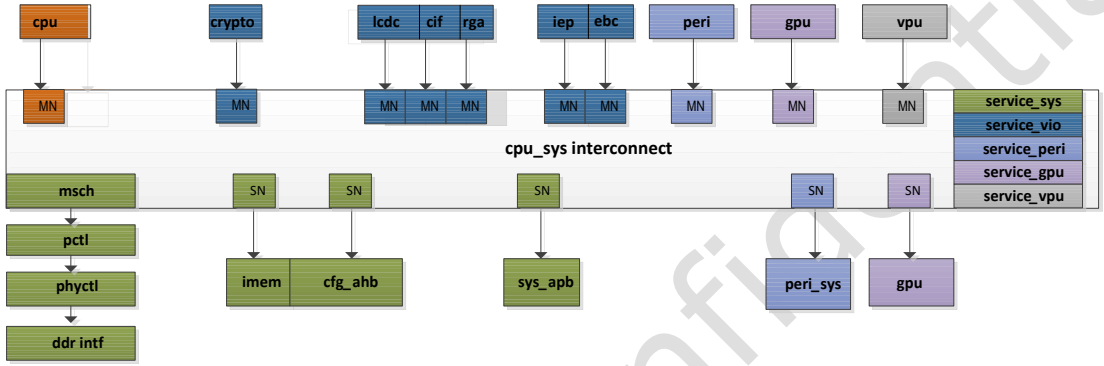


Fig. 6-1 cpu_sys Interconnect Block Diagram

6.2.1 Integration Description

The main interconnect is connected with all the related IPs of the system, the interface between the IP and interconnect is called as NIU (native interface unit). All the connected NIU are list as bellowing.

Table 6-1 Master NIU

Master NIU	Description
MN_cpu	Master interface of pd_core system
MN_crypto	Master interface of crypto in pd_cpu system
MN_vop	Master interface of VOP in pd_vio system
MN_ebc	Master interface of EBC in pd_vio system
MN_cif	Master interface of CIF in pd_vio system
MN_rga	Master interface of RGA in pd_vio system
MN_iep	Master interface of IEP in pd_vio system
MN_peri	Master interface of pd_peri system
MN_gpu	Master interface of GPU in pd_gpu system
MN_vpu	Master interface of Video codec in pd_vcodec system

Table 6-2 Slave NIU

Master NIU	Description
SN_msch	Memory scheduler of pd_cpu system
SN_imem	Internal memory of pd_cpu system
SN_cfg_ahb	AHB interface for ahb slave of pd_cpu system
SN_sys_apb	APB slave interface of pd_cpu system
SN_peri_sys	AXI slave interface of pd_peri system
SN_gpu_slv	GPU slave interface of pd_gpu system

6.2.2 Connectivity

Following figure lists the functional paths between the cpu_sys interconnect master NIUs and the slave NIUs.

	Slave										
Master	gpu_apb_slv	msch	peri_sys	sys_apb	imem	cfg_ahb	service_sys	service_peri	service_vio	service_gpu	service_vpu
CPU0	√	√	√	√	√	√	√	√	√	√	√
CRYPTO		√			√						
DMAC		√	√								
GPU		√									
LCDC0		√									
IEP		√									
RGA		√									
CIF		√									
EBC		√									
VPU		√									
PERI		√									

Fig. 6-2 Connectivity

6.2.3 Memory Scheduler

Memory scheduler is a special NIU of the interconnect, it mainly deal with the transaction inside the interconnect and convert it to the transaction which the ddr protocol controller can recognize.

Following table shows the software configurable setting for the memory scheduler when the system connected to different size of ddr device.

The DDRCONF[3:0] is a configurable register inside the interconnect.

R: indicates Row bits

B: indicates Bank bits

C: indicates Column bits

D: indicates Chip selects bits

Table 6-3 DDR Configuration

DDRCONF[3:0]	Description
0	C RRRD RRRR RRRR RRRR RBBB CCCC CCCC CCCC
1	C RRDR RRRR RRRR RRRR RBBB CCCC CCCC CCCC
2	C RDRR RRRR RRRR RRRR RBBB CCCC CCCC CCCC
3	C DRRR RRRR RRRR RRRR RBBB CCCC CCCC CCCC
4	R RDRR RRRR RRRR RRRR BBBC CCCC CCCC CCCC
5	R DRRR RRRR RRRR RRRR BBBC CCCC CCCC CCCC
6	D RRRR RRRR RRRR RRRR BBBC CCCC CCCC CCCC
7	C CRRR DRRR RRRR RRRR RRBB BCCC CCCC CCCC
8	C CRRD RRRR RRRR RRRR RRBB BCCC CCCC CCCC
9	C CRDR RRRR RRRR RRRR RRBB BCCC CCCC CCCC
10	C CDRR RRRR RRRR RRRR RRBB BCCC CCCC CCCC
11	C RRCD RRRR RRRR RRRR RRRB BCCC CCCC CCCC
12	C CRRD BBBR RRRR RRRR RRRR RCCC CCCC CCCC
13	C RRDB BBRR RRRR RRRR RRRR CCCC CCCC CCCC
14	C RRRR RRRR RRRR RRRR DBBB CCCC CCCC CCCC
15	C CRRR RRRR RRRR RRRR RDBB BCCC CCCC CCCC

'ddrconf' and other memory scheduler registers can be configured through a slave NIU called 'service_sys'. It is base address is 0x10128000. Refer to Register Description chapter for more detail.

6.2.4 QoS Management

The interconnect offers 4 modes of qos management:

- None, QoSGenerator is disabled, and priority information are stuck at 0.
- Fixed, QoSGenerator drives applies a fixed urgency to read transactions, and a (possibly different) urgency to write transactions.
- Limiter, QoSGenerator behaves as in fixed mode, but limits the traffic bandwidth coming from that socket, possibly stalling requests if the initiator attempts to exceed its budget.
- Regulator, QoSGenerator promotes are demotes hurry, depending the bandwidth obtained by the initiator is below or beyond a bandwidth budget. As transactions exceeding the bandwidth limit are sent (even though demoted), the regulator mode may be considered as a softer version of the limiter mode.

Limiter Behavior

When configured in bandwidth limiter, the unit uses a 23 bit counter to measure the average bandwidth. This counter has a 1/256 byte resolution and works as follows:

- Adds the number of byte rounded up to 16 (1 -> 16) and then multiplied by 256 to the current value, each time a request is sent.
- Subtracts the Bandwidth register value every cycle. If the Counter becomes negative, force it to 0.
- If the Counter value is greater than the Saturation register value multiplied by 16*256, any incoming request is stalled until this condition disappears. Note that the Counter cannot wrap-around because the maximum value it can reach is: SaturationMax*16*256 + BurstMax*256 = 1023*4K+4K*256 = 5116K or 223 = 8192K.

The following example will show the Counter behavior: 32 byte bursts, F=400MHz, BW=200MB/s, T=0.32us. The Bandwidth register will be set to 256*200/400 = 128, and the

Saturation register to 128*0.32*400/4096 = 4 (which corresponds to 64 bytes).

Regulator Behavior

When configured in bandwidth regulator, the unit uses a 23 bit counter to measure the average bandwidth. This counter has a 1/256 byte resolution and works as follows:

- Adds the number of byte rounded up to 16 and then multiplied by 256 to the current value, each time a response is received. If the result is greater than the Saturation register value multiplied by 16*256, saturation to this value is applied.
- Subtracts the Bandwidth register value every cycle. If the Counter becomes negative, force it to 0.
- If the Counter value is less than or equal to the Saturation register value multiplied by 16*256/2, the SocketMst Hurry signal will be set to the HurryHigh register, and HurryLow otherwise. Note that Urgency and Press will be also set to the same value.

The following example will show the Counter behavior: 1Kbyte bursts, F=500MHz, BW=2GB/s,T=2.048us. The Bandwidth register will be set to 256*2000/500 = 1024, and the Saturation register to 1024*2.048*500/4096 = 256 (which corresponds to 4 Kbytes).

QoS Generator Programming

Bandwidth: This log2(socket.wData/8)+8 bits register defines the bandwidth in 1/256th byte per cycle unit. This allows a 2 MByte/s resolution at 500MHz. When thebandwidth is given in MByte/s, the value of this register will

be equal to 256*BWMB/s / FMHz.

Saturation: This 10 bits register defines the number of byte used for bandwidth measurement. It is expressed in 16bytes unit (up to 16 Kbyte). Usually the integration window is given in us or in cycle: the value of this register will beequal to Bandwidh*Tus*FMHz/(256*16)or Bandwidh*Ncycle/(256*16).

Parts of the master NIUs have QoS control function through QoS registers. The base address of each master NIUs is shown below.

Table 6-4 QoS Register Base Address

Master NIU	Base Address
rga_qos	0x1012f000
iep_qos	0x1012f100
vop_qos	0x1012f180
cif_qos	0x1012f200
peri_qos	0x1012c000
gpu_qos	0x1012d000
vpu_qos	0x1012e000
cpu_qos	0x1012a000
crypto_qos	0x10128080
ebc_qos	0x1012f080

6.3 Peri AXI interconnect

6.3.1 Integration Description

PERI AXI interconnect is a 64bits data width AXI interconnect which mainly response for peripheral devices' data transaction.

This AXI interconnect has the following master and slave:

MASTER

1). DMAC

A 64bits axi master, mainly responses for data transfer between peripheral and DDR SDRAM.

2). AHB_MASTER

GPS/NANDC/USB OTG/USB Host2.0/TSP/SFC/GMAC, these masters are used to transfer data between DDR SDRAM and each of their own module.

SLAVE

1).AHB_SLAVE

All ahb slave devices in peri system for register access, also for data transfer such as SD/MMC,SDIO,eMMC, GPS,TSP. SFC, I2S_2ch,I2S_8ch, SPDIF, USB OTG,USB HOST2.0 and NandC.

2). APB_SLAVE

DMAC/PWM/WDT/GPIO/UART/I2C/SPI/TIMER/EFUSE/SARADC/SCR/GMAC, these apb slaves are used to register configuration.

6.3.2 Function description

PERI AXI interconnect is a 64bits data width axi interconnect which mainly response for

A Global Programmers View (GPV) module exists to configure some properties of the interconnect. The arbitration scheme of this interconnect is configurable through GPV.

The priority from high to low is as follow:

- CPU AXI interconnect
- DMAC
- USB OTG2.0/USB Host2.0
- PD8_AHBM/ GPS_AHBM

Customers can configure the Qos value through the GPV to change this priority. If you configure them to same priority, then the interconnect uses a Least Recently Used (LRU) algorithm

6.4 Register Description

6.4.1 Memory Scheduler Registers Summary

The base address of this register is 0x10128000.

Name	Offset	Size	Reset Value	Description
coreid	0x0000	W	0x21501602	core id of memory scheduler 0
revisionid	0x0004	W	0x0126f200	revisionid
ddrconf	0x0008	W	0x00000000	ddr configuration register
ddrtiming	0x000c	W	0x2475931c	ddr timing register
ddrmode	0x0010	W	0x00000000	ddr mode register
readlatency	0x0014	W	0x00000032	read latency register

*Notes:Size:**B**- Byte (8 bits) access, **HW**- Half WORD (16 bits) access, **W**-WORD (32 bits) access*

6.4.2 Memory Scheduler Detail Register Description

memory scheduler coreid

Address: Operational Base + offset (0x0000)

Bit	Attr	Reset Value	Description
31:0	RO	0x21501602	Core type id

memory scheduler revisionid

Address: Operational Base + offset (0x0004)

Bit	Attr	Reset Value	Description
31:0	RO	0x0126f200	msch revision id

memory scheduler ddrconf

Address: Operational Base + offset (0x0008)

Memory scheduler configuration register

Bit	Attr	Reset Value	Description
31:4	RO	0x0	reserved
3:0	RW	0x0	DdrConf select the ddr rank,row,bank,col sequence

memory scheduler ddrtiming

Address: Operational Base + offset (0x000c)

Memory scheduler timing register

Bit	Attr	Reset Value	Description
31	RW	0x0	BwRatio When set to zero, one DRAM clock cycle (two DDR transfers) is used to transfer each word of data. When set to one, two DRAM clock cycles (four DDR transfers) are used to transfer each word of data. This is applicable when half of a DRAM data bus width is used.
30:26	RW	0x07	WrToRd The minimum number of scheduler clock cycles between the last DRAM Write command and a Read command (WL x tCkD + tWTR). tCkD is the DRAM clock period.
25:21	RW	0x02	RdToWr The minimum number of scheduler clock cycles between the last DRAM Read command and a Write command (DDR2: 2 x tCkD, DDR3: (RL - WL + 2) x tCkD). tCkD is the DRAM clock period.
20:18	RW	0x2	BurstLen The DRAM burst duration on the DRAM data bus in scheduler clock cycles. Also equal to scheduler clock cycles between two DRAM commands (BL / 2 x tCkD). tCkD is the DRAM clock period.
17:12	RW	0x14	WrToMiss The minimum number of scheduler clock cycles between the last DRAM Write command and a new Read or Write command in another page of the same bank (WL x tCkD + tWR + tRP + tRCD). tCkD is the DRAM clock period.
11:6	RW	0x09	RdToMiss The minimum number of scheduler clock cycles between the last DRAM Read command and a new Read or Write command in another page of the same bank (tRTP + tRP + tRCD - BL x tCkD / 2). tCkD is the DRAM clock period.
5:0	RW	0x16	ActToAct The minimum number of scheduler clock cycles between two consecutive DRAM Activate commands on the same bank (tRC/ tCkG). tCkG is the clock period of the SoC DRAM scheduler.

memory scheduler ddrmode

Address: Operational Base + offset (0x0010)

Memory scheduler mode register

Bit	Attr	Reset Value	Description
31:2	RO	0x0	reserved
1	RW	0x0	BwRatioExtended When set to 1, support for 4x Bwratio.

Bit	Attr	Reset Value	Description
0	RW	0x0	AutoPrecharge When set to one, pages are automatically closed after each access, when set to zero, pages are left opened until an access in a different page occurs

memory scheduler readlatency

Address: Operational Base + offset (0x0014)

Memory scheduler read latency register

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:0	RW	0x32	ReadLatency The DRAM type-specific number of cycles from a scheduler request to a protocol controller response. This is a fixed value depending on the type of DRAM memory. <See SoC-specific memory controller documentation>.

memory scheduler activate

Address: Operational Base + offset (0x0038)

Memory scheduler activate register

Bit	Attr	Reset Value	Description
31:11	RO	0x0	reserved
10	RW	0x1	FawBank The number of Banks of a given device involved in the FAW period. Set to zero for 2-bank memories (WideIO). Set to one for memories with 4 banks or more (DDR).
9:4	RW	0x00	Faw The number of cycles for the four bank activate (FAW) period (tFAW).
3:0	RW	0x0	Rrd 'The number of cycles between two consecutive Activate commands on different Banks of the same device (tRRD).

memory scheduler devtodev

Address: Operational Base + offset (0x003c)

Memory scheduler devtodev register

Bit	Attr	Reset Value	Description
31:6	RO	0x0	reserved

Bit	Attr	Reset Value	Description
5:4	RW	0x0	BusWrToRd Field0001 Abstract The number of cycles between the last write data to a device and the first read data of another device of a memory array with multiple ranks (2 x tCkD). tCkD is the DRAM clock period.
3:2	RW	0x0	BusRdToWr Field0000 Abstract The number of cycles between the last read data of a device and the first write data to another device of a memory array with multiple ranks (2 x tCkD). tCkD is the DRAM clock period.
1:0	RW	0x0	BusRdToRd The number of cycles between the last read data of a device and the first read data of another device of a memory array with multiple ranks (tCkD). tCkD is the DRAM clock period.

6.4.3 cpu_sys QoS Registers Summary

Name	Offset	Size	Reset Value	Description
QOS_Id_CoreId	0x0000	W	0x0d867004	Core ID register
QOS_Id_RevisionId	0x0004	W	0x0001aa00	Revision ID register
QOS_Priority	0x0008	W	0x80000005	Priority register
QOS_Mode	0x000c	W	0x00000003	Mode register
QOS_Bandwidth	0x0010	W	0x0000018a	Bandwidth register
QOS_Saturation	0x0014	W	0x00000040	Saturation register
QOS_ExtControl	0x0018	W	0x00000000	External inputs control

*Notes:Size:**B**- Byte (8 bits) access, **HW**- Half WORD (16 bits) access, **W**-WORD (32 bits) access*

6.4.4 cpu_sys QoS Detail Register Description

QOS_Id_CoreId

Address: Operational Base + offset (0x0000)

Core ID register

Bit	Attr	Reset Value	Description
31:8	RO	0x0d8670	CoreChecksum Field containing a checksum of the parameters of the IP.
7:0	RO	0x04	CoreTypeId Field identifying the type of IP.

QOS_Id_RevisionId

Address: Operational Base + offset (0x0004)

Revision ID register

Bit	Attr	Reset Value	Description
31:0	RO	0x0001aa00	RevisionId Constant.

QOS_Priority

Address: Operational Base + offset (0x0008)

Priority register

Bit	Attr	Reset Value	Description
31	RO	0x1	Mark Backward compatibility marker when 0.
30:10	RO	0x0	reserved
9:8	RW	0x1	P1 In Programmable or Bandwidth Limiter mode, the priority level for read transactions. In Bandwidth regulator mode, the priority level when the used throughput is below the threshold. In Bandwidth Regulator mode, P1 should have a value equal or greater than P0.
7:2	RO	0x0	reserved
1:0	RW	0x1	P0 In Programmable or Bandwidth Limiter mode, the priority level for write transactions. In Bandwidth Regulator mode, the priority level when the used throughput is above the threshold. In Bandwidth Regulator mode, P0 should have a value equal or lower than P1.

QOS_Mode

Address: Operational Base + offset (0x000c)

Mode register

Bit	Attr	Reset Value	Description
31:2	RO	0x0	reserved

Bit	Attr	Reset Value	Description
1:0	RW	0x3	Mode 0 = Programmable mode: a programmed priority is assigned to each read or write, 1 = Bandwidth Limiter Mode: a hard limit restricts throughput, 2 = Bypass mode: (<See SoC-specific QoS generator documentation>), 3 = Bandwidth Regulator mode: priority decreases when throughput exceeds a threshold.

QOS_Bandwidth

Address: Operational Base + offset (0x0010)

Bandwidth register

Bit	Attr	Reset Value	Description
31:12	RO	0x0	reserved
11:0	RW	0x18a	Bandwidth In Bandwidth Limiter or Bandwidth Regulator mode, the bandwidth threshold in units of 1/256th bytes per cycle. For example, 80 MBps on a 250 MHz interface is value 0x0052. The valid bits may be different for different master NIU.

QOS_Saturation

Address: Operational Base + offset (0x0014)

Saturation register

Bit	Attr	Reset Value	Description
31:10	RO	0x0	reserved
9:0	RW	0x040	Saturation In Bandwidth Limiter or Bandwidth Regulator mode, the maximum data count value, in units of 16 bytes. This determines the window of time over which bandwidth is measured. For example, to measure bandwidth within a 1000 cycle window on a 64-bit interface is value 0x1F4.

QOS_ExtControl

Address: Operational Base + offset (0x0018)

External inputs control

Bit	Attr	Reset Value	Description
31:3	RO	0x0	reserved
2	RW	0x0	IntClkEn n/a
1	RW	0x0	ExtThrEn n/a
0	RW	0x0	SocketQosEn n/a

6.4.5 Peri QoS Registers Summary

Name	Offset	Size	Reset Value	Description
PERI_RQos_M0	0x42100	4bits	0x0003	PERI_AXI Port Read channel QoS value.
PERI_WQos_M0	0x42104	4bits	0x0003	PERI_AXI Port Write channel quality value.
PERI_RQos_M1	0x43100	4bits	0x0002	DMAC1 Port Read channel QoS value.
PERI_WQos_M1	0x43104	4bits	0x0002	DMAC1 Port Write channel quality value.
PERI_RQos_M2	0x44100	4bits	0x0001	USBM Port Read channel QoS value.
PERI_WQos_M2	0x44104	4bits	0x0001	USBM Port Write channel quality value.
PERI_RQos_M3	0x45100	4bits	0x0000	PD8_AHBM Port Read channel QoS value.
PERI_WQos_M3	0x45104	4bits	0x0000	PD8_AHBM Port Write channel quality value.
PERI_RQos_M4	0x46100	4bits	0x0000	GPS_AHBM Port Read channel QoS value.
PERI_WQos_M4	0x46104	4bits	0x0000	GPS_AHBM Port Write channel quality value.

*Notes:Size:**B**- Byte (8 bits) access, **HW**- Half WORD (16 bits) access, **W**-WORD (32 bits) access*

6.4.6 Peri QoS Detail Register Description

PERI_RQos_M0

Address: Operational Base+0x42100

PERI_AXI Port Read Qos register

Bit	Attr	Reset Value	Description
3:0	RW	0x1	Read channel QoSvalue.Higher value indicates higher priority.

PERI_WQos_M0

Address: Operational Base+0x42104

PERI_AXI Port Write Qos register

Bit	Attr	Reset Value	Description
3:0	RW	0x1	Write channel QoSvalue.Higher value indicates higher priority.

PERI_RQos_M1

Address: Operational Base+0x43100

DMAC1 Port Read Qos register

Bit	Attr	Reset Value	Description
3:0	RW	0x0	Read channel QoSvalue.Higher value indicates higher priority.

PERI_WQos_M1

Address: Operational Base+0x43104

DMAC1 Port Write Qos register

Bit	Attr	Reset Value	Description
3:0	RW	0x0	Write channel QoSvalue.Higher value indicates higher priority.

PERI_RQos_M2

Address: Operational Base+0x44100

USBM Port Read Qos register

Bit	Attr	Reset Value	Description

3:0	RW	0x0	Read channel QoSvalue.Higher value indicates higher priority.

PERI_WQos_M2

Address: Operational Base+0x44104

USBM Port Write Qos register

Bit	Attr	Reset Value	Description
3:0	RW	0x0	Write channel QoSvalue.Higher value indicates higher priority.

PERI_RQos_M3

Address: Operational Base +0x45100

PD8_AHBM Port Read Qos register

Bit	Attr	Reset Value	Description
3:0	RW	0x0	Read channel QoSvalue.Higher value indicates higher priority.

PERI_WQos_M3

Address: Operational Base +0x45104

PD8_AHBM Port Write Qos register

Bit	Attr	Reset Value	Description
3:0	RW	0x0	Write channel QoSvalue.Higher value indicates higher priority.

PERI_RQos_M4

Address: Operational Base +0x46100

PD8_AHBM Port Read Qos register

Bit	Attr	Reset Value	Description

3:0	RW	0x0	Read channel QoSvalue.Higher value indicates higher priority.

PERI_WQos_M4

Address: Operational Base +0x46104

PD8_AHBM Port Write Qos register

Bit	Attr	Reset Value	Description
3:0	RW	0x0	Write channel QoSvalue.Higher value indicates higher priority.

6.5 Application Notes

6.5.1 QoS setting

The VOP read channel have the external QoS control. After reset each master port both have priority setting as 1. It's recommended that field 0 of QoS. ExtControl set to 1 to enable the external QoS control. And priority setting of each master kept at 1.

6.5.2 Idle request

The main interconnect supports flushing the ongoing transaction when the software needed to do so.

If the GPU power domain need to disconnect from the main interconnect, Idle request has to be sent to GPU NIU, the NIU will respond a 'ack', and when it's ready to be disconnect, one Idle signal will be send out . Then, if GPU still have transaction to be sent to the memory scheduler, it will be stalled by the NIU.

If the GPU system power domain is disconnected as the above flow, then CPU want to access to the GPU system, it will response error to CPU.

The sequence is like following figure shows:

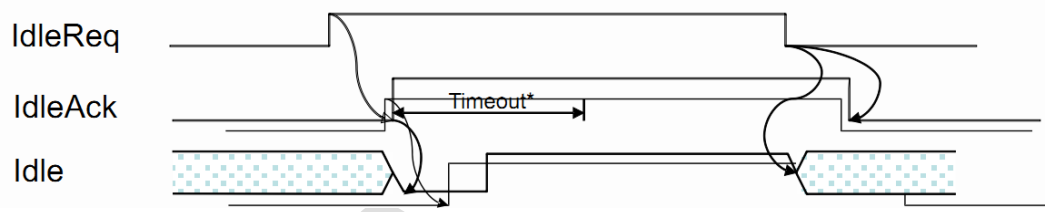


Fig. 6-3 Connectivity

The idle request is set by GRF register.

Chapter 7 DMC (Dynamic Memory Interface)

7.1 Overview

The DMC includes two section: dynamic sdram protocol controller(PCTL) and phy controller.

The PCTL SoC application bus interface supports a lowest-latency native application interface (NIF). To maximize data transfer efficiency, NIF commands transfer data without flow control. To simplify command processing, the NIF accepts addresses in rank, bank, row, column format.

The DDR PHY provides control features to ease the customer implementation of digitally controlled features of the PHY such as initialization, DQS gate training, and programmable configuration controls. The DDR PHY has built-in self test features to provide support for production testing of the compatible PHY. It also provides a DFI 2.1 interface to the PHY.

The DMC supports the following features:

- Complete, integrated DDR3, LPDDR2 solution
- DFI 2.1 interface compatibility
- Up to 800 Mbps in 1:1 frequency ratio, using a 400MHz controller clock and 400MHz memory clock.
- Support for x16, x32 DDR3 memories, for a total memory data path width of 32 bits
- Support for x32 LPDDR2 memories, for a total memory data path width of 32 bits
- Up to 2 memory ranks; devices within a rank tie to a common chip select
- Up to 8 open memory banks, maximum of eight per rank
- Per-NIF transaction controllable bank management policies: open-page, close-page
- Low area, low power architecture with minimal buffering on the data, avoiding duplication of storage resources within the system
- PCTL NIF slave interface facilitates easy integration with an external scheduler or standard on-chip buses
- Efficient DDR protocol implementation with in-order column (Read and Write) commands and out-of-order Activate and Precharge commands
- Three clock cycles best case command latency (best case is when a command is to an open page and the shift array in the PCTL is empty)
- 1T or 2T memory command timing
- Automatic power-down and self-refresh entry and exit
- Software and hardware driven self-refresh entry and exit
- Programmable memory initialization
- Partial population of memories, where not all DDR byte lanes are populated with memory chips
- Programmable per rank memory ODT (On-Die Termination) support for reads and writes
- APB interface for controller software-accessible registers
- Programmable data training interface:

  Assists in training of the data eye of the memory channel

  Provides a method for testing large sections of memory
- Automatic DQS gate training
- At-speed built-in-self-test (BIST) loopback testing on both the address and data channels for DDR PHYs
- PHY control and configuration registers
- Optional, additional JTAG interface to configure registers

7.2 Block Diagram

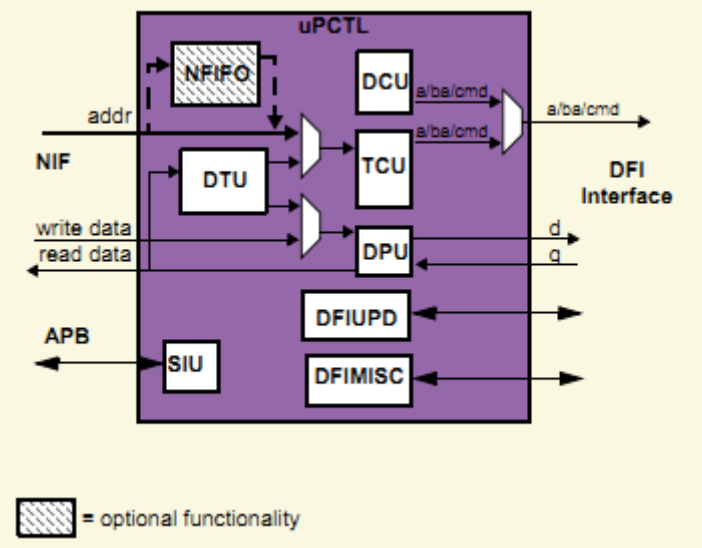


Fig. 7-1 Protocol controller architecture

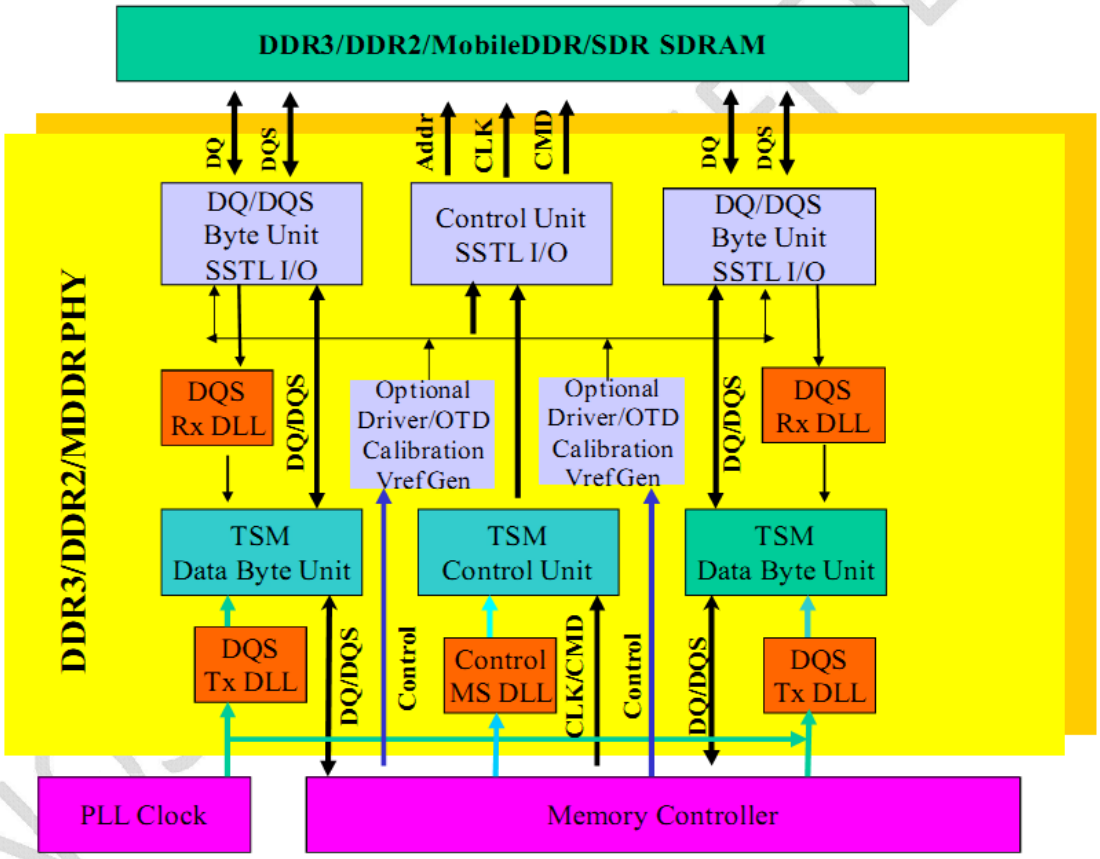


Fig. 7-2 PHY controller architecture

7.3 Function Description

7.3.1 Protocol controller(PCTL)

PCTL operations are defined in terms of the current state of the Operational State Machine. Software can move PCTL in any of the operational states by issuing commands via the SCTL register. Transitions from one operational state to the other occur pass through a "transitional" state. Transitional states are exited automatically by the PCTL after all the necessary actions required to change operational state have been completed. The current operational state of PCTL is reported by the STAT register and is also available from the p_ctl_stat output.

PCTL supports the following operational states:

- Init_mem - This state is the default state entered after reset. All writable registers can be programmed. While in this state software can program PCTL and initialize the PHY and the memories. The memories are not refreshed and data that has previously been written to the memories may be lost as a result. The Init_mem state

is also used when it is desirable to stop any automatic PCTL function that directly affects the memories, like Power Down and Refresh, or when a software reset of the memory subsystem has to be executed.

- Config - This state is used to suspend temporarily the normal NIF traffic and allow software to reprogram PCTL and memories if necessary, while still keeping active the periodic generation of Refresh cycles to the memories. Power Down entry and exit sequences are possible while in Config state.
- Access - This is the operational state where NIF transactions are accepted by the PCTL and converted into memory read and writes. None of the registers can be programmed except SCFG, SCTL, ECCCLR and DTU* registers.
- Low_power - Memories are in self refresh mode. The PCTL does not generate refresh cycles while in this state.

Access and Low_power states can also be entered and exited by the hardware low power signals (c_*). In case of conflicting software and hardware low-power commands, the resulting operational state taken by the controller can be either one of the two conflicting requests.

Following figure illustrates the operational and transitional states.

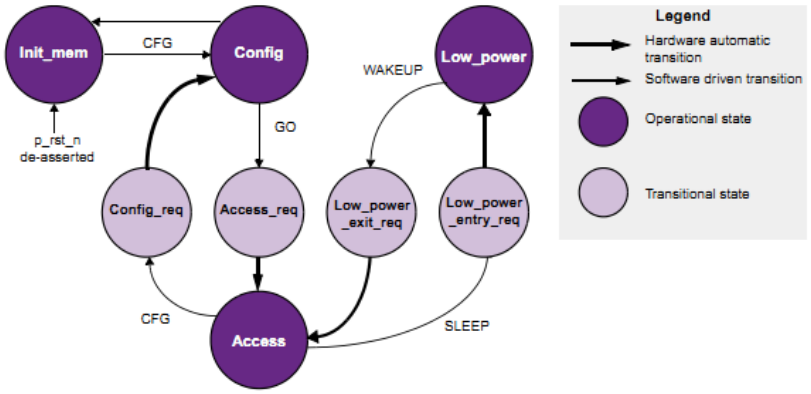


Fig. 7-3 Protocol controller architecture

The controller clock is the same clock driving the memory controller and will be the same frequency as the SDRAM clock (ck). The configuration clock can run at a frequency equal to or less than the controller clock. The configuration clock drives all non-DDR timing logic, such as configuration registers, PHY initialization, output impedance, and so on.

7.3.2 DDR PHY

DDR PHY provides turn key physical interface solutions for chip requiring access to DDR3 SDRAM device. It is optimized for low power and high speed (up to 800Mbps for DDR3 and LPDDR2) applications with robust timing and small silicon area. It supports DDR3 and LPDDR2 SDRAM components in the market. The PHY components contain DDR specialized functional and utility SSTL I/Os up to 800MHz, critical timing synchronization module (TSM) and a low power/jitter DLLs with programmable fine-grain control for any SDRAM interface. DDR PHY uses a DFI digital interface to connect the memory controller. All interface timing is in 1X SDR clock domain. The controller to PHY interface is running at single data rate (SDR) therefore read/write bus is double width. DDR muxing is done in the PHY block together with all related per-byte lane timing adjustment. The interface is fairly generic and support high performance input and output data flow gearing toward 100Mbps to 800Mbps DDR3 and LPDDR2 SDRAM speed in wide range.

With configurable timing and driving strength and ODT parameters to interface to the wide variety of SDRAMs, the PHY is very flexible with advanced command capability to increase SDRAM operation efficiency.

7.4 Register Description

7.4.1 Registers Summary

Name	Offset	Size	Reset Value	Description
DDR_PCTL_SCFG	0x0000	W	0x00000300	State Configuration Register
DDR_PCTL_SCTL	0x0004	W	0x00000000	Operational State Control Register
DDR_PCTL_STAT	0x0008	W	0x00000000	Operational State Status Register

Name	Offset	Size	Reset Value	Description
DDR_PCTL_INTRSTAT	0x000c	W	0x00000000	Interrupt Status Register
DDR_PCTL_MCMD	0x0040	W	0x00100000	Memory Command Register
DDR_PCTL_POWCTL	0x0044	W	0x00000000	Power Up Control Register
DDR_PCTL_POWSTAT	0x0048	W	0x00000000	Power Up Status Register
DDR_PCTL_CMDTSTAT	0x004c	W	0x00000000	Command Timers Status Register
DDR_PCTL_CMDTSTATEN	0x0050	W	0x00000000	Command Timers Status Enable Register
DDR_PCTL_MRRCFG0	0x0060	W	0x00000000	Mode Register Read Configuration 0
DDR_PCTL_MRRSTAT0	0x0064	W	0x00000000	Mode Register Read Status 0 Register
DDR_PCTL_MRRSTAT1	0x0068	W	0x00000000	Mode Register Read Status 0 Register
DDR_PCTL_MCFG1	0x007c	W	0x00000000	Memory Configuration 1 Register
DDR_PCTL_MCFG	0x0080	W	0x00040020	Memory Configuration Register
DDR_PCTL_PPCFG	0x0084	W	0x00000000	Partially Populated Memories Configuration Register
DDR_PCTL_MSTAT	0x0088	W	0x00000000	Memory Status Register
DDR_PCTL_LPDDR2ZQCFG	0x008c	W	0xab0a560a	LPDDR2 ZQ Configuration Register
DDR_PCTL_DTUPDES	0x0094	W	0x00000000	DTU Status Register
DDR_PCTL_DTUNA	0x0098	W	0x00000000	DTU Number of Addresses Created Register
DDR_PCTL_DTUNE	0x009c	W	0x00000000	DTU Number of Errors Register
DDR_PCTL_DTUPRD0	0x00a0	W	0x00000000	DTU Parallel Read 0 Register
DDR_PCTL_DTUPRD1	0x00a4	W	0x00000000	DTU Parallel Read 1 Register
DDR_PCTL_DTUPRD2	0x00a8	W	0x00000000	DTU Parallel Read 2 Register
DDR_PCTL_DTUPRD3	0x00ac	W	0x00000000	DTU Parallel Read 3 Register
DDR_PCTL_DTUAWDT	0x00b0	W	0x00000290	DTU Address Width Register
DDR_PCTL_TOGCNT1U	0x00c0	W	0x00000064	Toggle Counter 1us Register
DDR_PCTL_TINIT	0x00c4	W	0x000000c8	t_init Timing Register
DDR_PCTL_TRSTH	0x00c8	W	0x00000000	t_rsth Timing Register
DDR_PCTL_TOGCNT100N	0x00cc	W	0x00000001	Toggle Counter 100ns
DDR_PCTL_TREFI	0x00d0	W	0x00000001	t_refi Timing Register
DDR_PCTL_TMRD	0x00d4	W	0x00000001	t_mrd Timing Register
DDR_PCTL_TRFC	0x00d8	W	0x00000001	
DDR_PCTL_TRP	0x00dc	W	0x00010006	t_trp Timing Register
DDR_PCTL_TRTW	0x00e0	W	0x00000002	t_rtw Timing Register
DDR_PCTL_TAL	0x00e4	W	0x00000000	AL Register
DDR_PCTL_TCL	0x00e8	W	0x00000004	CL Timing Register
DDR_PCTL_TCWL	0x00ec	W	0x00000003	CWL Timing Register
DDR_PCTL_TRAS	0x00f0	W	0x00000010	t_ras Timing Register
DDR_PCTL_TRC	0x00f4	W	0x00000016	t_rc Timing Register

Name	Offset	Size	Reset Value	Description
DDR_PCTL_TRCD	0x00f8	W	0x00000006	t_rcd Timing Register
DDR_PCTL_TRRD	0x00fc	W	0x00000004	t_rrd Timing Register
DDR_PCTL_TRTP	0x0100	W	0x00000003	t_rtp Timing Register
DDR_PCTL_TWR	0x0104	W	0x00000006	t_wr Register
DDR_PCTL_TWTR	0x0108	W	0x00000004	t_wtr Timing Register
DDR_PCTL_TEXSR	0x010c	W	0x00000001	t_exsr Timing Register
DDR_PCTL_TXP	0x0110	W	0x00000001	t_xp Timing Register
DDR_PCTL_TXPDLL	0x0114	W	0x00000000	t_xpdll Timing Register
DDR_PCTL_TZQCS	0x0118	W	0x00000000	t_zqcs Timing Register
DDR_PCTL_TZQCSI	0x011c	W	0x00000000	t_zqcsi Timing Register
DDR_PCTL_TDQS	0x0120	W	0x00000001	t_dqs Timing Register
DDR_PCTL_TCKSRE	0x0124	W	0x00000000	t_cksre Timing Register
DDR_PCTL_TCKSRX	0x0128	W	0x00000000	t_cksrx Timing Register
DDR_PCTL_TCKE	0x012c	W	0x00000003	t_cke Timing Register
DDR_PCTL_TMOD	0x0130	W	0x00000000	t_mod Timing Register
DDR_PCTL_TRSTL	0x0134	W	0x00000000	Reset Low Timing Register
DDR_PCTL_TZQCL	0x0138	W	0x00000000	t_zqcl Timing Register
DDR_PCTL_TMRR	0x013c	W	0x00000002	t_mrr Timing Register
DDR_PCTL_TCKESR	0x0140	W	0x00000004	t_ckesr Timing Register
DDR_PCTL_TDPD	0x0144	W	0x00000000	t_dpd Timing Register
DDR_PCTL_DTUWACTL	0x0200	W	0x00000000	DTU Write Address Control
DDR_PCTL_DTURACTL	0x0204	W	0x00000000	DTU Read Address Control Register
DDR_PCTL_DTUCFG	0x0208	W	0x00000000	DTU Configuration Control Register
DDR_PCTL_DTUECTL	0x020c	W	0x00000000	DTU Execute Control Register
DDR_PCTL_DTUWD0	0x0210	W	0x00000000	DTU Write Data #0 Register
DDR_PCTL_DTUWD1	0x0214	W	0x00000000	DTU Write Data #1 Register
DDR_PCTL_DTUWD2	0x0218	W	0x00000000	DTU Write Data #2 Register
DDR_PCTL_DTUWD3	0x021c	W	0x00000000	DTU Write Data #3 Register
DDR_PCTL_DTUWDM	0x0220	W	0x00000000	DTU Write Data Mask Register
DDR_PCTL_DTURD0	0x0224	W	0x00000000	DTU Read Data #0 Register
DDR_PCTL_DTURD1	0x0228	W	0x00000000	DTU Read Data #1 Register
DDR_PCTL_DTURD2	0x022c	W	0x00000000	DTU Read Data #2 Register
DDR_PCTL_DTURD3	0x0230	W	0x00000000	DTU Read Data #3 Register
DDR_PCTL_DTULFSRWD	0x0234	W	0x00000000	DTU LFSR Seed for Write Data Generation Register
DDR_PCTL_DTULFSRRD	0x0238	W	0x00000000	DTU LFSR Seed for Read Data Generation Register
DDR_PCTL_DTUEAF	0x023c	W	0x00000000	DTU Error Address FIFO Register
DDR_PCTL_DFITCTRLDELAY	0x0240	W	0x00000002	DFI tctrl_delay Register
DDR_PCTL_DFIODTCFG	0x0244	W	0x00000000	DFI ODT Configuration

Name	Offset	Size	Reset Value	Description
DDR_PCTL_DFIODTCFG1	0x0248	W	0x06060000	DFI ODT Timing Configuration 1 (for Latency and Length)
DDR_PCTL_DFIODTRANKMAP	0x024c	W	0x00008421	DFI ODT Rank Mapping
DDR_PCTL_DFITPHYWRDATA	0x0250	W	0x00000001	DFI tphy_wrdata Register
DDR_PCTL_DFITPHYWRLAT	0x0254	W	0x00000001	DFI tphy_wrlat Register
DDR_PCTL_DFITRDDATAEN	0x0260	W	0x00000001	DFI trddata_en Register
DDR_PCTL_DFITPHYRDLAT	0x0264	W	0x0000000f	DFI tphy_rdlat Register
DDR_PCTL_DFITPHYUPDTYPE0	0x0270	W	0x00000010	DFI tphyupd_type0 Register
DDR_PCTL_DFITPHYUPDTYPE1	0x0274	W	0x00000010	DFI tphyupd_type1 Register
DDR_PCTL_DFITPHYUPDTYPE2	0x0278	W	0x00000010	DFI tphyupd_type2 Register
DDR_PCTL_DFITPHYUPDTYPE3	0x027c	W	0x00000010	DFI tphyupd_type3 Register
DDR_PCTL_DFITCTRLUPDMIN	0x0280	W	0x00000010	DFI tctrlupd_min Register
DDR_PCTL_DFITCTRLUPDMAX	0x0284	W	0x00000040	DFI tctrlupd_max Register
DDR_PCTL_DFITCTRLUPDDLY	0x0288	W	0x00000008	DFI tctrlupddly Register
DDR_PCTL_DFIUPDCFG	0x0290	W	0x00000003	DFI Update Configuration Register
DDR_PCTL_DFITREFMSKI	0x0294	W	0x00000000	DFI Masked Refresh Interval
DDR_PCTL_DFITCTRLUPDI	0x0298	W	0x00000000	DFI tctrlupd_interval Register
DDR_PCTL_DFITRCFG0	0x02ac	W	0x00000000	DFI Training Configuration 0 Register
DDR_PCTL_DFITRSTAT0	0x02b0	W	0x00000000	DFI Training Status 0 Register
DDR_PCTL_DFITRWRLVLEN	0x02b4	W	0x00000000	DFI Training dfi_wrlvl_en Register
DDR_PCTL_DFITRRDLVLEN	0x02b8	W	0x00000000	DFI Training dfi_rdlvl_en Register
DDR_PCTL_DFITRRDLVLGATEEN	0x02bc	W	0x00000000	DFI Training dfi_rdlvl_gate_en Register
DDR_PCTL_DFISTSTAT0	0x02c0	W	0x00000000	DFI Status Status 0 Register
DDR_PCTL_DFISTCFG0	0x02c4	W	0x00000000	DFI Status Configuration 0 Register
DDR_PCTL_DFISTCFG1	0x02c8	W	0x00000000	DFI Status Configuration 1 Register

Name	Offset	Size	Reset Value	Description
DDR_PCTL_DFITDRAMCLKEN	0x02d0	W	0x00000002	DFI tdram_clk_enable Register
DDR_PCTL_DFITDRAMCLKDIS	0x02d4	W	0x00000002	DFI tdram_clk_disable Register
DDR_PCTL_DFISTCFG2	0x02d8	W	0x00000000	DFI Status Configuration 2 Register
DDR_PCTL_DFISTPARCLR	0x02dc	W	0x00000000	DFI Status Parity Clear Register
DDR_PCTL_DFISTPARLOG	0x02e0	W	0x00000000	DFI Status Parity Log Register
DDR_PCTL_DFILPCFG0	0x02f0	W	0x00070000	DFI Low Power Configuration 0 Register
DDR_PCTL_DFITRWRLVLRESP0	0x0300	W	0x00000000	DFI Training dfi_wrlvl_resp Status 0 Register
DDR_PCTL_DFITRWRLVLRESP1	0x0304	W	0x00000000	DFI Training dfi_wrlvl_resp Status 1 Register
DDR_PCTL_DFITRWRLVLRESP2	0x0308	W	0x00000000	DFI Training dfi_wrlvl_resp Status 2 Register
DDR_PCTL_DFITRRDLVLRESP0	0x030c	W	0x00000000	DFI Training dfi_rdlvl_resp Status 0 Register
DDR_PCTL_DFITRRDLVLRESP1	0x0310	W	0x00000000	DFI Training dfi_rdlvl_resp Status 1 Register
DDR_PCTL_DFITRRDLVLRESP2	0x0314	W	0x00000000	DFI Training dfi_rdlvl_resp Status 2 Register
DDR_PCTL_DFITRWRLVLDELAY0	0x0318	W	0x00000000	DFI Training dfi_wrlvl_delay Configuration 0 Register
DDR_PCTL_DFITRWRLVLDELAY1	0x031c	W	0x00000000	DFI Training dfi_wrlvl_delay Configuration 1 Register
DDR_PCTL_DFITRWRLVLDELAY2	0x0320	W	0x00000000	DFI Training dfi_wrlvl_delay Configuration 2 Register
DDR_PCTL_DFITRRDLVLDELAY0	0x0324	W	0x00000000	DFI Training dfi_rdlvl_delay Configuration 0 Register
DDR_PCTL_DFITRRDLVLDELAY1	0x0328	W	0x00000000	DFI Training dfi_rdlvl_delay Configuration 1 Register
DDR_PCTL_DFITRRDLVLDELAY2	0x032c	W	0x00000000	DFI Training dfi_rdlvl_delay Configuration 2 Register
DDR_PCTL_DFITRRDLVLGATEDELAY0	0x0330	W	0x00000000	DFI Training dfi_rdlvl_gate_delay Configuration 0
DDR_PCTL_DFITRRDLVLGATEDELAY1	0x0334	W	0x00000000	DFI Training dfi_rdlvl_gate_delay Configuration 1
DDR_PCTL_DFITRRDLVLGATEDELAY2	0x0338	W	0x00000000	DFI Training dfi_rdlvl_gate_delay Configuration 2
DDR_PCTL_DFITRCMD	0x033c	W	0x00000000	DFI Training Command Register
DDR_PCTL_IPVR	0x03f8	W	0x00000000	IP Version Register
DDR_PCTL_IPTR	0x03fc	W	0x44574300	IP Type Register

Name	Offset	Size	Reset Value	Description
DDRPHY_REG00	0x0000	W	0x000000ff	DDR PHY register 00
DDRPHY_REG01	0x0004	W	0x00000004	DDR PHY register 01
DDRPHY_REG02	0x0008	W	0x00000000	DDR PHY register 02
DDRPHY_REG03	0x000c	W	0x00000022	DDR PHY register 03
DDRPHY_REG04	0x0010	W	0x00000000	DDR PHY register 04
DDRPHY_REG0B	0x002c	W	0x00000060	DDR PHY register 0B
DDRPHY_REG0C	0x0030	W	0x00000000	DDR PHY register 0C
DDRPHY_REG11	0x0044	W	0x000000aa	DDR PHY register 11
DDRPHY_REG12	0x0048	W	0x00000002	DDR PHY register 12
DDRPHY_REG13	0x004c	W	0x0000000c	DDR PHY register 13
DDRPHY_REG14	0x0050	W	0x00000000	DDR PHY register 14
DDRPHY_REG16	0x0058	W	0x000000aa	DDR PHY register 16
DDRPHY_REG20	0x0080	W	0x000000aa	DDR PHY register 20
DDRPHY_REG21	0x0084	W	0x000000aa	DDR PHY register 21
DDRPHY_REG26	0x0098	W	0x0000000c	DDR PHY register 26
DDRPHY_REG27	0x009c	W	0x00000000	DDR PHY register 27
DDRPHY_REG28	0x00a0	W	0x00000001	DDR PHY register 28
DDRPHY_REG30	0x00c0	W	0x000000aa	DDR PHY register 30
DDRPHY_REG31	0x00c4	W	0x000000aa	DDR PHY register 31
DDRPHY_REG36	0x00d4	W	0x0000000c	DDR PHY register 36
DDRPHY_REG37	0x00d8	W	0x00000000	DDR PHY register 37
DDRPHY_REG38	0x00dc	W	0x00000001	DDR PHY register 38
DDRPHY_REG40	0x0100	W	0x000000aa	DDR PHY register 40
DDRPHY_REG41	0x0104	W	0x000000aa	DDR PHY register 41
DDRPHY_REG46	0x0118	W	0x0000000c	DDR PHY register 46
DDRPHY_REG47	0x011c	W	0x00000000	DDR PHY register 47
DDRPHY_REG48	0x0120	W	0x00000001	DDR PHY register 48
DDRPHY_REG50	0x0140	W	0x000000aa	DDR PHY register 50
DDRPHY_REG51	0x0144	W	0x000000aa	DDR PHY register 51
DDRPHY_REG56	0x0158	W	0x0000000c	DDR PHY register 56
DDRPHY_REG57	0x015c	W	0x00000000	DDR PHY register 57
DDRPHY_REG58	0x0160	W	0x00000001	DDR PHY register 58
DDRPHY_REGB0	0x02c0	W	0x00000077	DDR PHY register B0
DDRPHY_REGB1	0x02c4	W	0x00000077	DDR PHY register B1
DDRPHY_REGB2	0x02c8	W	0x00000077	DDR PHY register B2
DDRPHY_REGB3	0x02cc	W	0x00000077	DDR PHY register B3
DDRPHY_REGB4	0x02d0	W	0x00000077	DDR PHY register B4
DDRPHY_REGB5	0x02d4	W	0x00000077	DDR PHY register B5
DDRPHY_REGB6	0x02d8	W	0x00000077	DDR PHY register B6
DDRPHY_REGB7	0x02dc	W	0x00000077	DDR PHY register B7
DDRPHY_REGB8	0x02e0	W	0x00000077	DDR PHY register B8
DDRPHY_REGB9	0x02e4	W	0x00000077	DDR PHY register B9
DDRPHY_REGBA	0x02e8	W	0x00000077	DDR PHY register BA
DDRPHY_REGBB	0x02ec	W	0x00000077	DDR PHY register BB
DDRPHY_REGBC	0x02f0	W	0x00000077	DDR PHY register BC
DDRPHY_REGBD	0x02f4	W	0x00000077	DDR PHY register BD
DDRPHY_REGBE	0x02f8	W	0x00000077	DDR PHY register BE
DDRPHY_REGC0	0x0300	W	0x00000077	DDR PHY register C0
DDRPHY_REGC1	0x0304	W	0x00000077	DDR PHY register C1
DDRPHY_REGC2	0x0308	W	0x00000077	DDR PHY register C2
DDRPHY_REGC3	0x030c	W	0x00000077	DDR PHY register C3
DDRPHY_REGC4	0x0310	W	0x00000077	DDR PHY register C4
DDRPHY_REGC5	0x0314	W	0x00000077	DDR PHY register C5

Name	Offset	Size	Reset Value	Description
DDRPHY_REGC6	0x0318	W	0x00000077	DDR PHY register C6
DDRPHY_REGC7	0x031c	W	0x00000077	DDR PHY register C7
DDRPHY_REGC8	0x0320	W	0x00000077	DDR PHY register C8
DDRPHY_REGC9	0x0324	W	0x00000077	DDR PHY register C9
DDRPHY_REGCA	0x0328	W	0x00000077	DDR PHY register CA
DDRPHY_REGCB	0x032c	W	0x00000077	DDR PHY register CB
DDRPHY_REGCC	0x0330	W	0x00000077	DDR PHY register CC
DDRPHY_REGCD	0x0334	W	0x00000077	DDR PHY register CD
DDRPHY_REGCE	0x0338	W	0x00000077	DDR PHY register CE
DDRPHY_REGCF	0x033c	W	0x00000077	DDR PHY register CF
DDRPHY_REGD0	0x0340	W	0x00000077	DDR PHY register D0
DDRPHY_REGD1	0x0344	W	0x00000077	DDR PHY register D1
DDRPHY_REGD2	0x0348	W	0x00000077	DDR PHY register D2
DDRPHY_REGD3	0x034c	W	0x00000077	DDR PHY register D3
DDRPHY_REGD4	0x0350	W	0x00000077	DDR PHY register D4
DDRPHY_REGD5	0x0354	W	0x00000077	DDR PHY register D5
DDRPHY_REGD6	0x0358	W	0x00000077	DDR PHY register D6
DDRPHY_REGD7	0x035c	W	0x00000077	DDR PHY register D7
DDRPHY_REGD8	0x0360	W	0x00000077	DDR PHY register D8
DDRPHY_REGD9	0x0364	W	0x00000077	DDR PHY register D9
DDRPHY_REGDA	0x0368	W	0x00000077	DDR PHY register DA
DDRPHY_REGDB	0x036c	W	0x00000077	DDR PHY register DB
DDRPHY_REGDC	0x0370	W	0x00000077	DDR PHY register DC
DDRPHY_REGDD	0x0374	W	0x00000077	DDR PHY register DD
DDRPHY_REGDE	0x0378	W	0x00000077	DDR PHY register DE
DDRPHY_REGDF	0x037c	W	0x00000077	DDR PHY register DF
DDRPHY_REGE0	0x0380	W	0x00000007	DDR PHY register E0
DDRPHY_REGE1	0x0384	W	0x00000077	DDR PHY register E1
DDRPHY_REGE2	0x0388	W	0x00000077	DDR PHY register E2
DDRPHY_REGE3	0x038c	W	0x00000077	DDR PHY register E3
DDRPHY_REGE4	0x0390	W	0x00000077	DDR PHY register E4
DDRPHY_REGE5	0x0394	W	0x00000077	DDR PHY register E5
DDRPHY_REGE6	0x0398	W	0x00000077	DDR PHY register E6
DDRPHY_REGE7	0x039c	W	0x00000077	DDR PHY register E7
DDRPHY_REGE8	0x03a0	W	0x00000077	DDR PHY register E8
DDRPHY_REGE9	0x03a4	W	0x00000077	DDR PHY register E9
DDRPHY_REGEA	0x03a8	W	0x00000077	DDR PHY register EA
DDRPHY_REGEB	0x03ac	W	0x00000007	DDR PHY register EB
DDRPHY_REGFA	0x03e8	W	0x00000000	DDR PHY register FA
DDRPHY_REGFB	0x03ec	W	0x00000000	DDR PHY register FB
DDRPHY_REGFC	0x03f0	W	0x00000000	DDR PHY register FC
DDRPHY_REGFD	0x03f4	W	0x00000000	DDR PHY register FD
DDRPHY_REGFE	0x03f8	W	0x00000000	DDR PHY register FE
DDRPHY_REGFF	0x03fc	W	0x00000000	DDR PHY register FF

*Notes:Size:**B**- Byte (8 bits) access, **HW**- Half WORD (16 bits) access, **W**-WORD (32 bits) access*

7.4.2 Detail Register Description

DDR_PCTL_SCFG

Address: Operational Base + offset (0x0000)

State Configuration Register

Bit	Attr	Reset Value	Description
31:12	RO	0x0	reserved

Bit	Attr	Reset Value	Description
11:8	RW	0x3	bbflags_timing The n_bbflags is a NIF output vector which provides combined information about the status of each memory bank. The de-assertion is based on when precharge, activates,reads/writes are scheduled by the TCU block. It may be possible to de-assert n_bbflags earlier than calculated by the TCU block. Programming bbflags_timing is used to achieve this.The maximum recommended value is: UPCTL_TCU_SED_P - TRP.t_rp. The programmed value is the maximum number of "early" cycles that n_bbflags maybe de-asserted. The actual achieved de-assertion depends on the traffic profile. In 1:2 mode the maximum allowed programmable value is 4'b0111 In 1:1 mode the value can be 4'b1111
7	RO	0x0	reserved
6	RW	0x0	nfifo_nif1_dis For Synopsys internal use only for NFIFO testing. 1'b0 = Only supported setting. 1'b1 = For Synopsys internal use only.
5:1	RO	0x0	reserved
0	RW	0x0	hw_low_power_en Enables the hardware low-power interface. Allows the system to request via hardware (c_sysreq input) to enter the memories into Self-Refresh. The handshaking between the request and acknowledge hardware low power signals (c_sysreq and c_sysack, respectively) is always performed, but the uPCTL response depends on the value set on this register field and by the value driven on the c_active_in input pin. 1'b0 = Disabled. Requests are always denied and uPCTL is unaffected by c_sysreq 1'b1 = Enabled. Requests are accepted or denied, depending on the current operational state of uPCTL and on the value of c_active_in.

DDR_PCTL_SCTL

Address: Operational Base + offset (0x0004)

Operational State Control Register

Bit	Attr	Reset Value	Description
31:3	RO	0x0	reserved

Bit	Attr	Reset Value	Description
2:0	RW	0x0	state_cmd Issues an operational state transition request to the uPCTL. 3'b000 = INIT (move to Init_mem from Config) 3'b001 = CFG (move to Config from Init_mem or Access) 3'b010 = GO (move to Access from Config) 3'b011 = SLEEP (move to Low_power from Access) 3'b100 = WAKEUP (move to Access from Low_power) Others = Reserved

DDR_PCTL_STAT
Address: Operational Base + offset (0x0008)
Operational State Status Register

Bit	Attr	Reset Value	Description
31:7	RO	0x0	reserved
6:4	RO	0x0	lp_trig Reports the status of what triggered an entry to Low_power state. Is only set if in Low_power state. The individual bits report the following: - lp_trig[2]: Software driven due to SCTL.state_cmd==SLEEP. - lp_trig[1]: Hardware driven due to Hardware Low Power Interface. - lp_trig[0]: Hardware driven due to Auto Self Refresh (MCFG1.sr_idle>0). Note, if more than one trigger happens at the exact same time, more than one bit of lp_trig may be asserted high.
3	RO	0x0	reserved
2:0	RO	0x0	ctl_stat Returns the current operational state of the uPCTL. 3'b000 = Init_mem 3'b001 = Config 3'b010 = Config_req 3'b011 = Access 3'b100 = Access_req 3'b101 = Low_power 3'b110 = Low_power_entry_req 3'b111 = Low_power_exit_req Others = Reserved

DDR_PCTL_INTRSTAT
Address: Operational Base + offset (0x000c)
Interrupt Status Register

Bit	Attr	Reset Value	Description
31:2	RO	0x0	reserved

Bit	Attr	Reset Value	Description
1	RO	0x0	parity_intr Indicates that a DFI parity error has been detected 1'b0 = No error 1'b1 = Parity error
0	RO	0x0	ecc_intr Indicates that an ECC error has been detected 1'b0 = No error 1'b1 = Parity error

DDR_PCTL_MCMD
Address: Operational Base + offset (0x0040)
Memory Command Register

Bit	Attr	Reset Value	Description
31	R/WSC	0x0	start_cmd Start command. When this bit is set to 1, the command operation defined in the cmd_opcode field is started. This bit is automatically cleared by the uPCTL after the command is finished. The application can poll this bit to determine when uPCTL is ready to accept another command. This bit cannot be cleared to 1'b0 by software.
30:28	RO	0x0	reserved
27:24	RW	0x0	cmd_add_del Set the additional delay associated with each command to 2^n internal timers clock cycles, where n is the bit field value. If n=0, the delay is 0. Max value is n=10.
23:20	RW	0x1	rank_sel Rank select for the command to be executed. 4'b0001 = Rank 0 4'b0010 = Rank 1 4'b0100 = Rank 2 4'b1000 = Rank 3 4'b0000 = Reserved Multiple 1'b1s in rank_sel mean multiple ranks are selected, which is useful broadcasting commands in parallel to multiple ranks during initialization and configuration of the memories. If MCMD.cmd_opcode=RSTL, all ranks should be selected as it cannot be performed to individual ranks

Bit	Attr	Reset Value	Description
19:17	RW	0x0	bank_addr Mode Register address driven on the memory bank address bits, BA1, BA0, during a Mode Register Set operation, defined by cmd_opcode=MRS. For other values of cmd_opcode, this field is ignored. 3'b000 = MR0 (MR in DDR2) 3'b001 = MR1 (EMR in DDR2) 3'b010 = MR2 (EMR(2) in DDR2) 3'b011 = MR3 (EMR(3) in DDR2) Others = Reserved
16:4	RW	0x0000	cmd_addr Mode Register value driven on the memory address bits, A12 to A0, during a Mode Register Set operation defined by cmd_opcode=MRS. For other values of cmd_opcode this field is ignored. Refer to the memory specification for the correct settings of the various bits of this field during a MRS operation. If LPDDR2, this fields is merged into bank_addr - lpddr2_addr
3:0	RW	0x0	cmd_opcode Command to be issued to the memory. 4'b000 = Deselect. This is only used for timing purposes, no actual direct Deselect command is passed to the memories. 4'b0001 = Precharge All (PREA) 4'b0010 = Refresh (REF) 4'b0011 = Mode Register Set (MRS) - is MRW in LPDDR2, MRS otherwise 4'b0100 = ZQ Calibration Short (ZQCS, only applies to LPDDR2/DDR3) 4'b0101 = ZQ Calibration Long (ZQCL, only applies to LPDDR2/DDR3) 4'b0110 = Software Driven Reset (RSTL, only applies to DDR3) 4'b0111 = Reserved 4'b1000 - Mode Register Read (MRR) - is MRR in LPDDR2, is SRR in mDRR and is MPR in DDR3 4'b1001 - Deep Power Down Entry (DPDE, only applies to mDDR/LPDDR2) Others - Reserved

DDR_PCTL_POWCTL
Address: Operational Base + offset (0x0044)
Power Up Control Register

Bit	Attr	Reset Value	Description
31:1	RO	0x0	reserved

Bit	Attr	Reset Value	Description
0	R/W SC	0x0	power_up_start Start the memory power up sequence. When this bit is set to 1'b1, uPCTL starts the CKE and RESET# power up sequence to the memories. This bit is automatically cleared by uPCTL after the sequence is completed. This bit cannot be cleared to 1'b0 by software.

DDR_PCTL_POWSTAT
Address: Operational Base + offset (0x0048)
Power Up Status Register

Bit	Attr	Reset Value	Description
31:1	RO	0x0	reserved
0	RO	0x0	power_up_done Returns the status of the memory power-up sequence. 1'b0 = Power-up sequence has not been performed. 1'b1 = Power-up sequence has been performed.

DDR_PCTL_CMDTSTAT
Address: Operational Base + offset (0x004c)
Command Timers Status Register

Bit	Attr	Reset Value	Description
31:1	RO	0x0	reserved
0	RO	0x0	cmd_tstat Returns the status of the timers for memory commands. This ANDs all the command timers together. 1'b0 = One or more command timers has not expired. 1'b1 = All command timers have expired.

DDR_PCTL_CMDTSTATEN
Address: Operational Base + offset (0x0050)
Command Timers Status Enable Register

Bit	Attr	Reset Value	Description
31:1	RO	0x0	reserved
0	RW	0x0	cmd_tstat_en Enables the generation of the status of the timers for memory commands. Is enabled before CMDTSTAT register is read. 1'b0 - Disabled 1'b1 - Enabled

DDR_PCTL_MRRCFG0
Address: Operational Base + offset (0x0060)
Mode Register Read Configuration 0

Bit	Attr	Reset Value	Description
31:4	RO	0x0	reserved

Bit	Attr	Reset Value	Description
3:0	RW	0x0	mrr_byte_sel Selects which byte's data to store when performing an MRR command via MCMD. Legal Values: 0 .. 8

DDR_PCTL_MRRSTAT0
Address: Operational Base + offset (0x0064)
Mode Register Read Status 0 Register

Bit	Attr	Reset Value	Description
31:24	RO	0x00	mrrstat_beat3 MRR/MPR read data beat 3
23:16	RO	0x00	mrrstat_beat2 MRR/MPR read data beat 2
15:8	RO	0x00	mrrstat_beat1 MRR/MPR read data beat 1
7:0	RO	0x00	mrrstat_beat0 MRR/MPR read data beat 0

DDR_PCTL_MRRSTAT1
Address: Operational Base + offset (0x0068)
Mode Register Read Status 0 Register

Bit	Attr	Reset Value	Description
31:24	RO	0x00	mrrstat_beat7 MRR/MPR read data beat 7
23:16	RO	0x00	mrrstat_beat6 MRR/MPR read data beat 6
15:8	RO	0x00	mrrstat_beat5 MRR/MPR read data beat 5
7:0	RO	0x00	mrrstat_beat4 MRR/MPR read data beat 4

DDR_PCTL_MCFG1
Address: Operational Base + offset (0x007c)
Memory Configuration 1 Register

Bit	Attr	Reset Value	Description
31	RW	0x0	hw_exit_idle_en When this bit is programmed to 1'b1 the c_active_in pin can be used to exit from the automatic clock stop , power down or self-refresh modes.
30:24	RO	0x0	reserved
23:16	RW	0x00	hw_idle Hardware idle period. The c_active output is driven high if the NIF is idle in Access state for hw_idle * 32 * n_clk cycles. The hardware idle function is disabled when hw_idle=0.
15:11	RO	0x0	reserved

Bit	Attr	Reset Value	Description
10:8	RW	0x0	tfaw_cfg_offset used to fine tune the tFAW setting as programmed in MCFG.tfaw_cfg. tFAW=(4+MCFG.tfaw_cfg)*tRRD-tfaw_cfg_offset
7:0	RW	0x00	sr_idle Self Refresh idle period. Memories are placed into Self-Refresh mode if the NIF is idle in Access state for sr_idle * 32 * n_clk cycles. The automatic self refresh function is disabled when sr_idle=0.

DDR_PCTL_MCFG
Address: Operational Base + offset (0x0080)
Memory Configuration Register

Bit	Attr	Reset Value	Description
31:24	RW	0x00	mddr_lpddr2_clock_stop_idle Clock stop idle period in n_clk cycles. Memories are placed into clock stop mode if the NIF is idle for mddr_lpddr2_clkstop_idle n_clk cycles. The automatic clock stop function is disabled when mddr_lpddr2_clkstop_idle=0. Clock stop mode is only applicable in mDDR/LPDDR2.
23:22	RW	0x0	mddr_lpddr2_en mDDR/LPDDR2 Enable. Enables support for mDDR or LPDDR2. 2'b00 = mDDR/LPDDR2 Disabled 2'b10 = mDDR Enabled 2'b11 = LPDDR2 Enabled Others= Reserved.
21:20	RW	0x0	mddr_lpddr2_bl Field0000 Abstract mDDR/LPDDR2 Burst Length. The BL setting must be consistent with the value programmed into the BL field of MR. 2'b00 = BL2, Burst length of 2 (MR.BL=3'b001, mDDR only) 2'b01 = BL4, Burst length of 4 (MR.BL=3'b010, for mDDR and LPDDR2) 2'b10 = BL8, Burst length of 8 (MR.BL=3'b011, for mDDR and LPDDR2) 2'b11 = BL16, Burst length of 16 (MR.BL=3'b100, for mDDR and LPDDR2) This value is effective only if MCFG.mddr_lpddr2_en[1]=1'b1. Otherwise, MCFG.mem_bl is used to define uPCTL's Burst Length (for DDR2/DDR3).
19:18	RW	0x1	tfaw_cfg Field0000 Abstract Sets tFAW to be 4, 5 or 6 times tRRD. 2'b00 = set tFAW=4*tRRD 2'b01 = set tFAW=5*tRRD 2'b10 = set tFAW=6*tRRD

Bit	Attr	Reset Value	Description
17	RW	0x0	pd_exit_mode Selects the mode for Power Down Exit. For DDR2/DDR3, the power down exit mode setting in uPCTL must be consistent with the value programmed into the power down exit mode bit of MR0. For mDDR/LPDDR2, only fast exit mode is valid. 1'b0 = slow exit 1'b1 = fast exit
16	RW	0x0	pd_type Sets the Power down type. 1'b0 = Precharge Power Down 1'b1 = Active Power Down
15:8	RW	0x00	pd_idle Power-down idle period in n_clk cycles. Memories are placed into power-down mode if the NIF is idle for pd_idle n_clk cycles. The automatic power down function is disabled when pd_idle=0.
7	RO	0x0	reserved
6	RW	0x0	lpddr2_s4 Enables LPDDR2-S4 support. 1'b0 = LPDDR2-S4 disabled (LPDDR2-S2 enabled) 1'b1 = LPDDR2-S4 enabled
5	RW	0x1	ddr3_en Select DDR2 or DDR3 protocol. Ignored, if mDDR or LPDDR2 support is enabled. 1'b0 = DDR2 Protocol Rules 1'b1 = DDR3 Protocol Rules
4	RW	0x0	stagger_cs For multi-rank commands from the DCU, stagger the assertion of CS_N to odd and even ranks by one n_clk cycle. This is useful when using RDIMMs, when multi-rank commands may be interpreted as writes to control words in the register chip. 1'b0 = Do not stagger CS_N 1'b1 = Stagger CS_N
3	RW	0x0	two_t_en Enables 2T timing for memory commands. 1'b0= Disabled 1'b1 = Enabled
2	RW	0x0	bl8int_en Setting this bit enables the BL8 interrupt function of DDR2. This is the capability to early terminate a BL8 after only 4 DDR beats by issuing the next command two cycles earlier. This functionality is only available for DDR2 memories and this setting is ignored for mDDR/LPDDR2 and DDR3. 1'b0 = Disabled 1'b1 = Enabled

Bit	Attr	Reset Value	Description
1	RW	0x0	cke_or_en This bit is intended to be set for 4-rank RDIMMs, which have a 2-bit CKE input. If set, dfi_cke[0] is asserted to enable either of the even ranks (0 and 2), while dfi_cke[1] is asserted to enable either of the odd ranks (1 and 3). dfi_cke[3:2] are inactive (0) 1'b0: Disabled 1'b1: Enabled
0	RW	0x0	mem_bl DDR Burst Length. The BL setting in DDR2 / DDR3 must be consistent with the value programmed into the BL field of MR0. 1'b0 = BL4, Burst length of 4 (MR0.BL=3'b010, DDR2 only) 1'b1 = BL8, Burst length of 8 (MR0.BL=3'b011 for DDR2, MR0.BL=2'b00 for DDR3)

DDR_PCTL_PPCFG
Address: Operational Base + offset (0x0084)
Partially Populated Memories Configuration Register

Bit	Attr	Reset Value	Description
31:9	RO	0x0	reserved
8:1	RW	0x00	rpmem_dis Reduced Population Disable bits. Setting these bits disables the corresponding NIF/DDR data lanes from writing or reading data. Lane 0 is always present, hence only 8 bits are required for the remaining lanes including the ECC lane. In 1:2 mode bit 0 of rpmem_dis covers n_wdata/n_rdata/m_ctl_d/m_phy_q[63:32], bit 1 [95:64] etc. In 1:1 mode bit 0 of rpmem_dis covers n_wdata/n_rdata/m_ctl_d/m_phy_q[31:16], bit 2 [47:32] etc. There are no restrictions on which byte lanes can be disabled, other than byte lane 0 is required. Gaps between enabled byte lanes are allowed For each bit: 1'b0 = lane exists 1'b1 = lane is disabled
0	RW	0x0	ppmem_en Partially Population Enable bit. Setting this bit enables the partial population of external memories where the entire application bus is routed to a reduced size memory system. The lower half of the SDRAM data bus, bit 0 up to bit UPCTL_M_DW/2-1, is the active portion when Partially Populated memories are enabled. An example of this is shown in Example 8-1 on page 263. 1'b0 = Disabled 1'b1 = Enabled

DDR_PCTL_MSTAT

Address: Operational Base + offset (0x0088)
Memory Status Register

Bit	Attr	Reset Value	Description
31:3	RO	0x0	reserved
2	RO	0x0	self_refresh Indicates if uPCTL, through auto self refresh, has placed the memories in Self Refresh. 1'b0 = Memory is not in Self Refresh 1'b1 = Memory is in Self Refresh
1	RO	0x0	clock_stop Indicates if uPCTL has placed the memories in Clock Stop. 1'b0 = Memory is not in Clock Stop 1'b1 = Memory is in Clock Stop
0	RO	0x0	power_down Indicates if uPCTL has placed the memories in Power Down. 1'b0 = Memory is not in Power Down 1'b1 = Memory is in Power-Down

DDR_PCTL_LPDDR2ZQCFG
Address: Operational Base + offset (0x008c)
LPDDR2 ZQ Configuration Register

Bit	Attr	Reset Value	Description
31:24	RW	0xab	zqcl_op Value to drive on memory address bits [19:12] for an automatic hardware generated ZQCL command (LPDDR2). Corresponds to OP7 .. OP0 of Mode Register Write (MRW) command which is used to send ZQCL command to memory.
23:16	RW	0x0a	zqcl_ma Value to drive on memory address bits [11:4] for an automatic hardware generated ZQCL command (LPDDR2). Corresponds to MA7 .. MA0 of Mode Register Write (MRW) command which is used to send ZQCL command to memory.
15:8	RW	0x56	zqcs_op Value to drive on memory address bits [19:12] for an automatic hardware generated ZQCS command (LPDDR2). Corresponds to OP7 .. OP0 of Mode Register Write (MRW) command which is used to send ZQCS command to memory.
7:0	RW	0x0a	zqcs_ma Value to drive on memory address bits [11:4] for an automatic hardware generated ZQCS command (LPDDR2). Corresponds to MA7 .. MA0 of Mode Register Write (MRW) command which is used to send ZQCS command to memory.

DDR_PCTL_DTUPDES
Address: Operational Base + offset (0x0094)
DTU Status Register

Bit	Attr	Reset Value	Description
31:14	RO	0x0	reserved
13	RO	0x0	dtu_rd_missing Indicates if one or more read beats of data did not return from memory.
12:9	RO	0x0	dtu_eaffl Indicates the number of entries in the FIFO that is holding the log of error addresses for data comparison
8	RO	0x0	dtu_random_error Indicates that the random data generated had some failures when written and read to the memories
7	RO	0x0	dtu_err_b7 Detected at least 1 bit error for bit 7 in the programmable data buffers
6	RO	0x0	dtu_err_b6 Detected at least 1 bit error for bit 6 in the programmable data buffers
5	RO	0x0	dtu_err_b5 Detected at least 1 bit error for bit 5 in the programmable data buffers
4	RO	0x0	dtu_err_b4 Detected at least 1 bit error for bit 4 in the programmable data buffers
3	RO	0x0	dtu_err_b3 Detected at least 1 bit error for bit 3 in the programmable data buffers
2	RO	0x0	dtu_err_b2 Detected at least 1 bit error for bit 2 in the programmable data buffers
1	RO	0x0	dtu_err_b1 Detected at least 1 bit error for bit 1 in the programmable data buffers
0	RO	0x0	dtu_err_b0 Detected at least 1 bit error for bit 0 in the programmable data buffers

DDR_PCTL_DTUNA
Address: Operational Base + offset (0x0098)
DTU Number of Addresses Created Register

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	dtu_num_address Indicates the number of addresses that were created on the NIF interface during random data generation.

DDR_PCTL_DTUNE
Address: Operational Base + offset (0x009c)

DTU Number of Errors Register

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	dtu_num_errors Indicates the number of errors that were detected on the readback of the NIF data during random data generation.

DDR_PCTL_DTUPRD0

Address: Operational Base + offset (0x00a0)

DTU Parallel Read 0 Register

Bit	Attr	Reset Value	Description
31:16	RO	0x0000	dtu_allbits_1 Allows all the bit ones from each of the 16 received read bytes to be read in parallel. Used as part of read data eye training where a transition is required to be monitored to train the eye.
15:0	RO	0x0000	dtu_allbits_0 Allows all the bit zeros from each of the 16 received read bytes to be read in parallel. Used as part of read data eye training where a transition is required to be monitored to train the eye.

DDR_PCTL_DTUPRD1

Address: Operational Base + offset (0x00a4)

DTU Parallel Read 1 Register

Bit	Attr	Reset Value	Description
31:16	RO	0x0000	dtu_allbits_3 Allows all the bit threes from each of the 16 received read bytes to be read in parallel. Used as part of read data eye training where a transition is required to be monitored to train the eye.
15:0	RO	0x0000	dtu_allbits_2 Allows all the bit twos from each of the 16 received read bytes to be read in parallel. Used as part of read data eye training where a transition is required to be monitored to train the eye.

DDR_PCTL_DTUPRD2

Address: Operational Base + offset (0x00a8)

DTU Parallel Read 2 Register

Bit	Attr	Reset Value	Description
31:16	RO	0x0000	dtu_allbits_5 Allows all the bit fives from each of the 16 received read bytes to be read in parallel. Used as part of read data eye training where a transition is required to be monitored to train the eye.
15:0	RO	0x0000	dtu_allbits_4 Allows all the bit fours from each of the 16 received read bytes to be read in parallel. Used as part of read data eye training where a transition is required to be monitored to train the eye.

DDR_PCTL_DTUPRD3

Address: Operational Base + offset (0x00ac)
DTU Parallel Read 3 Register

Bit	Attr	Reset Value	Description
31:16	RO	0x0000	dtu_allbits_7 Allows all the bit sevens from each of the 16 received read bytes to be read in parallel. Used as part of read data eye training where a transition is required to be monitored to train the eye.
15:0	RO	0x0000	dtu_allbits_6 Allows all the bit sixes from each of the 16 received read bytes to be read in parallel. Used as part of read data eye training where a transition is required to be monitored to train the eye.

DDR_PCTL_DTUAWDT
Address: Operational Base + offset (0x00b0)
DTU Address Width Register

Bit	Attr	Reset Value	Description
31:11	RO	0x0	reserved
10:9	RW	0x1	number_ranks Number of supported memory ranks. 2'b00 = 1 rank 2'b01 = 2 ranks 2'b10 = 3 ranks 2'b11 = 4 ranks
8	RO	0x0	reserved
7:6	RW	0x2	row_addr_width Width of the memory row address bits. 2'b00 = 13 bits wide 2'b01 = 14 bits wide 2'b10 = 15 bits wide 2'b11 = 16 bits wide
5	RO	0x0	reserved
4:3	RW	0x2	bank_addr_width Field0000 Abstract Width of the memory bank address bits. 2'b00 = 2 bits wide (4 banks) 2'b01 = 3 bits wide (8 banks) Others = Reserved
2	RO	0x0	reserved
1:0	RW	0x0	column_addr_width Field0000 Abstract Width of the memory column address bits. 2'b00 = 7 bits wide 2'b01 = 8 bits wide 2'b10 = 9 bits wide 2'b11 = 10 bits wide

DDR_PCTL_TOGCNT1U
Address: Operational Base + offset (0x00c0)
Toggle Counter 1us Register

Bit	Attr	Reset Value	Description
31:10	RO	0x0	reserved
9:0	RW	0x064	toggle_counter_1u The number of internal timers clock cycles

DDR_PCTL_TINIT
Address: Operational Base + offset (0x00c4)
t_init Timing Register

Bit	Attr	Reset Value	Description
31:9	RO	0x0	reserved
8:0	RW	0x0c8	t_init Defines the time period (in us) to hold dfi_cke and dfi_reset_n stable during the memory power up sequence. The value programmed must correspond to at least 200us. The actual time period defined is TINIT * TOGCNT1U * internal timers clock .period

DDR_PCTL_TRSTH
Address: Operational Base + offset (0x00c8)
t_rsth Timing Register

Bit	Attr	Reset Value	Description
31:10	RO	0x0	reserved
9:0	RW	0x000	t_rsth Defines the time period (in us) to hold the dfi_reset_n signal high after it is de-asserted during the DDR3 Power Up/Reset sequence. The value programmed for DDR3 must correspond to minimum 500us of delay. For mDDR and DDR2, this register should be programmed to 0.The actual time period defined is TRSTH * TOGCNT1U * internal timers clock period.

DDR_PCTL_TOGCNT100N
Address: Operational Base + offset (0x00cc)
Toggle Counter 100ns

Bit	Attr	Reset Value	Description
31:7	RO	0x0	reserved
6:0	RW	0x01	toggle_counter_100n The number of internal timers clock cycles.

DDR_PCTL_TREFI
Address: Operational Base + offset (0x00d0)
t_refi Timing Register

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:0	RW	0x01	t_refi Defines the time period (in 100ns units) of the Refresh interval. The actual time period defined is TREFI * TOGCNT100N * internal timers clock period.

DDR_PCTL_TMRD
Address: Operational Base + offset (0x00d4)
t_mrd Timing Register

Bit	Attr	Reset Value	Description
31:3	RO	0x0	reserved
2:0	RW	0x1	t_mrd Mode Register Set command cycle time in memory clock cycles. mDDR: Time from MRS to any valid command. LPDDR2: Time from MRS (MRW) to any valid command. DDR2: Time from MRS to any valid command. DDR3: Time from MRS to MRS command. mDDR Legal Values: 2 LPDDR2 Legal Values: 5 DDR2 Legal Values: 2..3 DDR3 Legal Values: 2..4

DDR_PCTL_TRFC
Address: Operational Base + offset (0x00d8)

Bit	Attr	Reset Value	Description
31:9	RO	0x0	reserved
8:0	RW	0x001	t_rfc Refresh to Active/Refresh command time in memory clock cycles. mDDR Legal Values: 7..28 LPDDR2 Legal Values: 15..112 DDR2 Legal Values: 15..131 DDR3 Legal Values: 36.. 374

DDR_PCTL_TRP
Address: Operational Base + offset (0x00dc)
t_trp Timing Register

Bit	Attr	Reset Value	Description
31:18	RO	0x0	reserved

Bit	Attr	Reset Value	Description
17:16	RW	0x1	prea_extra Additional cycles required for a Precharge All (PREA) command - in addition to t_rp. In terms of memory clock cycles mDDR Value: 0 LPDDR2 Value: Value that corresponds (tRPab -tRPpb). Rounded up in terms of memory clock cycles. Values can be 0, 1, 2. DDR2 Value: 1 if 8 Banks, 0 otherwise DDR3 Value: 0
15:4	RO	0x0	reserved
3:0	RW	0x6	t_rp Precharge period in memory clock cycles. For LPDDR2, this should be set to TRPpb. mDDR Legal Values: 2..3 LPDDR2 Legal Values: 3..13 DDR2 Legal Values: 3..7 DDR3 Legal Values: 5..14

DDR_PCTL_TRTW
Address: Operational Base + offset (0x00e0)
t_rtw Timing Register

Bit	Attr	Reset Value	Description
31:4	RO	0x0	reserved
3:0	RW	0x2	t_rtw Read to Write turnaround time in memory clock cycles. mDDR Legal Values: 3..11 LPDDR2 Legal Values: 1..11 DDR2 Legal Values: 2..10 DDR3 Legal Values: 2..10

DDR_PCTL_TAL
Address: Operational Base + offset (0x00e4)
AL Register

Bit	Attr	Reset Value	Description
31:4	RO	0x0	reserved

Bit	Attr	Reset Value	Description
3:0	RW	0x0	t_al Additive Latency in memory clock cycles. For DDR2 this must match the value programmed into the AL field of MR1. For DDR3 this must be 0, CL-1, CL-2 depending weather the AL value in MR1 is 0,1, or 2 respectively. CL is the CAS latency programmed into MR0. For mDDR and LPDDR2, there is no AL field in the mode registers, and this setting should be set to 0 mDDR Legal Values: 0 LPDDR2 Legal Values: 0 DDR2 Legal Values: AL DDR3 Legal Values: 0, CL-1, CL-2 (depending on AL=0,1,2 in MR1)

DDR_PCTL_TCL
Address: Operational Base + offset (0x00e8)
CL Timing Register

Bit	Attr	Reset Value	Description
31:4	RO	0x0	reserved
3:0	RW	0x4	t_cl CAS Latency in memory clock cycles. If mDDR/DDR2/DDR3, the uPCTL setting must match the value programmed into the CL field of MR0. If LPDDR2, the uPCTL setting must match RL (ReadLatency), where RL is the value programmed into the "RL & W" field of MR2 mDDR/DDR2/3 Legal Value: CL LPDDR2 Legal Value: RL

DDR_PCTL_TCWL
Address: Operational Base + offset (0x00ec)
CWL Timing Register

Bit	Attr	Reset Value	Description
31:4	RO	0x0	reserved

Bit	Attr	Reset Value	Description
3:0	RW	0x3	t_cwl CAS Write Latency in memory clock cycles. For mDDR, the setting must be 1. For LPDDR2 the setting must match WL (Write Latency), where WL is the value programmed into the "RL & WL" field of MR2. For DDR2 the setting must match CL-1, where CL is the value programmed into the CL field of MR0. For DDR3, the setting must match the value programmed in the memory CWL field of MR2. mDDR Legal Value: 1 LPDDR2 Legal Values: WL DDR2 Legal Value: CL-1 DDR3 Legal Value: CWL

DDR_PCTL_TRAS
Address: Operational Base + offset (0x00f0)
t_ras Timing Register

Bit	Attr	Reset Value	Description
31:6	RO	0x0	reserved
5:0	RW	0x10	t_ras Activate to Precharge command time in memory clock cycles. mDDR Legal Values: 4..8 LPDDR2 Legal Values: 7..23 DDR2 Legal Values: 8..24 DDR3 Legal Values: 15..38

DDR_PCTL_TRC
Address: Operational Base + offset (0x00f4)
t_rc Timing Register

Bit	Attr	Reset Value	Description
31:6	RO	0x0	reserved
5:0	RW	0x16	t_rc Row Cycle time in memory clock cycles. Specifies the minimum Activate to Activate distance for accesses to same bank. mDDR Legal Values: 5..11 LPDDR2 Legal Values: 10..36 DDR2 Legal Values: 11..31 DDR3 Legal Values: 20..52

DDR_PCTL_TRCD
Address: Operational Base + offset (0x00f8)
t_rcd Timing Register

Bit	Attr	Reset Value	Description
31:4	RO	0x0	reserved

Bit	Attr	Reset Value	Description
3:0	RW	0x6	t_rcd Row to Column delay in memory clock cycles. Specifies the minimum Activate to Column distance. mDDR Legal Values: 2..3 LPDDR2 Legal Values: 3..13 DDR2 Legal Values: 3..7 DDR3 Legal Values: 5..14

DDR_PCTL_TRRD
Address: Operational Base + offset (0x00fc)
t_rrd Timing Register

Bit	Attr	Reset Value	Description
31:4	RO	0x0	reserved
3:0	RW	0x4	t_rrd Row-to-Row delay in memory clock cycles. Specifies the minimum Activate-to-Activate distance for consecutive accesses to different banks in the same rank. mDDR Legal Values: 1..2 LPDDR2 Legal Values: 2..6 DDR2 Legal Values: 2..6 DDR3 Legal Values: 4..8

DDR_PCTL_TRTP
Address: Operational Base + offset (0x0100)
t_rtp Timing Register

Bit	Attr	Reset Value	Description
31:4	RO	0x0	reserved
3:0	RW	0x3	t_rtp Read to Precharge time in memory clock cycles. Specifies the minimum distance Read to Precharge for consecutive accesses to same bank. mDDR Value: 0 LPDDR2 Legal Values: 2..4 DDR2 Legal Values: 2..4 DDR3 Legal Values: 3..8

DDR_PCTL_TWR
Address: Operational Base + offset (0x0104)
t_wr Register

Bit	Attr	Reset Value	Description
31:5	RO	0x0	reserved

Bit	Attr	Reset Value	Description
4:0	RW	0x06	t_wr Write recovery time in memory clock cycles. When using close page the uPCTL setting must be consistent with the WR field setting of MR0. mDDR Legal Values: 2..3 LPDDR2 Legal Values: 3..8 DDR2 Legal Values: 3..8 DDR3 Legal Values: 6..16

DDR_PCTL_TWTR
Address: Operational Base + offset (0x0108)
t_wtr Timing Register

Bit	Attr	Reset Value	Description
31:4	RO	0x0	reserved
3:0	RW	0x4	t_wtr Write to Read turnaround time, in memory clock cycles. mDDR Legal Values: 1..2 LPDDR2 Legal Values: 2..4 DDR2 Legal Values: 2..4 DDR3 Legal Values: 3..8

DDR_PCTL_TEXSR
Address: Operational Base + offset (0x010c)
t_exsr Timing Register

Bit	Attr	Reset Value	Description
31:10	RO	0x0	reserved
9:0	RW	0x001	t_exsr Exit Self Refresh to first valid command delay, in memory clock cycles. For mDDR, this should be programmed to match tXSR. For LPDDR2, this should be programmed to match tXSR. For DDR2, this should be programmed to match tXSRD (SRE to read-related command) as defined by the memory device specification. For DDR3, this should be programmed to match tXSDLL (SRE to a command requiring DLL locked) as defined by the memory device specification. mDDR Legal Values: 17..40 LPDDR2 Legal Values: 17..117 DDR2 Typical Value: 200 DDR3 Typical Value: 512

DDR_PCTL_TXP
Address: Operational Base + offset (0x0110)

t_xp Timing Register

Bit	Attr	Reset Value	Description
31:3	RO	0x0	reserved
2:0	RW	0x1	t_xp Exit Power Down to first valid command delay when DLL is on (fast exit), measured in memory clock cycles. Legal Values: 1..7

DDR_PCTL_TXPDLL
Address: Operational Base + offset (0x0114)
t_xpdll Timing Register

Bit	Attr	Reset Value	Description
31:6	RO	0x0	reserved
5:0	RW	0x00	t_xpdll Exit Power Down to first valid command delay when DLL is off (slow exit), measured in memory clock cycles. mDDR/LPDDR2 Value: 0 DDR2/DDR3 Legal Values: 3..63

DDR_PCTL_TZQCS
Address: Operational Base + offset (0x0118)
t_zqcs Timing Register

Bit	Attr	Reset Value	Description
31:7	RO	0x0	reserved
6:0	RW	0x00	t_zqcs SDRAM ZQ Calibration Short period, in memory clock cycles. Should be programmed to match the tZQCS timing value as defined in the memory specification. mDDR Value: 0 LPDDR2 Legal Values: 15..48 DDR2 Value: 0 DDR3 Typical Value: 64

DDR_PCTL_TZQCSI
Address: Operational Base + offset (0x011c)
t_zqcsi Timing Register

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	t_zqcsi SDRAM ZQCS interval, measured in Refresh interval units. The total time period defined is TZQCSI*TREFI * TOGCNT100N * internal timers clock period. Programming a value of 0 in t_zqcsi disables the auto-ZQCS functionality in uPCTL. mDDR Value: 0 LPDDR2 Legal Values: 0..4294967295 DDR2 Value: 0 DDR3 Legal Values: 0..4294967295

DDR_PCTL_TDQS
Address: Operational Base + offset (0x0120)
t_dqs Timing Register

Bit	Attr	Reset Value	Description
31:3	RO	0x0	reserved
2:0	RW	0x1	t_dqs Additional data turnaround time in memory clock cycles for accesses to different ranks. Used to increase the distance between column commands to different ranks, allowing more tolerance as the driver source changes on the bidirectional DQS and/or DQ signals. mDDR Legal Values: 1..7 LPDDR2 Legal Values: 1..7 DDR2 Legal Values: 1..7 DDR3 Legal Values: 1..7

DDR_PCTL_TCKSRE
Address: Operational Base + offset (0x0124)
t_cksre Timing Register

Bit	Attr	Reset Value	Description
31:5	RO	0x0	reserved
4:0	RW	0x00	t_cksre In DDR3, this is the time after Self Refresh Entry that CKE is held high before going low. In memory clock cycles. Specifies the clock disable delay after SRE. This should be programmed to match the greatest value between 10ns and 5 memory clock periods. mDDR Value: 0 LPDDR2 Value: 0 DDR2 Value: 0 DDR3 Legal Values: 5..15

DDR_PCTL_TCKSRX
Address: Operational Base + offset (0x0128)
t_cksrx Timing Register

Bit	Attr	Reset Value	Description
31:5	RO	0x0	reserved

Bit	Attr	Reset Value	Description
4:0	RW	0x00	t_cksrx In DDR3, this is the time (before Self Refresh Exit) that CKE is maintained high before issuing SRX. In memory clock cycles. Specifies the clock stable time before SRX. This should be programmed to match the greatest value between 10ns and 5 memory clock periods. mDDR Value: 0 LPDDR2 Value: 0 DDR2 Value: 0 DDR3 Legal Values: 5..15

DDR_PCTL_TCKE
Address: Operational Base + offset (0x012c)
t_cke Timing Register

Bit	Attr	Reset Value	Description
31:3	RO	0x0	reserved
2:0	RW	0x3	t_cke CKE minimum pulse width in memory clock cycles. mDDR Legal Value: 2 LPDDR2 Legal Values: 3 DDR2 Legal Value: 3 DDR3 Legal Values: 3..6

DDR_PCTL_TMOD
Address: Operational Base + offset (0x0130)
t_mod Timing Register

Bit	Attr	Reset Value	Description
31:5	RO	0x0	reserved
4:0	RW	0x00	t_mod In DDR3 mode, this is the time from MRS to any valid non-MRS command (except DESELECT or NOP) in memory clock cycles. mDDR Value: 0 LPDDR2 Value: 0 DDR2 Value: 0 DDR3 Legal Values: 0..31

DDR_PCTL_TRSTL
Address: Operational Base + offset (0x0134)
Reset Low Timing Register

Bit	Attr	Reset Value	Description
31:7	RO	0x0	reserved

Bit	Attr	Reset Value	Description
6:0	RW	0x00	t_rstl Memory Reset Low time, in memory clock cycles. Defines the time period to hold dfi_reset_n signal low during a software driven DDR3 Reset Operation. The value programmed must correspond to at least 100ns of delay. mDDR Value: 0 LPDDR2 Value: 0 DDR2 Value: 0 DDR3 Legal Values: 1..127

DDR_PCTL_TZQCL
Address: Operational Base + offset (0x0138)
t_zqcl Timing Register

Bit	Attr	Reset Value	Description
31:10	RO	0x0	reserved
9:0	RW	0x000	t_zqcl SDRAM ZQ Calibration Long period in memory clock cycles. If LPDDR2, should be programmed to tZQCL. If DDR3, should be programmed to match the memory tZQinit timing value for the first ZQCL command during memory initialization; should be programmed to match tZQoper timing value after reset and initialization. mDDR Value: 0 LPDDR2 Legal Values: 60..192 DDR2 Value: 0 DDR3 Legal Values: 0..1023

DDR_PCTL_TMRR
Address: Operational Base + offset (0x013c)
t_mrr Timing Register

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:0	RW	0x02	t_mrr Time for a Mode Register Read (MRR command from MCMD).

DDR_PCTL_TCKESR
Address: Operational Base + offset (0x0140)
t_ckesr Timing Register

Bit	Attr	Reset Value	Description
31:4	RO	0x0	reserved

Bit	Attr	Reset Value	Description
3:0	RW	0x4	t_ckesr Minimum CKE low width for Self Refresh entry to exit timing in memory clock cycles. Recommended settings: - mDDR : t_ckesr = 0 - LPDDR2 : t_ckesr = tCKESR setting from memories, rounded up in terms of memory cycles. - DDR2 : t_ckesr = 0 - DDR3 : t_ckesr = t_cke + 1 mDDR Value: 0 LPDDR2 Legal Values: 3..8 DDR2 Value: 0 DDR3 Legal Values: 4..7

DDR_PCTL_TDPD
Address: Operational Base + offset (0x0144)
t_dpd Timing Register

Bit	Attr	Reset Value	Description
31:10	RO	0x0	reserved
9:0	RW	0x000	t_dpd Minimum Deep Power Down time. Is in terms of us. When a MCMD.DPDE command occurs, TDPD time is waited before MCMD.start_cmd can be cleared. MCMD_cmd_add_del (if any) does not start until TDPD has completed. This ensures TDPD requirement for the memory is not violated. The actual time period defined is TDPD* TOGCNT1U * internal timers clock period. Only applies for mDDR and LPDDR2 as Deep Power Down (DPD) is only valid for these memory types. For mDDR, tDPD=0, while for LPDDR2, tDPD=500 us. For LPDDR2, if 500 us is waited externally by system, then set tDPD=0. mDDR Value: 0 LPDDR2 Legal Values: 0 or 500 DDR2 Legal Value: 0 DDR3 Legal Values: 0

DDR_PCTL_DTUWACTL
Address: Operational Base + offset (0x0200)
DTU Write Address Control

Bit	Attr	Reset Value	Description
31:30	RW	0x0	dtu_wr_rank Write rank to where data is to be targeted
29	RO	0x0	reserved
28:13	RW	0x0000	dtu_wr_row Write row to where data is to be targeted

Bit	Attr	Reset Value	Description
12:10	RW	0x0	dtu_wr_bank Write bank to where data is to be targeted
9:0	RW	0x000	dtu_wr_col FWrite column to where data is to be targeted

DDR_PCTL_DTURACTL
Address: Operational Base + offset (0x0204)
DTU Read Address Control Register

Bit	Attr	Reset Value	Description
31:30	RW	0x0	dtu_rd_rank Read rank from where data comes
29	RO	0x0	reserved
28:13	RW	0x0000	dtu_rd_row Read row from where data comes
12:10	RW	0x0	dtu_rd_bank Read bank from where data comes
9:0	RW	0x000	dtu_rd_col Read column from where data comes

DDR_PCTL_DTUCFG
Address: Operational Base + offset (0x0208)
DTU Configuration Control Register

Bit	Attr	Reset Value	Description
31:23	RO	0x0	reserved
22:16	RW	0x00	dtu_row_increments Number of times to increment the row address when generating random data, up to a maximum of 127 times.
15	RW	0x0	dtu_wr_multi_rd When set puts the DTU into write once multiple reads mode.
14	RW	0x0	dtu_data_mask_en Controls whether random generated data masks are transmitted. Unless enabled all data bytes are written to memory and expected to be read from memory.
13:10	RW	0x0	dtu_target_lane Selects one of the byte lanes for data comparison into the programmable read data buffer.
9	RW	0x0	dtu_generate_random Generate transfers using random data, otherwise generate transfers from the programmable write data buffers.
8	RW	0x0	dtu_incr_banks When the column address rolls over increment the bank address until we reach and conclude bank 7.
7	RW	0x0	dtu_incr_cols Increment the column address until we saturate. Return to zero if DTUCFG.dtu_incr_banks is set to 1 and we are not at bank 7.

Bit	Attr	Reset Value	Description
6:1	RW	0x00	dtu_nalen Length of the NIF transfer sequence that is passed through the uPCTL for each created address.
0	RW	0x0	dtu_enable When set, allows the DTU module to take ownership of the NIF interface: 1: DTU enabled 0: DTU disabled

DDR_PCTL_DTUECTL
Address: Operational Base + offset (0x020c)
DTU Execute Control Register

Bit	Attr	Reset Value	Description
31:3	RO	0x0	reserved
2	R/WSC	0x0	wr_multi_rd_rst When set, resets the DTU in write once multiple reads mode, to allow a new write to be performed. This bit automatically clears.
1	R/WSC	0x0	run_error_reports When set, initiates the calculation of the error status bits. This bit automatically clears when the re-calculation is done. This is only used in debug mode to verify the comparison logic.
0	R/WSC	0x0	run_dtu When set, initiates the running of the DTU read and write transfer. This bit automatically clears when the transfers are completed

DDR_PCTL_DTUWD0
Address: Operational Base + offset (0x0210)
DTU Write Data #0 Register

Bit	Attr	Reset Value	Description
31:24	RW	0x00	dtu_wr_byte3 Write data byte
23:16	RW	0x00	dtu_wr_byte2 Write data byte
15:8	RW	0x00	dtu_wr_byte1 Write data byte
7:0	RW	0x00	dtu_wr_byte0 Write data byte

DDR_PCTL_DTUWD1
Address: Operational Base + offset (0x0214)
DTU Write Data #1 Register

Bit	Attr	Reset Value	Description
31:24	RW	0x00	dtu_wr_byte7 Write data byte

Bit	Attr	Reset Value	Description
23:16	RW	0x00	dtu_wr_byte6 Write data byte
15:8	RW	0x00	dtu_wr_byte5 Write data byte
7:0	RW	0x00	dtu_wr_byte4 Write data byte

DDR_PCTL_DTUWD2
Address: Operational Base + offset (0x0218)
DTU Write Data #2 Register

Bit	Attr	Reset Value	Description
31:24	RW	0x00	dtu_wr_byte11 Write data byte
23:16	RW	0x00	dtu_wr_byte10 Write data byte
15:8	RW	0x00	dtu_wr_byte9 Write data byte
7:0	RW	0x00	dtu_wr_byte8 Write data byte

DDR_PCTL_DTUWD3
Address: Operational Base + offset (0x021c)
DTU Write Data #3 Register

Bit	Attr	Reset Value	Description
31:24	RW	0x00	dtu_wr_byte15 Write data byte
23:16	RW	0x00	dtu_wr_byte14 Write data byte
15:8	RW	0x00	dtu_wr_byte13 Write data byte
7:0	RW	0x00	dtu_wr_byte12 Write data byte

DDR_PCTL_DTUWDM
Address: Operational Base + offset (0x0220)
DTU Write Data Mask Register

Bit	Attr	Reset Value	Description
31:16	RO	0x0	reserved
15:0	RW	0x0000	dm_wr_byte0 Write data mask bit, one bit for each byte. Each bit should be 0 for a byte lane that contains valid write data.

DDR_PCTL_DTURD0
Address: Operational Base + offset (0x0224)
DTU Read Data #0 Register

Bit	Attr	Reset Value	Description
31:24	RO	0x00	dtu_rd_byte3 Read byte
23:16	RO	0x00	dtu_rd_byte2 Read byte
15:8	RO	0x00	dtu_rd_byte1 Read byte
7:0	RO	0x00	dtu_rd_byte0 Read byte

DDR_PCTL_DTURD1
Address: Operational Base + offset (0x0228)
DTU Read Data #1 Register

Bit	Attr	Reset Value	Description
31:24	RO	0x00	dtu_rd_byte7 Read byte
23:16	RO	0x00	dtu_rd_byte6 Read byte
15:8	RO	0x00	dtu_rd_byte5 Read byte
7:0	RO	0x00	dtu_rd_byte4 Read byte

DDR_PCTL_DTURD2
Address: Operational Base + offset (0x022c)
DTU Read Data #2 Register

Bit	Attr	Reset Value	Description
31:24	RO	0x00	dtu_rd_byte11 Read byte
23:16	RO	0x00	dtu_rd_byte10 Read byte
15:8	RO	0x00	dtu_rd_byte9 Read byte
7:0	RO	0x00	dtu_rd_byte8 Read byte

DDR_PCTL_DTURD3
Address: Operational Base + offset (0x0230)
DTU Read Data #3 Register

Bit	Attr	Reset Value	Description
31:24	RO	0x00	dtu_rd_byte15 Read byte
23:16	RO	0x00	dtu_rd_byte14 Read byte
15:8	RO	0x00	dtu_rd_byte13 Read byte

Bit	Attr	Reset Value	Description
7:0	RO	0x00	dtu_rd_byte12 Read byte

DDR_PCTL_DTULFSRWD
Address: Operational Base + offset (0x0234)
DTU LFSR Seed for Write Data Generation Register

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	dtu_lfsr_wseed This is the initial seed for the random write data generation LFSR (linear feedback shift register), shared with the write mask generation.

DDR_PCTL_DTULFSRRD
Address: Operational Base + offset (0x0238)
DTU LFSR Seed for Read Data Generation Register

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	dtu_lfsr_rseed This is the initial seed for the random read data generation LFSR (linear feedback shift register), this is shared with the read mask generation. The read data mask is reconstructed the same as the write data mask was created, allowing the " on the fly comparison" ignore bytes which were not written.

DDR_PCTL_DTUEAF
Address: Operational Base + offset (0x023c)
DTU Error Address FIFO Register

Bit	Attr	Reset Value	Description
31:30	RO	0x0	ea_rank Indicates the rank that the error occurred in during random data generation. There could be a number of entries in this FIFO. If FIFO is empty one reads zeroes.
29	RO	0x0	reserved
28:13	RO	0x0000	ea_row Indicates the row that the error occurred in during random data generation. There could be a number of entries in this FIFO. If FIFO is empty one reads zeroes.
12:10	RO	0x0	ea_bank Indicates the bank that the error occurred in during random data generation. There could be a number of entries in this FIFO. If FIFO is empty one reads zeroes
9:0	RO	0x000	ea_column Indicates the column address that the error occurred in during random data generation. There could be a number of entries in this FIFO. If FIFO is empty one reads zeroes.

DDR_PCTL_DFITCTRLDELAY
Address: Operational Base + offset (0x0240)
DFI tctrl_delay Register

Bit	Attr	Reset Value	Description
31:4	RO	0x0	reserved
3:0	RW	0x2	tctrl_delay Specifies the number of DFI clock cycles after an assertion or deassertion of the DFI control signals that the control signals at the PHY-DRAM interface reflect the assertion or de-assertion. If the DFI clock and the memory clock are not phase-aligned, this timing parameter should be rounded up to the next integer value.

DDR_PCTL_DFIODTCFG
Address: Operational Base + offset (0x0244)
DFI ODT Configuration

Bit	Attr	Reset Value	Description
31:29	RO	0x0	reserved
28	RW	0x0	rank3_odt_default Default ODT value of rank 3 when there is no read/write activity
27	RW	0x0	rank3_odt_write_sel Enable/disable ODT for rank 3 when a write access is occurring on this rank
26	RW	0x0	rank3_odt_write_nse Enable/disable ODT for rank 3 when a write access is occurring on a different rank
25	RW	0x0	rank3_odt_read_sel Enable/disable ODT for rank 3 when a read access is occurring on this rank
24	RW	0x0	rank3_odt_read_nsel Enable/disable ODT for rank 3 when a read access is occurring on a different rank
23:21	RO	0x0	reserved
20	RW	0x0	rank2_odt_default Default ODT value of rank 2 when there is no read/write activity
19	RW	0x0	rank2_odt_write_sel Enable/disable ODT for rank 2 when a write access is occurring on this rank
18	RW	0x0	rank2_odt_write_nse Enable/disable ODT for rank 2 when a write access is occurring on a different rank
17	RW	0x0	rank2_odt_read_sel Enable/disable ODT for rank 2 when a read access is occurring on this rank
16	RW	0x0	rank2_odt_read_nsel Enable/disable ODT for rank 2 when a read access is occurring on a different rank

Bit	Attr	Reset Value	Description
15:13	RO	0x0	reserved
12	RW	0x0	rank1_odt_default Default ODT value of rank 1 when there is no read/write activity
11	RW	0x0	rank1_odt_write_sel Enable/disable ODT for rank 1 when a write access is occurring on this rank
10	RW	0x0	rank1_odt_write_nse Enable/disable ODT for rank 1 when a write access is occurring on a different rank
9	RW	0x0	rank1_odt_read_sel Enable/disable ODT for rank 1 when a read access is occurring on this rank
8	RW	0x0	rank1_odt_read_nsel Enable/disable ODT for rank 1 when a read access is occurring on a different rank
7:5	RO	0x0	reserved
4	RW	0x0	rank0_odt_default Default ODT value of rank 0 when there is no read/write activity
3	RW	0x0	rank0_odt_write_sel Enable/disable ODT for rank 0 when a write access is occurring on this rank
2	RW	0x0	rank0_odt_write_nse Enable/disable ODT for rank 0 when a write access is occurring on a different rank
1	RW	0x0	rank0_odt_read_sel Enable/disable ODT for rank 0 when a read access is occurring on this rank
0	RW	0x0	rank0_odt_read_nsel Enable/disable ODT for rank 0 when a read access is occurring on a different rank

DDR_PCTL_DFIODTCFG1

Address: Operational Base + offset (0x0248)

DFI ODT Timing Configuration 1 (for Latency and Length)

Bit	Attr	Reset Value	Description
31:27	RO	0x0	reserved
26:24	RW	0x6	odt_len_bl8_r ODT length for BL8 read transfers Length of dfi_odt signal for BL8 reads. This is in terms of SDR cycles. For BL4 reads, the length of dfi_odt is always 2 cycles shorter than the value in this register field.
23:19	RO	0x0	reserved

Bit	Attr	Reset Value	Description
18:16	RW	0x6	odt_len_bl8_w ODT length for BL8 write transfers Length of dfi_odt signal for BL8 writes. This is in terms of SDR cycles. For BL4 writes, the length of dfi_odt is always 2 cycles shorter than the value in this register field.
15:13	RO	0x0	reserved
12:8	RW	0x00	odt_lat_r Field0000 Abstract ODT latency for reads Latency after a read command that dfi_odt is set. This is in terms of SDR cycles.
7:5	RO	0x0	reserved
4:0	RW	0x00	odt_lat_w ODT latency for writes Latency after a write command that dfi_odt is set. This is in terms of SDR cycles

DDR_PCTL_DFIODTRANKMAP
Address: Operational Base + offset (0x024c)
DFI ODT Rank Mapping

Bit	Attr	Reset Value	Description
31:16	RO	0x0	reserved
15:12	RW	0x8	odt_rank_map3 Rank mapping for dfi_odt[3] Determines whether dfi_odt[3] should be asserted when the uPCTL requires to terminate each rank Bit 15 = 1: dfi_odt[3] will be asserted to terminate rank 3 Bit 14 = 1: dfi_odt[3] will be asserted to terminate rank 2 Bit 13 = 1: dfi_odt[3] will be asserted to terminate rank 1 Bit 12 = 1: dfi_odt[3] will be asserted to terminate rank 0 This field exists only if UPCTL_M_NRANKS = 4
11:8	RW	0x4	odt_rank_map2 Rank mapping for dfi_odt[2] Determines which rank access(es) will cause dfi_odt[2] to be asserted Bit 11 = 1: dfi_odt[2] will be asserted to terminate rank 3 Bit 10 = 1: dfi_odt[2] will be asserted to terminate rank 2 Bit 9 = 1: dfi_odt[2] will be asserted to terminate rank 1 Bit 8 = 1: dfi_odt[2] will be asserted to terminate rank 0 This field exists only if UPCTL_M_NRANKS = 4

Bit	Attr	Reset Value	Description
7:4	RW	0x2	odt_rank_map1 Rank mapping for dfi_odt[1] Determines which rank access(es) will cause dfi_odt[1] to be asserted Bit 7= 1: dfi_odt[1] will be asserted to terminate rank 3 Bit 6= 1: dfi_odt[1] will be asserted to terminate rank 2 Bit 5= 1: dfi_odt[1] will be asserted to terminate rank 1 Bit 4= 1: dfi_odt[1] will be asserted to terminate rank 0 This field exists only if UPCTL_M_NRANKS >
3:0	RW	0x1	odt_rank_map0 Rank mapping for dfi_odt[0] Determines which rank access(es) will cause dfi_odt[0] to be asserted Bit 3= 1: dfi_odt[0] will be asserted to terminate rank 3 Bit 2= 1: dfi_odt[0] will be asserted to terminate rank 2 Bit 1= 1: dfi_odt[0] will be asserted to terminate rank 1 Bit 0= 1: dfi_odt[0] will be asserted to terminate rank 0

DDR_PCTL_DFITPHYWRDATA
Address: Operational Base + offset (0x0250)
DFI tphy_wrdata Register

Bit	Attr	Reset Value	Description
31:6	RO	0x0	reserved
5:0	RW	0x01	tphy_wrdata Specifies the number of DFI clock cycles between when the dfi_wrdata_en signal is asserted to when the associated write data is driven on the dfi_wrdata signal. This has no impact on performance, only adjusts the relative time between enable and data transfer.

DDR_PCTL_DFITPHYWRLAT
Address: Operational Base + offset (0x0254)
DFI tphy_wrlat Register

Bit	Attr	Reset Value	Description
31:6	RO	0x0	reserved
5:0	RW	0x01	tphy_wrlat Specifies the number of DFI clock cycles between when a write command is sent on the DFI control interface and when the dfi_wrdata_en signal is asserted.

DDR_PCTL_DFITRDDATAEN
Address: Operational Base + offset (0x0260)
DFI trddata_en Register

Bit	Attr	Reset Value	Description
31:6	RO	0x0	reserved

Bit	Attr	Reset Value	Description
5:0	RW	0x01	trddata_en Specifies the number of DFI clock cycles from the assertion of a read command on the DFI to the assertion of the dfi_rddata_en signal.

DDR_PCTL_DFITPHYRDLAT
Address: Operational Base + offset (0x0264)
DFI tphy_rdlat Register

Bit	Attr	Reset Value	Description
31:6	RO	0x0	reserved
5:0	RW	0x0f	tphy_rdlat Specifies the maximum number of DFI clock cycles allowed from the assertion of the dfi_rddata_en signal to the assertion of the dfi_rddata_valid signal.

DDR_PCTL_DFITPHYUPDTYPE0
Address: Operational Base + offset (0x0270)
DFI tphyupd_type0 Register

Bit	Attr	Reset Value	Description
31:12	RO	0x0	reserved
11:0	RW	0x010	tphyupd_type0 Specifies the maximum number of DFI clock cycles that the dfi_phyupd_req signal may remain asserted after the assertion of the dfi_phyupd_ack signal for dfi_phyupd_type = 0x0. The dfi_phyupd_req signal may de-assert at any cycle after the assertion of the dfi_phyupd_ack signal.

DDR_PCTL_DFITPHYUPDTYPE1
Address: Operational Base + offset (0x0274)
DFI tphyupd_type1 Register

Bit	Attr	Reset Value	Description
31:12	RO	0x0	reserved
11:0	RW	0x010	tphyupd_type1 Specifies the maximum number of DFI clock cycles that the dfi_phyupd_req signal may remain asserted after the assertion of the dfi_phyupd_ack signal for dfi_phyupd_type = 0x1. The dfi_phyupd_req signal may de-assert at any cycle after the assertion of the dfi_phyupd_ack signal.

DDR_PCTL_DFITPHYUPDTYPE2
Address: Operational Base + offset (0x0278)
DFI tphyupd_type2 Register

Bit	Attr	Reset Value	Description
31:12	RO	0x0	reserved

Bit	Attr	Reset Value	Description
11:0	RW	0x010	tphyupd_type2 Specifies the maximum number of DFI clock cycles that the dfi_phyupd_req signal may remain asserted after the assertion of the dfi_phyupd_ack signal for dfi_phyupd_type = 0x2. The dfi_phyupd_req signal may de-assert at any cycle after the assertion of the dfi_phyupd_ack signal.

DDR_PCTL_DFITPHYUPDTYPE3
Address: Operational Base + offset (0x027c)
DFI tphyupd_type3 Register

Bit	Attr	Reset Value	Description
31:12	RO	0x0	reserved
11:0	RW	0x010	tphyupd_type3 Specifies the maximum number of DFI clock cycles that the dfi_phyupd_req signal may remain asserted after the assertion of the dfi_phyupd_ack signal for dfi_phyupd_type = 0x3. The dfi_phyupd_req signal may de-assert at any cycle after the assertion of the dfi_phyupd_ack signal.

DDR_PCTL_DFITCTRLUPDMIN
Address: Operational Base + offset (0x0280)
DFI tctrlupd_min Register

Bit	Attr	Reset Value	Description
31:16	RO	0x0	reserved
15:0	RW	0x0010	tctrlupd_min Specifies the minimum number of DFI clock cycles that the dfi_ctrlupd_req signal must be asserted.

DDR_PCTL_DFITCTRLUPDMAX
Address: Operational Base + offset (0x0284)
DFI tctrlupd_max Register

Bit	Attr	Reset Value	Description
31:16	RO	0x0	reserved
15:0	RW	0x0040	tctrlupd_max Specifies the maximum number of DFI clock cycles that the dfi_ctrlupd_req signal can assert.

DDR_PCTL_DFITCTRLUPDDLY
Address: Operational Base + offset (0x0288)
DFI tctrlupddly Register

Bit	Attr	Reset Value	Description
31:4	RO	0x0	reserved

Bit	Attr	Reset Value	Description
3:0	RW	0x8	tctrlupd_dly Delay in DFI clock cycles between time a uPCTL-initiated update could be started and time uPCTL-initated update actually starts (dfi_ctrlupd_req going high).

DDR_PCTL_DFIUPDCFG
Address: Operational Base + offset (0x0290)
DFI Update Configuration Register

Bit	Attr	Reset Value	Description
31:2	RO	0x0	reserved
1	RW	0x1	dfi_phyupd_en Enables the support for acknowledging PHY-initiated updates: 1'b0 = Disabled 1'b1 = Enabled
0	RW	0x1	dfi_ctrlupd_en Enables the generation of uPCTL-initiated updates: 1'b0 = Disabled 1'b1 = Enabled

DDR_PCTL_DFITREFMSKI
Address: Operational Base + offset (0x0294)
DFI Masked Refresh Interval

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:0	RW	0x00	trefmski Time period of the masked Refresh interva This value is only used if TREFI==0. Defines the time period (in 100ns units) of the masked Refresh (REFMSK) interval. The actual time period defined is DFITREFMSKI* TOGCNT100N * internal timers clock period.

DDR_PCTL_DFITCTRLUPDI
Address: Operational Base + offset (0x0298)
DFI tctrlupd_interval Register

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	tctrlupd_interval DFI uPCTL-initiated updates interval, measured in terms of Refresh interval units. If TREFI!=0, the time period is defined as DFITCTRLUPDI*TREFI * TOGCNT100N * internal timers clock period. If TREFI==0 and DFITREFMSKI!=0, the period changes to DFITCTRLUPDI*DFITREFMSKI* * TOGCNT100N * internal timers clock period. Programming a value of 0 is the same as programming a value of 1; for instance, a uPCTL-initiated update occurs every Refresh interval.

DDR_PCTL_DFITRCFG0
Address: Operational Base + offset (0x02ac)
DFI Training Configuration 0 Register

Bit	Attr	Reset Value	Description
31:20	RO	0x0	reserved
19:16	RW	0x0	dfi_wrlvl_rank_sel Determines the value to drive on the output signal dfi_wrlvl_cs_n. The value on dfi_wrlvl_cs_n is the inverse of the setting in this field.
15:13	RO	0x0	reserved
12:4	RW	0x000	dfi_rdlvl_edge Determines the value to drive on the output signal dfi_rdlvl_edge. The value on dfi_rdlvl_edge is the same as the setting in this field.
3:0	RW	0x0	dfi_rdlvl_rank_sel Determines the value to drive on the output signal dfi_rdlvl_cs_n. The value on dfi_rdlvl_cs_n is the inverse of the setting in this field.

DDR_PCTL_DFITRSTAT0
Address: Operational Base + offset (0x02b0)
DFI Training Status 0 Register

Bit	Attr	Reset Value	Description
31:18	RO	0x0	reserved
17:16	RO	0x0	dfi_wrlvl_mode Reports the value of the input signal dfi_wrlvl_mode.
15:10	RO	0x0	reserved
9:8	RO	0x0	dfi_rdlvl_gate_mode Reports the value of the input signal dfi_rdlvl_gate_mode.
7:2	RO	0x0	reserved
1:0	RO	0x0	dfi_rdlvl_mode Reports the value of the input signal dfi_rdlvl_mode.

DDR_PCTL_DFITRWRLVLEN
Address: Operational Base + offset (0x02b4)
DFI Training dfi_wrlvl_en Register

Bit	Attr	Reset Value	Description
31:9	RO	0x0	reserved
8:0	RW	0x000	dfi_wrlvl_en Determines the value to drive on the output signal dfi_wrlvl_en.

DDR_PCTL_DFITRRDLVLEN
Address: Operational Base + offset (0x02b8)
DFI Training dfi_rdlvl_en Register

Bit	Attr	Reset Value	Description
31:9	RO	0x0	reserved
8:0	RW	0x000	dfi_rdlvl_en Determines the value to drive on the output signal dfi_rdlvl_en.

DDR_PCTL_DFITRRDLVLGATEEN
Address: Operational Base + offset (0x02bc)
DFI Training dfi_rdlvl_gate_en Register

Bit	Attr	Reset Value	Description
31:9	RO	0x0	reserved
8:0	RW	0x000	dfi_rdlvl_gate_en Determines the value to drive on the output signal dfi_rdlvl_gate_en.

DDR_PCTL_DFISTSTAT0
Address: Operational Base + offset (0x02c0)
DFI Status Status 0 Register

Bit	Attr	Reset Value	Description
31:25	RO	0x0	reserved
24:16	RO	0x000	dfi_data_byte_disable Reports the value of the output signal dfi_data_byte_disable.
15:6	RO	0x0	reserved
5:4	RO	0x0	dfi_freq_ratio Reports the value of the output signal dfi_freq_ratio.
3:2	RO	0x0	reserved
1	RO	0x0	dfi_init_start Reports the value of the output signal dfi_init_start.
0	RO	0x0	dfi_init_complete Reports the value of the input signal dfi_init_complete.

DDR_PCTL_DFISTCFG0
Address: Operational Base + offset (0x02c4)
DFI Status Configuration 0 Register

Bit	Attr	Reset Value	Description
31:3	RO	0x0	reserved
2	RW	0x0	dfi_data_byte_disable_en Enables the driving of the dfi_data_byte_disable signal. The value driven on dfi_data_byte_disable is dependent on the setting of PPCFG register. 1'b0 - Drive dfi_data_byte_disable to default value of all zeroes. 1'b1 - Drive dfi_data_byte_disable according to value as defined by PPCFG register setting. Note: should be set to 1only after PPCFG is correctly set.
1	RW	0x0	dfi_freq_ratio_en Enables the driving of the dfi_freq_ratio signal. When enabled, the dfi_freq_ratio value driven is dependent on configuration parameter UPCTL_FREQ_RATIO: 2纰is driven when UPCTL_FREQ_RATIO=1; 2'b01 is driven when UPCTL_FREQ_RATIO=2. 1'b0 - Drive dfi_freq_ratio to default value of 2'b00. 1'b1 - Drive dfi_freq_ratio value according to how configuration parameter is set.
0	RW	0x0	dfi_init_start Sets the value of the dfi_init_start signal. 1'b0 - dfi_init_start is driven low 1'b1 - dfi__init_start is driven high

DDR_PCTL_DFISTCFG1
Address: Operational Base + offset (0x02c8)
DFI Status Configuration 1 Register

Bit	Attr	Reset Value	Description
31:2	RO	0x0	reserved
1	RW	0x0	dfi_dram_clk_disable_en_dpd Enables support of the dfi_dram_clk_disable signal with Deep Power Down (DPD). DPD is only for mDDR/LPDDR2. 1'b0 - Disable dfi_dram_clk_disable support in relation to DPD 1'b1 - Enable dfi_dram_clk_disable support in relation to DPD
0	RW	0x0	dfi_dram_clk_disable_en Enables support of the dfi_dram_clk_disable signal with Self Refresh (SR). 1'b0 - Disable dfi_dram_clk_disable support in relation to SR 1'b1 - Enable dfi_dram_clk_disable support in relation to SR

DDR_PCTL_DFITDRAMCLKEN
Address: Operational Base + offset (0x02d0)
DFI tdram_clk_enable Register

Bit	Attr	Reset Value	Description
31:4	RO	0x0	reserved

Bit	Attr	Reset Value	Description
3:0	RW	0x2	tdram_clk_enable Specifies the number of DFI clock cycles from the de-assertion of the dfi_dram_clk_disable signal on the DFI until the first valid rising edge of the clock to the DRAM memory devices, at the PHY-DRAM boundary. If the DFI clock and the memory clock are not phase aligned, this timing parameter should be rounded up to the next integer value.

DDR_PCTL_DFITDRAMCLKDIS
Address: Operational Base + offset (0x02d4)
DFI tdram_clk_disable Register

Bit	Attr	Reset Value	Description
31:4	RO	0x0	reserved
3:0	RW	0x2	tdram_clk_disable Specifies the number of DFI clock cycles from the assertion of the dfi_dram_clk_disable signal on the DFI until the clock to the DRAM memory devices, at the PHY-DRAM boundary, maintains a low value. If the DFI clock and the memory clock are not phase aligned, this timing parameter should be rounded up to the next integer value.

DDR_PCTL_DFISTCFG2
Address: Operational Base + offset (0x02d8)
DFI Status Configuration 2 Register

Bit	Attr	Reset Value	Description
31:2	RO	0x0	reserved
1	RW	0x0	parity_en Enables the DFI parity generation feature (driven on output signal dfi_parity_in) 1'b0 - Disable DFI parity generation 1'b1 - Enable DFI parity generation
0	RW	0x0	parity_intr_en Enable interrupt generation for DFI parity error (from input signal dfi_parity_error). 1'b0 - Disable interrupt 1'b1 - Enable interrupt

DDR_PCTL_DFISTPARCLR
Address: Operational Base + offset (0x02dc)
DFI Status Parity Clear Register

Bit	Attr	Reset Value	Description
31:2	RO	0x0	reserved

Bit	Attr	Reset Value	Description
1	R/W SC	0x0	parity_log_clr Set this bit to 1'b1 to clear the DFI Status Parity Log register (DFISTPARLOG). 1'b0 = Do not clear DFI status Parity Log register 1'b1 = Clear DFI status Parity Log register
0	R/W SC	0x0	parity_intr_clr Set this bit to 1'b1 to clear the interrupt generated by an DFI parity error (as enabled by DFISTCFG2.parity_intr_en). It also clears the INTRSTAT.parity_intr register field. It is automatically cleared by hardware when the interrupt has been cleared.

DDR_PCTL_DFISTPARLOG
Address: Operational Base + offset (0x02e0)
DFI Status Parity Log Register

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	parity_err_cnt Increments any time the DFI parity logic detects a parity error(s) (on dfi_parity_error).

DDR_PCTL_DFILPCFG0
Address: Operational Base + offset (0x02f0)
DFI Low Power Configuration 0 Register

Bit	Attr	Reset Value	Description
31:28	RW	0x0	dfi_lp_wakeup_dpd Value to drive on dfi_lp_wakeup signal when Deep Power Down mode is entered. Determines the DFI's tlp_wakeup time: 4'b0000 - 16 cycles 4'b0001 - 32 cycles 4'b0010 - 64 cycles 4'b0011 - 128 cycles 4'b0100 - 256 cycles 4'b0101 - 512 cycles 4'b0110 - 1024 cycles 4'b0111 - 2048 cycles 4'b1000 - 4096 cycles 4'b1001 - 8192 cycles 4'b1010 - 16384 cycles 4'b1011 - 32768 cycles 4'b1100 - 65536 cycles 4'b1101 - 131072 cycles 4'b1110 - 262144 cycles 4'b1111 - Unlimited
27:25	RO	0x0	reserved

Bit	Attr	Reset Value	Description
24	RW	0x0	dfi_lp_en_dpd Enables DFI Low Power interface handshaking during Deep Power Down Entry/Exit. 1'b0 - Disabled 1'b1 - Enabled
23:20	RO	0x0	reserved
19:16	RW	0x7	dfi_tlp_resp Setting for tlp_resp time. Same value is used for both Power Down and Self refresh and Deep Power Down modes. DFI 2.1 specification, recommends using value of 7 always.
15:12	RW	0x0	dfi_lp_wakeup_sr Value to drive on dfi_lp_wakeup signal when Self Refresh mode is entered. Determines the DFI's tlp_wakeup time: 4'b0000 - 16 cycles 4'b0001 - 32 cycles 4'b0010 - 64 cycles 4'b0011 - 128 cycles 4'b0100 - 256 cycles 4'b0101 - 512 cycles 4'b0110 - 1024 cycles 4'b0111 - 2048 cycles 4'b1000 - 4096 cycles 4'b1001 - 8192 cycles 4'b1010 - 16384 cycles 4'b1011 - 32768 cycles 4'b1100 - 65536 cycles 4'b1101 - 131072 cycles 4'b1110 - 262144 cycles 4'b1111 - Unlimited
11:9	RO	0x0	reserved
8	RW	0x0	dfi_lp_en_sr Enables DFI Low Power interface handshaking during Self Refresh Entry/Exit. 1'b0 - Disabled 1'b1 - Enabled

Bit	Attr	Reset Value	Description
7:4	RW	0x0	dfi_lp_wakeup_pd Value to drive on dfi_lp_wakeup signal when Power Down mode is entered. Determines the DFI's tlp_wakeup time: 4'b0000 - 16 cycles 4'b0001 - 32 cycles 4'b0010 - 64 cycles 4'b0011 - 128 cycles 4'b0100 - 256 cycles 4'b0101 - 512 cycles 4'b0110 - 1024 cycles 4'b0111 - 2048 cycles 4'b1000 - 4096 cycles 4'b1001 - 8192 cycles 4'b1010 - 16384 cycles 4'b1011 - 32768 cycles 4'b1100 - 65536 cycles 4'b1101 - 131072 cycles 4'b1110 - 262144 cycles 4'b1111 - Unlimited
3:1	RO	0x0	reserved
0	RW	0x0	dfi_lp_en_pd Enables DFI Low Power interface handshaking during Power Down Entry/Exit. 1'b0 - Disabled 1'b1 - Enabled

DDR_PCTL_DFITRWRLVLRESP0
Address: Operational Base + offset (0x0300)
DFI Training dfi_wrlvl_resp Status 0 Register

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	dfi_wrlvl_resp0 Reports the status of the dif_wrlvl_resp[31:0] signal.

DDR_PCTL_DFITRWRLVLRESP1
Address: Operational Base + offset (0x0304)
DFI Training dfi_wrlvl_resp Status 1 Register

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	dfi_wrlvl_resp1 Reports the status of the dif_wrlvl_resp[63:32] signal.

DDR_PCTL_DFITRWRLVLRESP2
Address: Operational Base + offset (0x0308)
DFI Training dfi_wrlvl_resp Status 2 Register

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:0	RO	0x00	dfi_wrlvl_resp2 Reports the status of the dif_wrlvl_resp[71:64] signal.

DDR_PCTL_DFITRRDLVLRESP0
Address: Operational Base + offset (0x030c)
DFI Training dfi_rdlvl_resp Status 0 Register

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	dfi_rdlvl_resp0 Reports the status of the dif_rdlvl_resp[31:0] signal.

DDR_PCTL_DFITRRDLVLRESP1
Address: Operational Base + offset (0x0310)
DFI Training dfi_rdlvl_resp Status 1 Register

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	dfi_rdlvl_resp1 Reports the status of the dif_rdlvl_resp[63:32] signal.

DDR_PCTL_DFITRRDLVLRESP2
Address: Operational Base + offset (0x0314)
DFI Training dfi_rdlvl_resp Status 2 Register

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:0	RO	0x00	dfi_rdlvl_resp2 Reports the status of the dif_rdlvl_resp[71:64] signal.

DDR_PCTL_DFITRWRLVLDELAY0
Address: Operational Base + offset (0x0318)
DFI Training dfi_wrlvl_delay Configuration 0 Register

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	dfi_wrlvl_delay0 Sets the value to be driven on the signal dfi_wrlvl_delay_x[31:0].

DDR_PCTL_DFITRWRLVLDELAY1
Address: Operational Base + offset (0x031c)
DFI Training dfi_wrlvl_delay Configuration 1 Register

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	dfi_wrlvl_delay1 Sets the value to be driven on the signal dfi_wrlvl_delay_x[63:32].

DDR_PCTL_DFITRWRLVLDELAY2
Address: Operational Base + offset (0x0320)
DFI Training dfi_wrlvl_delay Configuration 2 Register

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:0	RW	0x00	dfi_wrlvl_delay2 Sets the value to be driven on the signal dfi_wrlvl_delay_x[71:64].

DDR_PCTL_DFITRRDLVLDELAY0
Address: Operational Base + offset (0x0324)
DFI Training dfi_rdlvl_delay Configuration 0 Register

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	dfi_rdlvl_delay0 Sets the value to be driven on the signal dfi_rdlvl_delay_x[31:0].

DDR_PCTL_DFITRRDLVLDELAY1
Address: Operational Base + offset (0x0328)
DFI Training dfi_rdlvl_delay Configuration 1 Register

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	dfi_rdlvl_delay1 Sets the value to be driven on the signal dfi_rdlvl_delay_x[63:32].

DDR_PCTL_DFITRRDLVLDELAY2
Address: Operational Base + offset (0x032c)
DFI Training dfi_rdlvl_delay Configuration 2 Register

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:0	RW	0x00	dfi_rdlvl_delay2 Sets the value to be driven on the signal dfi_rdlvl_delay_x[71:64].

DDR_PCTL_DFITRRDLVLGATEDELAY0
Address: Operational Base + offset (0x0330)
DFI Training dfi_rdlvl_gate_delay Configuration 0

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	dfi_rdlvl_gate_delay0 Sets the value to be driven on the signal dfi_rdlvl_gate_delay_x[31:0].

DDR_PCTL_DFITRRDLVLGATEDELAY1
Address: Operational Base + offset (0x0334)
DFI Training dfi_rdlvl_gate_delay Configuration 1

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	dfi_rdlvl_gate_delay1 Sets the value to be driven on the signal dfi_rdlvl_gate_delay_x[63:32].

DDR_PCTL_DFITRRDLVLGATEDELAY2
Address: Operational Base + offset (0x0338)
DFI Training dfi_rdlvl_gate_delay Configuration 2

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:0	RW	0x00	dfi_rdlvl_gate_delay2 Sets the value to be driven on the signal dfi_rdlvl_gate_delay_x[71:64].

DDR_PCTL_DFITRCMD
Address: Operational Base + offset (0x033c)
DFI Training Command Register

Bit	Attr	Reset Value	Description
31	R/WSC	0x0	dfitrcmd_start DFI Training Command Start. When this bit is set to 1, the command operation defined in the dfitrcmd_opcode field is started. This bit is automatically cleared by the uPCTL after the command is finished. The application can poll this bit to determine when uPCTL is ready to accept another command. This bit cannot be cleared to 1b0 by software.
30:13	RO	0x0	reserved
12:4	RW	0x000	dfitrcmd_en DFI Training Command Enable. Selects which bits of chosen DFI Training command to drive to 1'b1.
3:2	RO	0x0	reserved
1:0	RW	0x0	dfitrcmd_opcode DFI Training Command Opcode. Select which DFI Training command to generate for one n_clk cycle: 2'b00 - dfi_wrlvl_load 2'b01 - dfi_wrlvl_strobe 2'b10 - dfi_rdlvl_load 2'b11 - Reserved.

DDR_PCTL_IPVR
Address: Operational Base + offset (0x03f8)
IP Version Register

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	ip_version ASCII value for each number in the version, followed by a *.

DDR_PCTL_IPTR
Address: Operational Base + offset (0x03fc)
IP Type Register

Bit	Attr	Reset Value	Description
31:0	RO	0x44574300	ip_type Contains the IP's identification code, which is an ASCII value to identify the component and it is currently set to the string "DWC". This value never changes.

DDRPHY_REG00
Address: Operational Base + offset (0x0000)
DDR PHY register 00

Bit	Attr	Reset Value	Description
31:4	RO	0xf	reserved
3	RW	0x1	soft reset 1, active low
2	RW	0x1	soft reset 0, active low
1:0	RO	0x3	reserved

DDRPHY_REG01
Address: Operational Base + offset (0x0004)
DDR PHY register 01

Bit	Attr	Reset Value	Description
31:2	RO	0x1	reserved
1:0	RW	0x0	PHY working mode: 0x0: ddr3 PHY mode 0x1: ddr2 PHY mode 0x2: lpddr2 PHY mode 0x3: reserved

DDRPHY_REG02
Address: Operational Base + offset (0x0008)
DDR PHY register 02

Bit	Attr	Reset Value	Description
31:1	RO	0x0	reserved
0	RW	0x0	RXMEM calibration control, active high

DDRPHY_REG03
Address: Operational Base + offset (0x000c)
DDR PHY register 03

Bit	Attr	Reset Value	Description
31:7	RO	0x0	reserved
6:4	RW	0x2	right channel A read ODT delay
2:0	RW	0x2	left channel A read ODT delay

DDRPHY_REG04
Address: Operational Base + offset (0x0010)
DDR PHY register 04

Bit	Attr	Reset Value	Description
31:7	RO	0x0	reserved
6:4	RW	0x2	right channel B read ODT delay
2:0	RW	0x2	left channel B read ODT delay

DDRPHY_REG0B
Address: Operational Base + offset (0x002c)
DDR PHY register 0B

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:4	RW	0x6	CL value DDR3/LPDDR2 CAS Latency
3:0	RW	0x0	AL value DDR3/LPDDR2 additive latency

DDRPHY_REG0C
Address: Operational Base + offset (0x0030)
DDR PHY register 0C

Bit	Attr	Reset Value	Description
31:4	RO	0x0	reserved
3:0	RW	0x0	CWL value DDR3/LPDDR2 write latency

DDRPHY_REG11
Address: Operational Base + offset (0x0044)
DDR PHY register 11

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:4	RW	0xa	CMD P RCOMP, except for CK/CKB. The larger the value, the stronger the drive strength
3:0	RW	0xa	CMD N RCOMP, except for CK/CKB. The larger the value, the stronger the drive strength

DDRPHY_REG12
Address: Operational Base + offset (0x0048)
DDR PHY register 12

Bit	Attr	Reset Value	Description
31:2	RO	0x0	reserved
1	RW	0x1	CMD weak pull up enable, active LOW
0	RW	0x0	CMD weak pull down enable, active HIGH

DDRPHY_REG13
Address: Operational Base + offset (0x0048)
DDR PHY register 12

Bit	Attr	Reset Value	Description
31:5	RO	0x0	reserved
4	RW	0x0	CMD DLL clock phase select in bypass mode 1'b0: no delay 1'b1: 180 deg delay
3	RW	0x1	CMD DLL enable 1'b0: disable 1'b1: enable

Bit	Attr	Reset Value	Description
2:0	RW	0x4	CMD and ADDRESS DLL delay 3'd0: no delay 3'd1: 22.5 deg delay 3'd2: 45 deg delay 3'd3: 67.5 deg delay 3'd4: 90 deg delay 3'd5: 112.5 deg delay 3'd6: 135 deg delay 3'd7: 157.5 deg delay

DDRPHY_REG14
Address: Operational Base + offset (0x0050)
DDR PHY register 14

Bit	Attr	Reset Value	Description
31:4	RO	0x0	reserved
3	RW	0x0	CK DLL clock phase select in bypass mode 1'b0: no delay 1'b1: 180 deg delay
2:0	RW	0x0	CK DLL delay 3'd0: no delay 3'd1: 22.5 deg delay 3'd2: 45 deg delay 3'd3: 67.5 deg delay 3'd4: 90 deg delay 3'd5: 112.5 deg delay 3'd6: 135 deg delay 3'd7: 157.5 deg delay

DDRPHY_REG16
Address: Operational Base + offset (0x0058)
DDR PHY register 16

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:4	RW	0xa	CK/CKB PRCOMP. The larger the value, the stronger the drive strength
3:0	RW	0xa	CK/CKB NRCOMP. The larger the value, the stronger the drive strength

DDRPHY_REG20
Address: Operational Base + offset (0x0080)
DDR PHY register 20

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:4	RW	0xa	Left channel A PRCOMP. The larger the value, the stronger the drive strength in the scope from A_DQ0 to A_DQ7
3:0	RW	0xa	Left channel A NRCOMP. The larger the value, the stronger the drive strength in the scope from A_DQ0 to A_DQ7

DDRPHY_REG21
Address: Operational Base + offset (0x0084)
DDR PHY register 21

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:4	RW	0xa	Left channel A read pull-up ODT. The larger the value, the smaller the pull-up resistance in the scope from A_DQ0 to A_DQ7
3:0	RW	0xa	Left channel A read pull-down ODT. The larger the value, the smaller the pull-down resistance in the scope from A_DQ0 to A_DQ7

DDRPHY_REG26
Address: Operational Base + offset (0x0098)
DDR PHY register 26

Bit	Attr	Reset Value	Description
31:5	RO	0x0	reserved
4	RW	0x0	left channel A write DQ DLL phase select 1'b0: no delay 1'b1: 180 deg delay
3	RW	0x1	left channel A write DQ DLL enable 1'b0: disable 1'b1: enable

DDRPHY_REG27
Address: Operational Base + offset (0x009c)
DDR PHY register 27

Bit	Attr	Reset Value	Description
31:4	RO	0x0	reserved
3	RW	0x0	left channel A write DQS DLL phase select 1'b0: no delay 1'b1: 180 deg delay
2:0	RW	0x0	left channel A write DQS DLL delay 3'd0: no delay 3'd1: 22.5 deg delay 3'd2: 45 deg delay 3'd3: 67.5 deg delay 3'd4: 90 deg delay 3'd5: 112.5 deg delay 3'd6: 135 deg delay 3'd7: 157.5 deg delay

DDRPHY_REG28
Address: Operational Base + offset (0x00a0)
DDR PHY register 28

Bit	Attr	Reset Value	Description
31:2	RO	0x0	reserved
1:0	RW	0x0	left channel A read DQS DLL delay 2'd0: no delay 2'd1: 22.5 deg delay 2'd2: 45 deg delay 2'd3: 67.5 deg delay

DDRPHY_REG30
Address: Operational Base + offset (0x00c0)
DDR PHY register 30

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:4	RW	0xa	Right channel A PRCOMP. The larger the value, the stronger the drive strength in the scope from A_DQ8 to A_DQ15
3:0	RW	0xa	Right channel A NRCOMP. The larger the value, the stronger the drive strength in the scope from A_DQ8 to A_DQ15

DDRPHY_REG31
Address: Operational Base + offset (0x00c4)
DDR PHY register 31

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:4	RW	0xa	Right channel A read pull-up ODT. The larger the value, the smaller the pull-up resistance in the scope from A_DQ8 to A_DQ15
3:0	RW	0xa	Right channel A read pull-down ODT. The larger the value, the smaller the pull-down resistance in the scope from A_DQ8 to A_DQ15

DDRPHY_REG36
Address: Operational Base + offset (0x00d4)
DDR PHY register 36

Bit	Attr	Reset Value	Description
31:5	RO	0x0	reserved
4	RW	0x0	Right channel A write DQ DLL phase select 1'b0: no delay 1'b1: 180 deg delay
3	RW	0x1	Right channel A write DQ DLL enable 1'b0: disable 1'b1: enable
2:0	RW	0x4	Right channel A write DQ DLL delay 3'd0: no delay 3'd1: 22.5 deg delay 3'd2: 45 deg delay 3'd3: 67.5 deg delay 3'd4: 90 deg delay 3'd5: 112.5 deg delay 3'd6: 135 deg delay 3'd7: 157.5 deg delay

DDRPHY_REG37
Address: Operational Base + offset (0x00d8)
DDR PHY register 37

Bit	Attr	Reset Value	Description
31:4	RO	0x0	reserved
3	RW	0x0	Right channel A write DQS DLL phase select 1'b0: no delay 1'b1: 180 deg delay

Bit	Attr	Reset Value	Description
2:0	RW	0x0	Right channel A write DQS DLL delay 3'd0: no delay 3'd1: 22.5 deg delay 3'd2: 45 deg delay 3'd3: 67.5 deg delay 3'd4: 90 deg delay 3'd5: 112.5 deg delay 3'd6: 135 deg delay 3'd7: 157.5 deg delay

DDRPHY_REG38
Address: Operational Base + offset (0x00dc)
DDR PHY register 38

Bit	Attr	Reset Value	Description
31:2	RO	0x0	reserved
1:0	RW	0x0	Right channel A read DQS DLL delay 2'd0: no delay 2'd1: 22.5 deg delay 2'd2: 45 deg delay 2'd3: 67.5 deg delay

DDRPHY_REG40
Address: Operational Base + offset (0x0100)
DDR PHY register 40

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:4	RW	0xa	Left channel B PRCOMP. The larger the value, the stronger the drive strength in the scope from B_DQ0 to B_DQ7
3:0	RW	0xa	Left channel B NRCOMP. The larger the value, the stronger the drive strength in the scope from B_DQ0 to B_DQ7

DDRPHY_REG41
Address: Operational Base + offset (0x0104)
DDR PHY register 41

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:4	RW	0xa	Left channel B read pull-up ODT. The larger the value, the smaller the pull-up resistance in the scope from B_DQ0 to B_DQ7
3:0	RW	0xa	Left channel B read pull-down ODT. The larger the value, the smaller the pull-down resistance in the scope from B_DQ0 to B_DQ7

DDRPHY_REG46
Address: Operational Base + offset (0x0118)
DDR PHY register 46

Bit	Attr	Reset Value	Description
31:5	RO	0x0	reserved
4	RW	0x0	left channel B write DQ DLL phase select 1'b0: no delay 1'b1: 180 deg delay

Bit	Attr	Reset Value	Description
3	RW	0x1	left channel B write DQ DLL enable 1'b0: disable 1'b1: enable
2:0	RW	0x4	left channel B write DQ DLL delay 3'd0: no delay 3'd1: 22.5 deg delay 3'd2: 45 deg delay 3'd3: 67.5 deg delay 3'd4: 90 deg delay 3'd5: 112.5 deg delay 3'd6: 135 deg delay 3'd7: 157.5 deg delay

DDRPHY_REG47

Address: Operational Base + offset (0x011c)

DDR PHY register 47

Bit	Attr	Reset Value	Description
31:4	RO	0x0	reserved
3	RW	0x0	left channel B write DQS DLL phase select 1'b0: no delay 1'b1: 180 deg delay
2:0	RW	0x0	left channel B write DQS DLL delay 3'd0: no delay 3'd1: 22.5 deg delay 3'd2: 45 deg delay 3'd3: 67.5 deg delay 3'd4: 90 deg delay 3'd5: 112.5 deg delay 3'd6: 135 deg delay 3'd7: 157.5 deg delay

DDRPHY_REG48

Address: Operational Base + offset (0x0120)

DDR PHY register 48

Bit	Attr	Reset Value	Description
31:2	RO	0x0	reserved
1:0	RW	0x0	left channel B read DQS DLL delay 2'd0: no delay 2'd1: 22.5 deg delay 2'd2: 45 deg delay 2'd3: 67.5 deg delay

DDRPHY_REG50

Address: Operational Base + offset (0x0140)

DDR PHY register 50

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:4	RW	0xa	Right channel B PRCOMP. The larger the value, the stronger the drive strength in the scope from B_DQ8 to B_DQ15
3:0	RW	0xa	Right channel B NRCOMP. The larger the value, the stronger the drive strength in the scope from B_DQ8 to B_DQ15

DDRPHY_REG51
Address: Operational Base + offset (0x0144)
DDR PHY register 51

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:4	RW	0xa	Right channel B read pull-up ODT. The larger the value, the smaller the pull-up resistance in the scope from B_DQ8 to B_DQ15
3:0	RW	0xa	Right channel B read pull-down ODT. The larger the value, the smaller the pull-down resistance in the scope from B_DQ8 to B_DQ15

DDRPHY_REG56
Address: Operational Base + offset (0x0158)
DDR PHY register 56

Bit	Attr	Reset Value	Description
31:5	RO	0x0	reserved
4	RW	0x0	Right channel B write DQ DLL phase select 1'b0: no delay 1'b1: 180 deg delay
3	RW	0x1	Right channel B write DQ DLL enable 1'b0: disable 1'b1: enable
2:0	RW	0x4	Right channel B write DQ DLL delay 3'd0: no delay 3'd1: 22.5 deg delay 3'd2: 45 deg delay 3'd3: 67.5 deg delay 3'd4: 90 deg delay 3'd5: 112.5 deg delay 3'd6: 135 deg delay 3'd7: 157.5 deg delay

DDRPHY_REG57
Address: Operational Base + offset (0x015c)
DDR PHY register 57

Bit	Attr	Reset Value	Description
31:4	RO	0x0	reserved
3	RW	0x0	Right channel B write DQS DLL phase select 1'b0: no delay 1'b1: 180 deg delay
2:0	RW	0x0	Right channel B write DQS DLL delay 3'd0: no delay 3'd1: 22.5 deg delay 3'd2: 45 deg delay 3'd3: 67.5 deg delay 3'd4: 90 deg delay 3'd5: 112.5 deg delay 3'd6: 135 deg delay 3'd7: 157.5 deg delay

DDRPHY_REG58
Address: Operational Base + offset (0x0160)

DDR PHY register 58

Bit	Attr	Reset Value	Description
31:2	RO	0x0	reserved
1:0	RW	0x0	Right channel B read DQS DLL delay 2'd0: no delay 2'd1: 22.5 deg delay 2'd2: 45 deg delay 2'd3: 67.5 deg delay

DDRPHY_REGB0

Address: Operational Base + offset (0x02c0)

DDR PHY register B0

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:4	RW	0x7	A1 de-skew
3:0	RW	0x7	A0 de-skew

DDRPHY_REGB1

Address: Operational Base + offset (0x02c4)

DDR PHY register B1

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:4	RW	0x7	A3 de-skew
3:0	RW	0x7	A2 de-skew

DDRPHY_REGB2

Address: Operational Base + offset (0x02c8)

DDR PHY register B2

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:4	RW	0x7	A5 de-skew
3:0	RW	0x7	A4 de-skew

DDRPHY_REGB3

Address: Operational Base + offset (0x02cc)

DDR PHY register B3

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:4	RW	0x7	A7 de-skew
3:0	RW	0x7	A6 de-skew

DDRPHY_REGB4

Address: Operational Base + offset (0x02d0)

DDR PHY register B4

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:4	RW	0x7	A9 de-skew
3:0	RW	0x7	A8 de-skew

DDRPHY_REGB5

Address: Operational Base + offset (0x02d4)
DDR PHY register B5

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:4	RW	0x7	A11 de-skew
3:0	RW	0x7	A10 de-skew

DDRPHY_REGB6
Address: Operational Base + offset (0x02d8)
DDR PHY register B6

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:4	RW	0x7	A13 de-skew
3:0	RW	0x7	A12 de-skew

DDRPHY_REGB7
Address: Operational Base + offset (0x02dc)
DDR PHY register B7

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:4	RW	0x7	A15 de-skew
3:0	RW	0x7	A14 de-skew

DDRPHY_REGB8
Address: Operational Base + offset (0x02e0)
DDR PHY register B8

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:4	RW	0x7	B1 de-skew
3:0	RW	0x7	B0 de-skew

DDRPHY_REGB9
Address: Operational Base + offset (0x02e4)
DDR PHY register B9

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:4	RW	0x7	RASB de-skew
3:0	RW	0x7	B2 de-skew

DDRPHY_REGBA
Address: Operational Base + offset (0x02e8)
DDR PHY register BA

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:4	RW	0x7	WEB de-skew
3:0	RW	0x7	CASB de-skew

DDRPHY_REGBB
Address: Operational Base + offset (0x02ec)
DDR PHY register BB

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:4	RW	0x7	CKB de-skew
3:0	RW	0x7	CK de-skew

DDRPHY_REGBC
Address: Operational Base + offset (0x02f0)
DDR PHY register BC

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:4	RW	0x7	CKE de-skew
3:0	RW	0x7	ODT0 de-skew

DDRPHY_REGBD
Address: Operational Base + offset (0x02f4)
DDR PHY register BD

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:4	RW	0x7	CSB0 de-skew
3:0	RW	0x7	RESETN de-skew

DDRPHY_REGBE
Address: Operational Base + offset (0x02f8)
DDR PHY register BE

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:4	RW	0x7	CSB1 de-skew
3:0	RW	0x7	ODT1 de-skew

DDRPHY_REGC0
Address: Operational Base + offset (0x0300)
DDR PHY register C0

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:4	RW	0x7	A_DM0 RX de-skew
3:0	RW	0x7	A_DM0 TX de-skew

DDRPHY_REGC1
Address: Operational Base + offset (0x0304)
DDR PHY register C1

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:4	RW	0x7	A_DQ0 RX de-skew
3:0	RW	0x7	A_DQ0 TX de-skew

DDRPHY_REGC2
Address: Operational Base + offset (0x0308)
DDR PHY register C2

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved

Bit	Attr	Reset Value	Description
7:4	RW	0x7	A_DQ1 RX de-skew
3:0	RW	0x7	A_DQ1 TX de-skew

DDRPHY_REGC3
Address: Operational Base + offset (0x030c)
DDR PHY register C3

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:4	RW	0x7	A_DQ2 RX de-skew
3:0	RW	0x7	A_DQ2 TX de-skew

DDRPHY_REGC4
Address: Operational Base + offset (0x0310)
DDR PHY register C4

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:4	RW	0x7	A_DQ3 RX de-skew
3:0	RW	0x7	A_DQ3 TX de-skew

DDRPHY_REGC5
Address: Operational Base + offset (0x0314)
DDR PHY register C5

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:4	RW	0x7	A_DQ4 RX de-skew
3:0	RW	0x7	A_DQ4 TX de-skew

DDRPHY_REGC6
Address: Operational Base + offset (0x0318)
DDR PHY register C6

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:4	RW	0x7	A_DQ5 RX de-skew
3:0	RW	0x7	A_DQ5 TX de-skew

DDRPHY_REGC7
Address: Operational Base + offset (0x031c)
DDR PHY register C7

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:4	RW	0x7	A_DQ6 RX de-skew
3:0	RW	0x7	A_DQ6 TX de-skew

DDRPHY_REGC8
Address: Operational Base + offset (0x0320)
DDR PHY register C8

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:4	RW	0x7	A_DQ7 RX de-skew

Bit	Attr	Reset Value	Description
3:0	RW	0x7	A_DQ7 TX de-skew

DDRPHY_REGC9
Address: Operational Base + offset (0x0324)
DDR PHY register C9

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:4	RW	0x7	A_DQS0 RX de-skew
3:0	RW	0x7	A_DQS0 TX de-skew

DDRPHY_REGCA
Address: Operational Base + offset (0x0328)
DDR PHY register CA

Bit	Attr	Reset Value	Description
31:4	RO	0x0	reserved
3:0	RW	0x7	A_DQSB0 TX de-skew

DDRPHY_REGCB
Address: Operational Base + offset (0x032c)
DDR PHY register CB

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:4	RW	0x7	A_DM1 RX de-skew
3:0	RW	0x7	A_DM1 TX de-skew

DDRPHY_REGCC
Address: Operational Base + offset (0x0330)
DDR PHY register CC

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:4	RW	0x7	A_DQ8 RX de-skew
3:0	RW	0x7	A_DQ8 TX de-skew

DDRPHY_REGCD
Address: Operational Base + offset (0x0334)
DDR PHY register CD

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:4	RW	0x7	A_DQ9 RX de-skew
3:0	RW	0x7	A_DQ9 TX de-skew

DDRPHY_REGCE
Address: Operational Base + offset (0x0338)
DDR PHY register CE

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:4	RW	0x7	A_DQ10 RX de-skew
3:0	RW	0x7	A_DQ10 TX de-skew

DDRPHY_REGCF
Address: Operational Base + offset (0x033c)
DDR PHY register CF

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:4	RW	0x7	A_DQ11 RX de-skew
3:0	RW	0x7	A_DQ11 TX de-skew

DDRPHY_REGD0
Address: Operational Base + offset (0x0340)
DDR PHY register D0

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:4	RW	0x7	A_DQ12 RX de-skew
3:0	RW	0x7	A_DQ12 TX de-skew

DDRPHY_REGD1
Address: Operational Base + offset (0x0344)
DDR PHY register D1

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:4	RW	0x7	A_DQ13 RX de-skew
3:0	RW	0x7	A_DQ13 TX de-skew

DDRPHY_REGD2
Address: Operational Base + offset (0x0348)
DDR PHY register D2

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:4	RW	0x7	A_DQ14 RX de-skew
3:0	RW	0x7	A_DQ14 TX de-skew

DDRPHY_REGD3
Address: Operational Base + offset (0x034c)
DDR PHY register D3

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:4	RW	0x7	A_DQ15 RX de-skew
3:0	RW	0x7	A_DQ15 TX de-skew

DDRPHY_REGD4
Address: Operational Base + offset (0x0350)
DDR PHY register D4

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:4	RW	0x7	A_DQS1 RX de-skew
3:0	RW	0x7	A_DQS1 TX de-skew

DDRPHY_REGD5

Address: Operational Base + offset (0x0354)
DDR PHY register D5

Bit	Attr	Reset Value	Description
31:4	RO	0x0	reserved
3:0	RW	0x7	A_DQSB1 TX de-skew

DDRPHY_REGD6
Address: Operational Base + offset (0x0358)
DDR PHY register D6

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:4	RW	0x7	B_DM0 RX de-skew
3:0	RW	0x7	B_DM0 TX de-skew

DDRPHY_REGD7
Address: Operational Base + offset (0x035c)
DDR PHY register D7

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:4	RW	0x7	B_DQ0 RX de-skew
3:0	RW	0x7	B_DQ0 TX de-skew

DDRPHY_REGD8
Address: Operational Base + offset (0x0360)
DDR PHY register D8

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:4	RW	0x7	B_DQ1 RX de-skew
3:0	RW	0x7	B_DQ1 TX de-skew

DDRPHY_REGD9
Address: Operational Base + offset (0x0364)
DDR PHY register D9

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:4	RW	0x7	B_DQ2 RX de-skew
3:0	RW	0x7	B_DQ2 TX de-skew

DDRPHY_REGDA
Address: Operational Base + offset (0x0368)
DDR PHY register DA

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:4	RW	0x7	B_DQ3 RX de-skew
3:0	RW	0x7	B_DQ3 TX de-skew

DDRPHY_REGDB
Address: Operational Base + offset (0x036c)
DDR PHY register DB

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:4	RW	0x7	B_DQ4 RX de-skew
3:0	RW	0x7	B_DQ4 TX de-skew

DDRPHY_REGDC
Address: Operational Base + offset (0x0370)
DDR PHY register DC

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:4	RW	0x7	B_DQ5 RX de-skew
3:0	RW	0x7	B_DQ5 TX de-skew

DDRPHY_REGDD
Address: Operational Base + offset (0x0374)
DDR PHY register DD

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:4	RW	0x7	B_DQ6 RX de-skew
3:0	RW	0x7	B_DQ6 TX de-skew

DDRPHY_REGDE
Address: Operational Base + offset (0x0378)
DDR PHY register DE

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:4	RW	0x7	B_DQ7 RX de-skew
3:0	RW	0x7	B_DQ7 TX de-skew

DDRPHY_REGDF
Address: Operational Base + offset (0x037c)
DDR PHY register DF

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:4	RW	0x7	B_DQS0 RX de-skew
3:0	RW	0x7	B_DQS0 TX de-skew

DDRPHY_REGE0
Address: Operational Base + offset (0x0380)
DDR PHY register E0

Bit	Attr	Reset Value	Description
31:4	RO	0x0	reserved
3:0	RW	0x7	B_DQSB0 TX de-skew

DDRPHY_REGE1
Address: Operational Base + offset (0x0384)
DDR PHY register E1

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:4	RW	0x7	B_DM1 RX de-skew

Bit	Attr	Reset Value	Description
3:0	RW	0x7	B_DM1 TX de-skew

DDRPHY_REGE2
Address: Operational Base + offset (0x0388)
DDR PHY register E2

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:4	RW	0x7	B_DQ8 RX de-skew
3:0	RW	0x7	B_DQ8 TX de-skew

DDRPHY_REGE3
Address: Operational Base + offset (0x038c)
DDR PHY register E3

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:4	RW	0x7	B_DQ9 RX de-skew
3:0	RW	0x7	B_DQ9 TX de-skew

DDRPHY_REGE4
Address: Operational Base + offset (0x0390)
DDR PHY register E4

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:4	RW	0x7	B_DQ10 RX de-skew
3:0	RW	0x7	B_DQ10 TX de-skew

DDRPHY_REGE5
Address: Operational Base + offset (0x0394)
DDR PHY register E5

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:4	RW	0x7	B_DQ11 RX de-skew
3:0	RW	0x7	B_DQ11 TX de-skew

DDRPHY_REGE6
Address: Operational Base + offset (0x0398)
DDR PHY register E6

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:4	RW	0x7	B_DQ12 RX de-skew
3:0	RW	0x7	B_DQ12 TX de-skew

DDRPHY_REGE7
Address: Operational Base + offset (0x039c)
DDR PHY register E7

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:4	RW	0x7	B_DQ13 RX de-skew
3:0	RW	0x7	B_DQ13 TX de-skew

DDRPHY_REGE8
Address: Operational Base + offset (0x03a0)
DDR PHY register E8

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:4	RW	0x7	B_DQ14 RX de-skew
3:0	RW	0x7	B_DQ14 TX de-skew

DDRPHY_REGE9
Address: Operational Base + offset (0x03a4)
DDR PHY register E9

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:4	RW	0x7	B_DQ15 RX de-skew
3:0	RW	0x7	B_DQ15 TX de-skew

DDRPHY_REGEA
Address: Operational Base + offset (0x03a8)
DDR PHY register EA

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:4	RW	0x7	B_DQS1 RX de-skew
3:0	RW	0x7	B_DQS1 TX de-skew

DDRPHY_REGEB
Address: Operational Base + offset (0x03ac)
DDR PHY register EB

Bit	Attr	Reset Value	Description
31:4	RO	0x0	reserved
3:0	RW	0x7	B_DQSB1 TX de-skew

DDRPHY_REGFA
Address: Operational Base + offset (0x03e8)
DDR PHY register FA

Bit	Attr	Reset Value	Description
31:4	RO	0x0	reserved
3	RO	0x0	Channel B High 8bit dqs gate sample dqs value(idqs)
2	RO	0x0	Channel B Low 8bit dqs gate sample dqs value(idqs)
1	RO	0x0	Channel A High 8bit dqs gate sample dqs value(idqs)
0	RO	0x0	Channel A Low 8bit dqs gate sample dqs value(idqs)

DDRPHY_REGFB
Address: Operational Base + offset (0x03ec)
DDR PHY register FB

Bit	Attr	Reset Value	Description
31:7	RO	0x0	reserved
6:4	RO	0x0	Calibration get the dll configure channel A low 8bit
3	RO	0x0	Calibration get the ophsel configure channel A low 8bit
2:0	RO	0x0	Calibration get the cyclesel configure channle A low 8bit

DDRPHY_REGFC
Address: Operational Base + offset (0x03f0)
DDR PHY register FC

Bit	Attr	Reset Value	Description
31:7	RO	0x0	reserved
6:4	RO	0x0	Calibration get the dll configure channel A high 8bit
3	RO	0x0	Calibration get the ophsel configure channel A high 8bit
2:0	RO	0x0	Calibration get the cyclesel configure channle A high 8bit

DDRPHY_REGFD
Address: Operational Base + offset (0x03f4)
DDR PHY register FD

Bit	Attr	Reset Value	Description
31:7	RO	0x0	reserved
6:4	RO	0x0	Calibration get the dll configure channel B low 8bit
3	RO	0x0	Calibration get the ophsel configure channel B low 8bit
2:0	RO	0x0	Calibration get the cyclesel configure channle B low 8bit

DDRPHY_REGFE
Address: Operational Base + offset (0x03f8)
DDR PHY register FE

Bit	Attr	Reset Value	Description
31:7	RO	0x0	reserved
6:4	RO	0x0	Calibration get the dll configure channel B high 8bit
3	RO	0x0	Calibration get the ophsel configure channel B high 8bit
2:0	RO	0x0	Calibration get the cyclesel configure channle B high 8bit

DDRPHY_REGFF
Address: Operational Base + offset (0x03fc)
DDR PHY register FF

Bit	Attr	Reset Value	Description
31:4	RO	0x0	reserved
3	RO	0x0	Channel B High 8bit Calibration done
2	RO	0x0	Channel B Low 8bit Calibration done
1	RO	0x0	Channel A High 8bit Calibration done
0	RO	0x0	Channel A Low 8bit Calibration done

Note:
(1) left channel A signals: A_DQS0, A_DQSB0, A_DQ7~A_DQ0, A_DM0
right channel A signals: A_DQS1, A_DQSB1, A_DQ15~A_DQ8, A_DM1
left channel B signals: B_DQS0, B_DQSB0, B_DQ7~B_DQ0, B_DM0
right channel B signals: B_DQS1, B_DQSB1, B_DQ15~B_DQ8, B_DM1
The delayed phase is in 1X clock domain, whose frequency is equivalent to SDRAM clock.
(2) per-bit de-skew, for detailed information, refer to following section.

7.5 Interface Description

DDR IOs are listed as following Table.

Table 7-1 DDR IO description

Pin Name	Description
CK	Active-high clock signal to the memory device.

CK_B	Active-low clock signal to the memory device.
CKE	Active-high clock enable signal to the memory device for two chip select.
CS_Bi (i=0,1)	Active-low chip select signal to the memory device. There are two chip select.
RAS_B	Active-low row address strobe to the memory device.
CAS_B	Active-low column address strobe to the memory device.
WE_B	Active-low write enable strobe to the memory device.
BA[2:0]	Bank address signal to the memory device.
A[15:0]	Address signal to the memory device.
DQ[15:0]	Bidirectional data line to the memory device.
DQS[1:0]	Active-high bidirectional data strobes to the memory device.
DQS_B[1:0]	Active-low bidirectional data strobes to the memory device.
DM[1:0]	Active-low data mask signal to the memory device.
ODTi (i=0,1)	On-Die Termination output signal for two chip select.
RESET	DDR3 reset signal.

7.5.1 State transition of PCTL

To operate PCTL, the programmer must be familiar with the available operational states and how to transition to each state from the current state.

Every software programmable register is accessible only during certain operational states. For information about what registers are accessible in each state, refer to "Software Registers," which provides this information in each register description. The general rule is that the PCTL must be in the Init_mem or Config states to successfully write most of the registers.

The following tables provide the programming sequences for moving to the various states of the state machine.

Moving to the Init_mem State

Step	Application	PCTL
1	Read STAT register	Returns the current PCTL state.
2	If STAT.ctl_stat = Init_mem, go to END.	
3	If STAT.ctl_stat = Config, go to Step9.	
4	If STAT.ctl_stat = Access, go to Step8.	
5	If STAT.ctl_stat = Low_power, go to Step7.	
6	Goto Step1.	PCTL is in a Transitional state and not in any of the previous operational states.
7	Write WAKEUP to SCTL.state_cmd and poll STAT.ctl_stat= Access.	Issues SRX, moves to the Access state, updates STAT.ctl_stat =Access when complete.
8	Write CFG to SCTL.state_cmd and poll STAT.ctl_stat= Config.	PCTL stalls the NIF; completes any pending transaction; issues PREA if required; moves into the Config state; updates STAT.ctl_stat =Config when complete.
9	Write INIT to SCTL.state_cmd and poll STAT.ctl_stat=Init_mem	Moves into the Init_mem state and updates STAT.ctl_stat =Init_mem.
END		PCTL is in Init_mem state.

Moving to Config State

Step	Application	PCTL
1	Read STAT register.	Returns the current PCTL state.
2	If STAT.ctl_stat= Config, goto END.	
3	If STAT.ctl_stat= Low_power, go to Step6.	
4	If STAT.ctl_stat= Init_mem or Access, go to Step7.	
5	Go to Step1.	PCTL is in a transitional state and is not in any of the previous operational states.
6	Write WAKEUP to CTL.state_cmd and poll STAT.ctl_stat= Access.	Issues SRX, moves to the Access state, and updates STAT.ctl_stat= Access when complete.
7	Write CFG to SCTL.state_cmd and poll STAT.ctl_stat= Config.	PCTL stalls the NIF; completes any pending transaction; issues PREA if required; moves into the Config state; and updates STAT.ctl_stat = Config when complete.

END		PCTL is in Config state.

Moving to Access State

Step	Application	PCTL
1	Read STAT register	Returns the current PCTL state.
2	If STAT.ctl_stat= Access, go to END.	
3	If STAT.ctl_stat= Config, go to Step9	
4	If STAT.ctl_stat= Init_mem, go to Step8	
5	If STAT.ctl_stat= Low_power, go to Step7.	
6	Goto Step1.	PCTL is in a transitional state and is not in any of the previous operational states.
7	Write WAKEUP to SCTL.state_cmd and poll STAT.ctl_stat= Access. Goto END	Issues SRX, moves to the Access state, updates STAT.ctl_stat= Access when complete.
8	Write CFG to SCTL.state_cmd and poll STAT.ctl_stat= Config.	Moves into the Config state, updates STAT.ctl_stat= Config when complete.
9	Write GO to SCTL.state_cmd and poll STAT.ctl_stat= Access.	Moves into the Access state, updates STAT.ctl_stat= Access when complete.
END		PCTL is in Access state.

Moving to Low Power State

Step	Application	PCTL
1	Read STAT register.	Returns current PCTL state.
2	If STAT.ctl_stat =Low_power, go to END.	
3	If STAT.ctl_stat = Access, go to Step9	
4	If STAT.ctl_stat = Config, go to Step8	
5	If STAT.ctl_stat = Init_mem, go to Step7.	
6	Goto Step1.	PCTL is in transitional state and is not in any of the previous operational states.
7	Write CFG to SCTL.state_cmd and poll STAT.ctl_stat= Config.	Moves into the Config state, updates STAT.ctl_stat = Config when complete.
8	Write GO to SCTL.state_cmd and poll STAT.ctl_stat= Access.	Moves into the Access state, updates STAT.ctl_stat =Access when complete.
9	Write SLEEP to SCTL.state_cmd and poll STAT.ctl_stat= Low_power.	Issues PDX if necessary; completes any pending transactions; issues PREA command; finally, issues SRE and updates STAT.ctl_stat = Low_power.
END		PCTL is in Low Power state

7.5.2 Initialization

PHY Initialization

DDR PHY power-up reset sequence:

1. PHY Register Reset: reset the PHY register block through presetn;
2. Configure registers : AL ,CL etc.（register address 0x38）
3. System reset：reset the PHY through system_rstn;
4. Soft reset(optional): after system reset, execute the soft reset by changing PHY_REG0 values. Soft reset function is on when PHY_REG0[3:2] =2'b11.
5. Start PHY initialization: after pclk, dfi_clk1x and dfi_clk2x clock signals are stable, initialize the PHY.
6. Start PHY Calibration: after ddr sdram initialization done, you can set PHY calibration start.
7. Close PHY Calibration: after step 5us, you can set close PHY calibration.
8. Start Write and Read.

DDR3 Initialization Sequence

The initialization steps for DDR3 SDRAMs are as follows:

1. Optionally maintain RESET# low for a minimum of either 200 us (power-up initialization) or 100ns (power-on initialization). The DDR PHY drives RESET# low from the beginning of reset assertion and therefore this step may be skipped when DRAM initialization is triggered if enough time may already have expired to satisfy the RESET# low time.
2. After RESET# is de-asserted, wait a minimum of 500 us with CKE low.
3. Apply NOP and drive CKE high.
4. Wait a minimum of tXPR.

5. Issue a load Mode Register 2 (MR2) command.
6. Issue a load Mode Register 3 (MR3) command.
7. Issue a load Mode Register (MR1) command (to set parameters and enable DLL).
8. Issue a load Mode Register (MR0) command to set parameters and reset DLL.
9. Issue ZQ calibration command.
10. Wait 512 SDRAM clock cycles for the DLL to lock (tDLLK) and ZQ calibration (tZQinit) to finish. This wait time is relative to Step 8, i.e. relative to when the DLL reset command was issued onto the SDRAM command bus.

LPDDR2 Initialization Sequence

The initialization steps for LPDDR2 SDRAMs are as follows:
1. Wait a minimum of 100 ns (tINIT1) with CKE driven low.
2. Apply NOP and set CKE high.
3. Wait a minimum of 200 us (tINIT3).
4. Issue a RESET command.
5. Wait a minimum of 1 us + 10 us (tINIT4 + tINIT5).
6. Issue a ZQ calibration command.
7. Wait a minimum of 1 us (tZQINIT).
8. Issue a Write Mode Register to MR1.
9. Issue a Write Mode Register to MR2
10. Issue a Write Mode Register to MR3

7.5.3 Low Power Operation

Low_power state can be entered/exited via following ways:
- Software control of PCTL State machine (highest priority)
- Hardware Low Power Interface (middle priority)
- Auto Self Refresh feature (lowest priority)

Note the priority of requests from Access to Low_power is highlighted above. The STAT.lp_trig register field reports which of the 3 requests caused the entry to Low_power state.

Software control of PCTL State

The application can request via software to enter the memories into Self Refresh state by issuing the SLEEP command by programming SCTL.PCTL responds to the software request by moving into the Low_power operational state and issuing the SRE command to the memories. Note that the Low_power state can only be reached from the Access state.
In a similar fashion, the application requests to exit the memories from Self Refresh by issuing a WAKEUP command by programming SCTL.. PCTL responds to the WAKEUP command issuing SRX and restoring normal NIF address channel operation.

Hardware Low Power Interface

The hardware low power interface can also be used to enter/exit Self Refresh. The functionality is enabled by setting SCFG.hw_low_power_en=1. Once that bit is set, the input c_sysreq has the ability to trigger entry into the Low Power configuration state just like the software methodology (SCTL.state_cmd = SLEEP). A hardware Low Power entry trigger will be ignored/denied if the input c_active_in=1 or n_avalid=1. It may be accepted if c_active_in=0 and n_avalid=0, depending on the current state of the PCTL. When SCFG.hw_low_power_en=1, the outputs c_sysack and c_active provide feedback as required by the AXI low power interface specification (this interface's operation is defined by the AXI specification). c_sysack acknowledges the request to go into the Low_power state, and c_active indicates when the PCTL is actually in the Low_power state.
The c_active output could also be used by an external Low Power controller to decide when to request a transistion to low power. When MCFG1.hw_idle > 0, c_active = 1'b0 indicates that the NIF has been idle for at least MCFG1.hw_idle * 32 * n_clk cycles while in the Access state.
When in low power the c_active output can used by an external Low Power controller to trigger a low power exit. c_active will be driven high when either c_active_in or n_avalid are high. The path from c_active_in and n_valid to c_active is asynchronous so even if the clocks have been removed c_active will assert. The Low Power controller should re-enable the clocks when c_active is driven high while in the Low_power state.

Auto Power Down/Self Refresh

The Power Down and/or Self Refresh sequence is automatically started by PCTL when the NIF address channel is idle for a number of cycles, depending on the programmed value in MCFG.pd_idle and MCFG1.sr_idle.
Following table outlines the effect of these settings in conjunction with NIF being idle.

pd_idle	sr_idle	Memory modes	Memory Type
0	0	none	All
>0	0	Power Down	All
0	>0	Self Refresh	All
>0	>0	Power Down -> Self Refresh[3]	All

Note:
*1. Power Down is entered if NIF is idle for pd_idle. Following on from that, if NIF continues to be idle for a further sr_idle*32 cycles, Power Down is exited and Self Refresh is entered.*
*2. Following on from that, if NIF continues to be idle for a further sr_idle*32 cycles, Power Down is exited and Self Refresh is entered.*

Removing PCTL's n_clk

In DDR3 and LPDDR2, the relationship between SRE/SRX and stopping/starting the memory clock (CK) are formalized and are accounted for automatically by PCTL. With DDR3 and LPDDR2, CK should only be stopped after PCTL has reached the Low_power state. The current operational state can be verified by reading STAT.ctl_stat. The CK must be started and stable before the Software or Hardware Low Power Interface attempts to take the memory out of Self Refresh.
PCTL's n_clk can be safely removed when PCTL is in Low Power state. The sequences outlined in following two tables should be followed for safe operation:

Table 7-2 Steps to safely remove PCTL's n_clk

Step	Application	PCTL
1	Write SLEEP to SCTL.state_cmd and poll STAT.ctl_stat = LOW_POWER.	Tells PCTL to move memories into Self Refresh and waits until this completes.
2	Write TREFI=0. Also, write DFITCRLUPDI=0 and DFIREFMSKI=0, if they are not already 0.	Stops any MC-driven DFI updates occurring internally with PCTL
3	Wait a minimum interval which is equivalent to the PCTL's Refresh Interval (previous value of TREFI*TOGCNT100N*internal timers clock period;	Ensures any already scheduled PHY/PVT updates have completed successfully.
4	Stop toggling n_clk to PCTL.	n_clk logic inside PCTL is stopped.
end		

Step	Application	PCTL
1	Drive c_active_in low	Confirms that system external to PCTL can accept a Low- power request
2	Drive c_sysreq low	System Low-power request
3	Wait for PCTL to drive c_sysack low	PCTL Low-power request acknowledgement
4	Check value of c_active when Step 3 occurs. - if c_active=1, request denied. Cannot remove n_clk. Go to END. - if c_active=0, request accepted.	PCTL low-power request status response
5	Stop toggling n_clk to PCTL	n_clk logic inside PCTL is stopped
end		

7.5.4 TX DLLs

All high speed IO signals' phase can be adjusted by TX DLLs. Following table illustrates these DLLs.

Table 7-3 DDR PHY TX DLLs Delay Step

Offset	Bit	Control Signal Phase	Default	Description
0x4c	2~0	CMD	0x4	CMD DLL delay step
0x50	2~0	CK	0x0	CK DLL delay step
0x98	2~0	A_DM0, A_DQ7~A_DQ0	0x4	DM and DQ DLL Signal delay step
0xd8	2~0	A_DM1, A_DQ15~A_DQ8	0x4	
0x118	2~0	B_DM0, B_DQ7~B_DQ0	0x4	
0x158	2~0	B_DM1, B_DQ15~B_DQ8	0x4	

0x9c	2~0	A_DQS0, A_DQSB0	0x0	TX DQS DLL Signal delay step
0xdc	2~0	A_DQS1, A_DQSB1	0x0	
0x11c	2~0	B_DQS0, B_DQSB0	0x0	
0x15c	2~0	B_DQS1, B_DQSB1	0x0	

Step 0x0 values means no phase delay, and 0x4 increases delay phase to 90 deg, 0x7 values corresponds to maximum phase delay. All DLLs having 8 delay steps which can get 90 deg phase delay by setting 0x4.

7.5.5 RX DLLs

The RX DLLs are used for sample RX DQS signals with proper phase delay and pulse edges. The DQS squelch (RXMEM) signal opens a window for passing RX DQS pulses, both RX DQS and DQS squelch signal phase can be adjusted by separate DLLs.

Table 7-4 DDR PHY RX DQS Delay Step

Offset	Bit	Control Signal Phase	Default	Description
0xe0	5~3	Left DQS squelch	0x0	Left DQS squelch delay step
	1~0	DQS0, DQSB0	0x0	Left Rx DQS delay step
0x120	5~3	Right DQS squelch	0x0	Right DQS squelch delay step
	1~0	DQS1, DQSB1	0x0	Right Rx DQS delay step
0xa0	1~0	A_DQS0, A_DQSB0	0x1	read DQS delay phase
0xe0	1~0	A_DQS1, A_DQSB1	0x1	
0x120	1~0	B_DQS0, B_DQSB0	0x1	
0x160	1~0	B_DQS1, B_DQSB1	0x1	
0xb0	6~4	left channel A RXMEM	0x1	RXMEM delay, unit x1 clock cycle
0xf0	6~4	right channel A RXMEM	0x1	
0x130	6~4	left channel B RXMEM	0x1	
0x170	6~4	right channel B RXMEM	0x1	
0xb0	3	left channel A RXMEM	0x1	additive and accumulative RXMEM delay, unit 2x clock cycle
0xf0	3	right channel A RXMEM	0x1	
0x130	3	left channel B RXMEM	0x1	
0x170	3	right channel B RXMEM	0x1	
0xb0	2~0	left channel A RXMEM	0x4	additive and accumulative RXMEM delay, unit per DLL step
0xf0	2~0	right channel A RXMEM	0x4	
0x130	2~0	left channel B RXMEM	0x4	
0x170	2~0	right channel B RXMEM	0x4	

7.5.6 High Speed IO Drive Strength

The tuning rage of driver resistance is 28ohm to 138ohm. By default, 0x5 is 46ohm for DDR3 CMD driver. When the control bit is set to be larger, the drive strength becomes stronger.

Table 7-5 CK, CMD Signal Drive Strength Register

Offset	Bit	Default	Description
0x44	7~4	0xa	adjustable CMD pull-up resistance
	3~0	0xa	adjustable CMD pull-down resistance
0x58	7~4	0xa	adjustable CK pull-up resistance
	3~0	0xa	adjustable CK pull-down resistance

Table 7-6 DM, DQ Signal Drive Strength Register

Offset	Bit	Default	Description
0x80	7~4	0xa	pull-up driving resistance for A_DQ0~A_DQ7
	3~0	0xa	pull-down driving resistance for A_DQ0~A_DQ7
0xc0	7~4	0xa	pull-up driving resistance for A_DQ8~A_DQ15
	3~0	0xa	pull-down driving resistance for A_DQ8~A_DQ15
0x100	7~4	0xa	pull-up driving resistance for B_DQ0~B_DQ7
	3~0	0xa	pull-down driving resistance for B_DQ0~B_DQ7
0x140	7~4	0xa	pull-up driving resistance for B_DQ8~B_DQ15
	3~0	0xa	pull-down driving resistance for B_DQ8~B_DQ15

The value is larger, the drive strength is stronger.

Table 7-7 DM/DQ/DQS/CMD Driver output resistance with control bit

Control bit	4'b0000	4'b0001	4'b0010	4'b0011	4'b0100	4'b0101
Pull-up resistance	+∞	309ohm	155ohm	103ohm	77ohm	63ohm
Pull-down resistance	+∞	309ohm	155ohm	103ohm	77ohm	63ohm
Control bit	**4'b0110**	**4'b0111**	**4'b1000**	**4'b1001**	**4'b1010**	**4'b1011**
Pull-up resistance	52ohm	45ohm	77ohm	62ohm	52ohm	44ohm
Pull-down resistance	52ohm	45ohm	77ohm	62ohm	52ohm	44ohm
Control bit	**4'b1100**	**4'b1101**	**4'b1110**	**4'b1111**		
Pull-up resistance	39ohm	34ohm	31ohm	28ohm		
Pull-down resistance	39ohm	34ohm	31ohm	28ohm		

Table 7-8 DM/DQ/DQS RX ODT resistance with control bit

Control bit	4'b0000	4'b0001	4'b0010	4'b0011	4'b0100	4'b0101
Pull-up resistance	+∞	861ohm	431ohm	287ohm	216ohm	172ohm
Pull-down resistance	+∞	861ohm	431ohm	287ohm	216ohm	172ohm
Control bit	**4'b0110**	**4'b0111**	**4'b1000**	**4'b1001**	**4'b1010**	**4'b1011**
Pull-up resistance	145ohm	124ohm	215ohm	172ohm	144ohm	123ohm
Pull-down resistance	145ohm	124ohm	215ohm	172ohm	144ohm	123ohm
Control bit	**4'b1100**	**4'b1101**	**4'b1110**	**4'b1111**		
Pull-up resistance	108ohm	96ohm	86ohm	78ohm		
Pull-down resistance	108ohm	96ohm	86ohm	78ohm		

7.5.7 PHY Low Speed Mode (200MHz)

DDR PHY supports low speed to high speed DDR3 and LPDDR2 by using two operating mode: normal delay line mode up to 800Mbps or more, low power mode where we support any speed up to 533Mbps. If all TX DLLs are bypassed, the PHY will enter low power state. The DDR PHY entries low power mode when setting DLLs into Bypass mode. Table 10-9 illustrates related register settings.

Table 7-9 Low Power DLL Setting

Offset	Bit	Default	Low power Setting	Description
0x290	4	0x0	0x1	right channel B TX DQ DLL in bypass mode
	3	0x0	0x1	left channel B TX DQ DLL in bypass mode
	2	0x0	0x1	right channel A TX DQ DLL in bypass mode
	1	0x0	0x1	left channel A TX DQ DLL in bypass mode
	0	0x0	0x1	CMD/CK DLL in bypass mode
0x4c	4	0x0	0x1	CMD DLL phase select
0x50	3	0x0	0x0	CK DLL phase select
0x98	4	0x0	0x1	A_DQ0~A_DQ7 TX DLL phase select
0x9c	3	0x0	0x0	A_DQS0/A_DQSB0 TX DLL phase select
0xd8	4	0x0	0x1	A_DQ8~A_DQ15 TX DLL phase select
0xdc	3	0x0	0x0	A_DQS1/A_DQSB1 TX DLL phase select
0x118	4	0x1	0x1	B_DQ0~B_DQ7 TX DLL phase select
0x11c	3	0x0	0x0	B_DQS0/B_DQSB0 TX DLL phase select
0x158	4	0x1	0x1	B_DQ8~B_DQ15 TX DLL phase select
0x15c	3	0x0	0x0	B_DQS1/B_DQSB1 TX DLL phase select

Those bits lead to command DLL in bypass mode. Both DQS0 DLL and DQ7~DQ0 DLLs are bypassed, but only DQS0 DLL should be in inversion mode because DQS0 step delay is 0x00 and DQ7~DQ0 DLL's step delay is 0x4.

7.5.8 Per bit de-skew tuning

Per-bit de-skew is designed for compensating PCB trace mismatch, DDR PHY support skew individually adjustable for all PHY signals. There are eight steps for each bit de-skew adjusting, and the adjust resolution under different corners is shown below:

Table 7-10 per-bit de-skew tuning resolution

	ff	tt	ss
de-skew resolution	15ps	20ps	30ps

Pre-bit de-skew is realized with inverter chain delay, per-bit de-skew control signals select how much inverters are connected to data path, the minimum resolution is determined by the two inverters minimum delay.

TX path deskew and RX path deskew employ same delay line, and they have same deskew

tuning resolution. Minimum RX deskew tuning resolution can be about 28ps with SMIC40lp tt corner process, and we can re-design tuning resolution according to system and customer requirement.

7.5.9 DDR PHY Calibration

DDR PHY auto-adjustment function has been implemented in the PHY. The entire training processes only need to modify the register to start and complete enough.

The entire training process is as follows:

1. PHY’s register is reset, the setup is complete.
2. Controller to send the initial command and to complete initialization.
3. Set the PHY's register beginning calibration.

Offset	Bit	Default	Description
0x8	7~2	0x0	Others register
	1	0x0	set calibration bypass mode (1:bypass mode; 0:nomal)
	0	0x0	set calibration start (1: start; 0: stop)

4. After setting register in 5us complete calibration, see the calibration register can be read at this time is complete.
5. Normal read and writes operation can begin.

The software training process is as follows:

1. PHY’s register is reset, the setup is complete.
2. Controller to send the initial command and to complete initialization.
3. Set the PHY's register calibration bypass mode.
4. Send 4 consecutive burst8/4 read command.
5. Read register address 0x3c4 value idqs.
6. According to the state diagram shown below to change the cycle ophse dll value.
7. Repeat 4,5,6 until you have completed.

Chapter 8 Process-Voltage-Temperature Monitor (PVTM)

8.1 Overview

The Process-Voltage-Temperature Monitor (PVTM) is used to monitor the chip performance variance caused by chip process, voltage and temperature.
PVTM supports the following features:

- A clock oscillation ring is integrated and used to generate a clock like signal, the frequency of this clock is determined by the cell delay value of clock oscillation ring circuit.
- A frequency counter is used to measure the frequency of the clock oscillation ring.
- There are three PVTMs inside, one for CORE, GPU and FUNC respectively.

8.2 Block Diagram

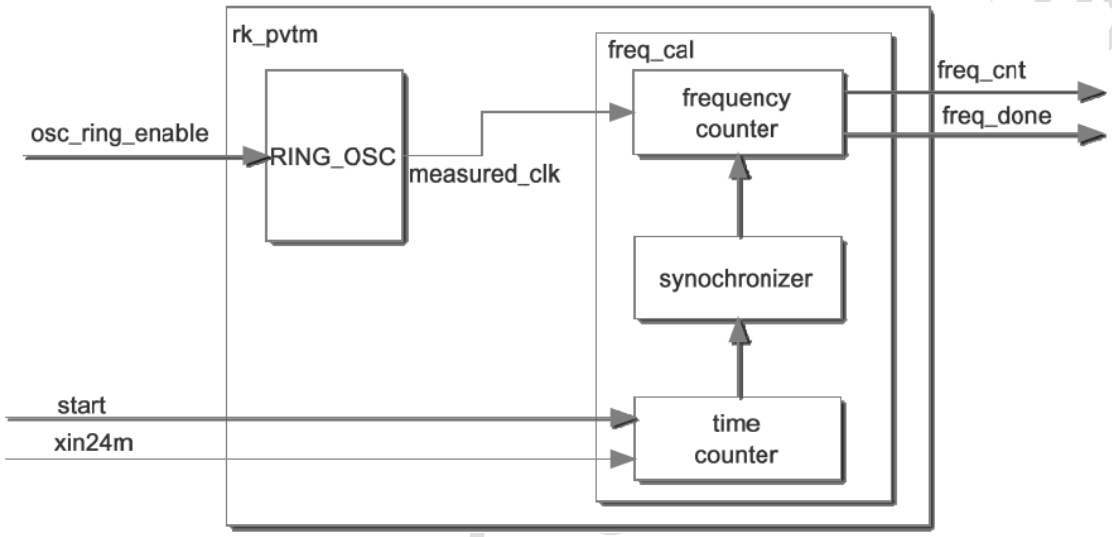


Fig. 8-1 PVTM Block Diagram

The PVTM include two main blocks:

- RING_OSC, composed with inverters with odd number, which is used to generate a clock.
- freq_cal, used to measure the frequency of clock which generated from the RING_SOC block.

8.3 Function Description

8.3.1 Frequency Calculation

A clock is generated by the RING_OSC, and a frequency fixed clock (24MHz) is used to calculate the cycles of the clock. Suppose the time period is 1s, then the clock period of RING_OSC clock is T=1/clock_counter(s), the cell delay value is T/2.

8.3.2 Controller and Status Information

The PVTM is controlled by GRF, and the monitor result is sent to GRF. Following table shows the PVTM controller and status information.

Table 8-1 CORE PVTM Controller and Status Information

Signal Name	Source/Destination	Default	Description
start	GRF_PVTM_CON0[0]	0	PVTM start control, high active
osc_en	GRF_ PVTM_CON0[1]	0	PVTM oscillator enable, high active
cal_cnt	GRF_ PVTM_CON1[31:0]	0	PVTM calculator count
freq_done	GRF_ PVTM_STATUS0[0]	0	PVTM frequency calculate done status, high active
freq_cnt	GRF_ PVTM_STATUS1[31:0]	0	PVTM frequency count

Table 8-2 GPU PVTM Controller and Status Information

Signal Name	Source/Destination	Default	Description
start	GRF_PVTM_CON0[8]	0	PVTM start control, high active
osc_en	GRF_PVTM_CON0[9]	0	PVTM oscillator enable, high active
cal_cnt	GRF_PVTM_CON2[31:0]	0	PVTM calculator count
freq_done	GRF_PVTM_STATUS0[1]	0	PVTM frequency calculate done status, high active
freq_cnt	GRF_PVTM_STATUS2[31:0]	0	PVTM frequency count

Table 8-3 FUNC PVTM Controller and Status Information

Signal Name	Source/Destination	Default	Description
start	GRF_PVTM_CON0[12]	0	PVTM start control, high active
osc_en	GRF_PVTM_CON0[13]	0	PVTM oscillator enable, high active
cal_cnt	GRF_PVTM_CON3[31:0]	0	PVTM calculator count
freq_done	GRF_PVTM_STATUS0[2]	0	PVTM frequency calculate done status, high active
freq_cnt	GRF_PVTM_STATUS3[31:0]	0	PVTM frequency count

8.4 Application Notes

8.4.1 PVTM Usage Flow

1. Enable the frequency fixed clock xin24m.
2. Reset the PVTM.
3. Set osc_ring_enable '1' to enable the generated clock.
4. Configure the cal_cnt to an appropriate value.
5. Set start '1' to calculate the cycles of the generated clock.
6. Wait the freq_done is asserted, then get the value of freq_cnt. The period of RING_OSC clock is T= cal_cnt*(Period of 24MHz clock)/freq_cnt, the cell delay value is T/2.

Chapter 9 Process-Voltage-Temperature Monitor (PVTM)

9.1 Overview

The Process-Voltage-Temperature Monitor (PVTM) is used to monitor the chip performance variance caused by chip process, voltage and temperature.
PVTM supports the following features:

- A clock oscillation ring is integrated and used to generate a clock like signal, the frequency of this clock is determined by the cell delay value of clock oscillation ring circuit.
- A frequency counter is used to measure the frequency of the clock oscillation ring.
- There are three PVTMs inside, one for CORE, GPU and FUNC respectively.

9.2 Block Diagram

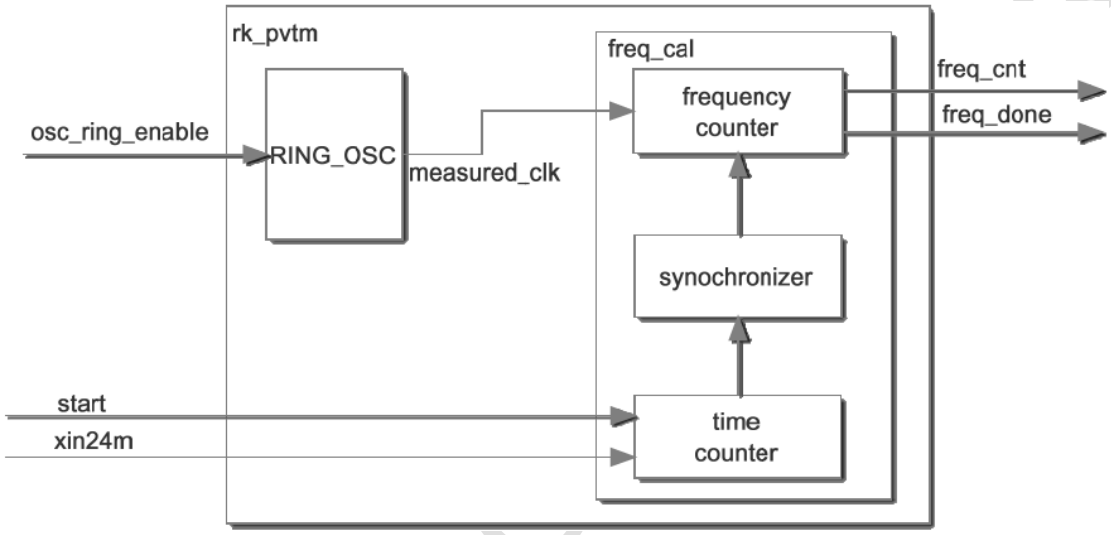


Fig. 9-1 PVTM Block Diagram

The PVTM include two main blocks:

- RING_OSC, composed with inverters with odd number, which is used to generate a clock.
- freq_cal, used to measure the frequency of clock which generated from the RING_SOC block.

9.3 Function Description

9.3.1 Frequency Calculation

A clock is generated by the RING_OSC, and a frequency fixed clock (24MHz) is used to calculate the cycles of the clock. Suppose the time period is 1s, then the clock period of RING_OSC clock is T=1/clock_counter(s), the cell delay value is T/2.

9.3.2 Controller and Status Information

The PVTM is controlled by GRF, and the monitor result is sent to GRF. Following table shows the PVTM controller and status information.

Table 9-1 CORE PVTM Controller and Status Information

Signal Name	Source/Destination	Default	Description
start	GRF_PVTM_CON0[0]	0	PVTM start control, high active
osc_en	GRF_ PVTM_CON0[1]	0	PVTM oscillator enable, high active
cal_cnt	GRF_ PVTM_CON1[31:0]	0	PVTM calculator count
freq_done	GRF_ PVTM_STATUS0[0]	0	PVTM frequency calculate done status, high active
freq_cnt	GRF_ PVTM_STATUS1[31:0]	0	PVTM frequency count

Table 9-2 GPU PVTM Controller and Status Information

Signal Name	Source/Destination	Default	Description
start	GRF_PVTM_CON0[8]	0	PVTM start control, high active
osc_en	GRF_PVTM_CON0[9]	0	PVTM oscillator enable, high active
cal_cnt	GRF_PVTM_CON2[31:0]	0	PVTM calculator count
freq_done	GRF_PVTM_STATUS0[1]	0	PVTM frequency calculate done status, high active
freq_cnt	GRF_PVTM_STATUS2[31:0]	0	PVTM frequency count

Table 9-3 FUNC PVTM Controller and Status Information

Signal Name	Source/Destination	Default	Description
start	GRF_PVTM_CON0[12]	0	PVTM start control, high active
osc_en	GRF_PVTM_CON0[13]	0	PVTM oscillator enable, high active
cal_cnt	GRF_PVTM_CON3[31:0]	0	PVTM calculator count
freq_done	GRF_PVTM_STATUS0[2]	0	PVTM frequency calculate done status, high active
freq_cnt	GRF_PVTM_STATUS3[31:0]	0	PVTM frequency count

9.4 Application Notes

9.4.1 PVTM Usage Flow

1. Enable the frequency fixed clock xin24m.
2. Reset the PVTM.
3. Set osc_ring_enable '1' to enable the generated clock.
4. Configure the cal_cnt to an appropriate value.
5. Set start '1' to calculate the cycles of the generated clock.
6. Wait the freq_done is asserted, then get the value of freq_cnt. The period of RING_OSC clock is T= cal_cnt*(Period of 24MHz clock)/freq_cnt, the cell delay value is T/2.

Chapter 10 Mobile Storage Host Controller

10.1 Overview

The Mobile Storage Host Controller is designed to support Secure Digital memory (SD- max version 3.01) with 1 bits or 4 bits data width, Multimedia Card(MMC-max version 4.51) with 1 bits or 4 bits or 8 bits data width.
The Host Controller is applied as SDMMC, SDIO and EMMC. The difference between these applications is shown in "Interface Description" and "Card Interface Features".
The Host Controller supports following features:

- Bus Interface Features:
  - Support AMBA AHB interface for master and slave
  - Support optional external DMA controllers for data transfers
  - Support combined single FIFO for both transmit and receive operations
  - Support FIFO size of 256x32
  - Support FIFO over-run and under-run prevention by stopping card clock
- Card Interface Features:
  - SDMMC:
    - Support Secure Digital memory protocol commands
    - Support Secure Digital I/O protocol commands
    - Support Multimedia Card protocol commands
    - Support CRC generation and error detection
    - Support programmable baud rate
    - Support power management and power switch
    - Support card detection
    - Support write protection
    - Support block size of 1 to 65,535 bytes
    - Support 1-bit and 4-bit SDR modes
    - Support 4-bit DDR mode
  - SDIO:
    - Support Secure Digital memory protocol commands
    - Support Secure Digital I/O protocol commands
    - Support Multimedia Card protocol commands
    - Support Command Completion Signal and interrupts to host
    - Support CRC generation and error detection
    - Support programmable baud rate
    - Support power management and power switch
    - Support block size of 1 to 65,535 bytes
    - Support 1-bit and 4-bit SDR modes
    - Support 4-bit DDR mode
  - EMMC:
    - Support Multimedia Card protocol commands
    - Support Command Completion Signal and interrupts to host
    - Support CRC generation and error detection
    - Support programmable baud rate
    - Support power management and power switch
    - Support hardware reset
    - Support block size of 1 to 65,535 bytes
    - Support 1-bit, 4-bit and 8-bit SDR modes
    - Support 4-bit and 8-bit DDR modes
- Clock Interface Features:
  - Internal frequency divider by 2 for cclk_in/cclk_in_sample/cclk_in_drv relative to CRU output
  - Support 0/90/180/270-degree phase shift operation for sample clock (cclk_in_sample) and drive clock(cclk_in_drv) relative to function clock(cclk_in) respectively
  - Support phase tuning using delay line for sample clock(cclk_in_sample) and drive clock(cclk_in_drv) relative to function clock (cclk_in) respectively.

- The max number of delay element number is 256, and the available range is affected by implementation and PVT
- The average value for every delay element is in the range of 40ps ~ 60ps

10.2 Block Diagram

The Host Controller consists of the following main functional blocks.

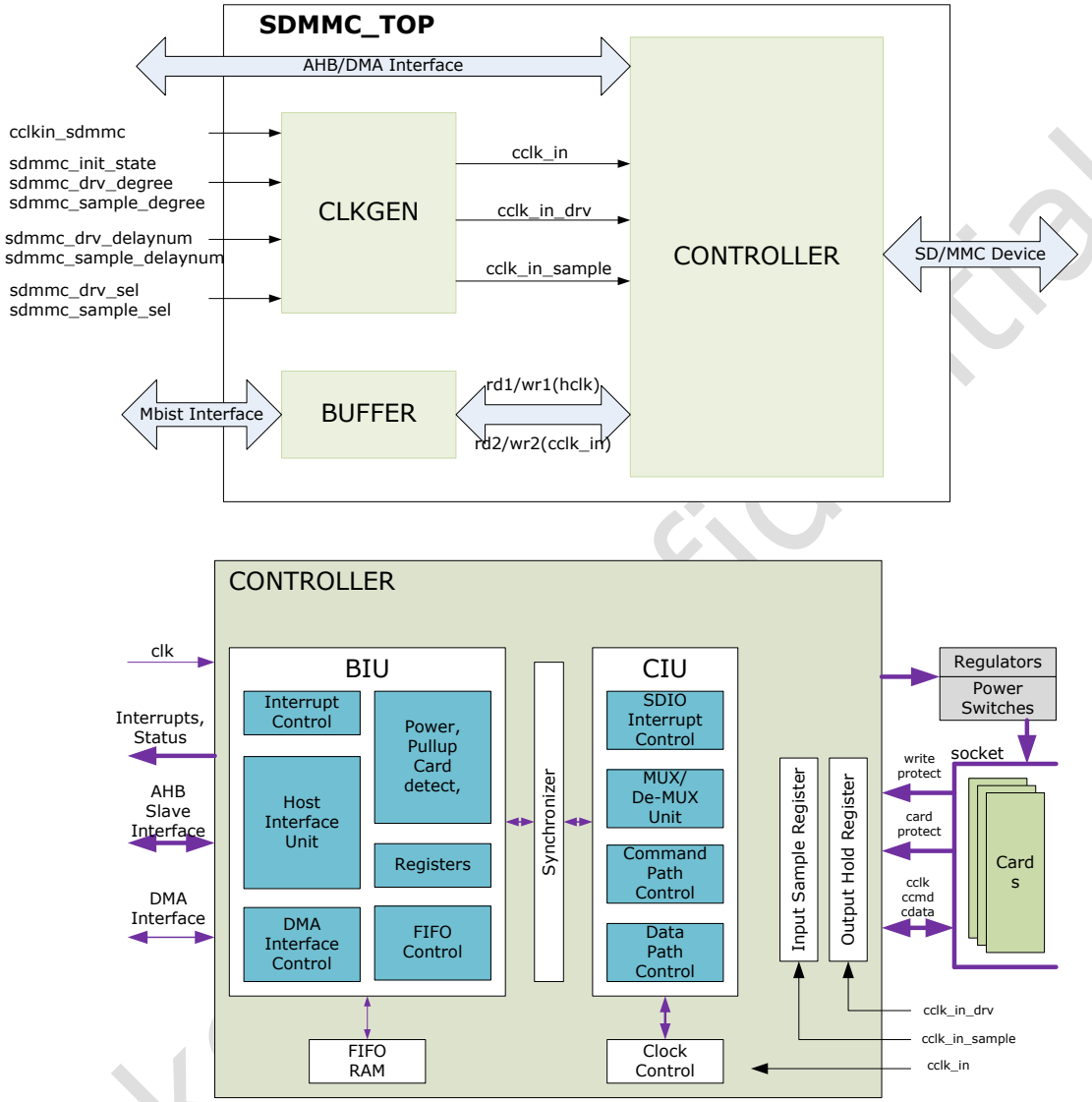


Fig. 10-1 Host Controller Block Diagram

- Clock Generate Unit(CLKGEN): generates card interface clock cclk_in/ cclk_sample/cclk_drv based on cclkin and configuration information.
- Asynchronous dual-port memory(BUFFER): Uses a two-clock synchronous read and synchronous write dual-port RAM. One of the ports is connected to the host clock, and the second port is connected to the card clock.
- Bus Interface Unit (BIU): Provides AMBA AHB interfaces for register and data read/writes.
- Card Interface Unit (CIU): Takes care of the SD/MMC protocols and provides clock management.

10.3 Function Description

10.3.1 Bus Interface Unit

The Bus Interface Unit provides the following functions:

- Host interface
- Interrupt control
- Register access

- External FIFO access
- Power control and card detection

1. Host Interface Unit

The Host Interface Unit is an AHB slave interface, which provides the interface between the SD/MMC card and the host bus.

2. Register Unit

The register unit is part of the bus interface unit; it provides read and write access to the registers.

All registers reside in the Bus Interface Unit clock domain. When a command is sent to a card by setting the start_bit, which is bit[31] of the SDMMC_CMD register, all relevant registers needed for the CIU operation are transferred to the CIU block. During this time, the registers that are transferred from the BIU to the CIU should not be written. The software should wait for the hardware to clear the start bit before writing to these registers again. The register unit has a hardware locking feature to prevent illegal writes to registers. The lock is necessary in order to avoid metastability violations, both because the host and card clock domains are different and to prevent illegal software operations.

Once a command start is issued by setting the start_bit of the SDMMC_CMD register, the following registers cannot be reprogrammed until the command is accepted by the card interface unit:

- SDMMC_CMD – Command
- SDMMC_CMDARG – Command Argument
- SDMMC_BYTCNT – Byte Count
- SDMMC_BLKSIZ – Block Size
- SDMMC_CLKDIV – Clock Divider
- SDMMC_CLKENA – Clock Enable
- SDMMC_CLKSRC – Clock Source
- SDMMC_TMOUT – Timeout
- SDMMC_CTYPE – Card Type

The hardware resets the start_bit once the CIU accepts the command. If a host write to any of these registers is attempted during this locked time, then the write is ignored and the hardware lock error bit is set in the raw interrupt status register. Additionally, if the interrupt is enabled and not masked for a hardware lock error, then an interrupt is sent to the host.

When the Card Interface Unit is in an idle state, it typically takes the following number of clocks for the command handshake, where clk is the BIU clock and cclk_in is the CIU clock:
3 (clk) + 3 (cclk_in)

Once a command is accepted, you can send another command to the CIU-which has a one-deep command queue-under the following conditions:

- If the previous command was not a data transfer command, the new command is sent to the SD/MMC card once the previous command completes.
- If the previous command is a data transfer command and if wait_prvdata_complete (bit[13]) of the Command register is set for the new command, the new command is sent to the SD/MMC card only when the data transfer completes.
- If the wait_prvdata_complete is 0, then the new command is sent to the SD/MMC card as soon as the previous command is sent. Typically, you should use this only to stop or abort a previous data transfer or query the card status in the middle of a data transfer.

3. Interrupt Controller Unit

The interrupt controller unit generates an interrupt that depends on the controller raw interrupt status, the interrupt-mask register, and the global interrupt-enable register bit. Once an interrupt condition is detected, it sets the corresponding interrupt bit in the raw interrupt status register. The raw interrupt status bit stays on until the software clears the bit by writing a 1 to the interrupt bit; a 0 leaves the bit untouched.

The interrupt port, int, is an active-high, level-sensitive interrupt. The interrupt port is active only when any bit in the raw interrupt status register is active, the corresponding interrupt mask bit is 1, and the global interrupt enable bit is 1. The interrupt port is registered in order to avoid any combinational glitches.

The int_enable is reset to 0 on power-on, and the interrupt mask bits are set to 32’h0,

which masks all the interrupts.

Notes:

Before enabling the interrupt, it is always recommended that you write 32'hffff_ffff to the raw interrupt status register in order to clear any pending unserviced interrupts. When clearing interrupts during normal operation, ensure that you clear only the interrupt bits that you serviced.

The SDIO Interrupts, Receive FIFO Data Request (RXDR), and Transmit FIFO Data Request (TXDR) are set by level-sensitive interrupt sources. Therefore, the interrupt source should be first cleared before you can clear the interrupt bit of the Raw Interrupt register. For example, on seeing the Receive FIFO Data Request (RXDR) interrupt, the FIFO should be emptied so that the "FIFO count greater than the RX-Watermark" condition, which triggers the interrupt, becomes inactive. The rest of the interrupts are triggered by a single clock-pulse-width source.

Table 10-1 Bits in Interrupt Status Register

Bits	Interrupt	Description
24	sdio_interrupt	Interrupt from SDIO card. In MMC-Ver3.3-only mode, these bits are always 0
16	Card no-busy	If card exit busy status, the interrupt happened
15	End Bit Error (read) /Write no CRC (EBE)	Error in end-bit during read operation, or no data CRC received during write operation. For MMC CMD19, there may be no CRC status returned by the card. Hence, EBE is set for CMD19. The application should not treat this as an error.
14	Auto Command Done (ACD)	Stop/abort commands automatically sent by card unit and not initiated by host; similar to Command Done (CD) interrupt. Recommendation: Software typically need not enable this for non CE-ATA accesses; Data Transfer Over (DTO) interrupt that comes after this interrupt determines whether data transfer has correctly competed.
13	Start Bit Error (SBE)	Error in data start bit when data is read from a card. In 4-bit mode, if DAT[0] line indicates start bit-that is, 0-and any of the other data bits do not have start bit, then this error is set. Busy Complete Interrupt when data is written to the card. This interrupt is generated after completion of busy driven by the card after the last data block is written into the card.
12	Hardware Locked write Error (HLE)	During hardware-lock period, write attempted to one of locked registers. When software sets the start_cmd bit in the SDMMC_CMD register, the Host Controller tries to load the command. If the command buffer is already filled with a command, this error is raised. The software then has to reload the command.
11	FIFO Underrun/ Overrun Error (FRUN)	Host tried to push data when FIFO was full, or host tried to read data when FIFO was empty. Typically this should not happen, except due to error in software. Card unit never pushes data into FIFO when FIFO is full, and pop data when FIFO is empty. If IDMAC is enabled, FIFO underrun/overrun can occur due to a programming error on MSIZE and watermark values in SDMMC_FIFOTH register.
10	Data Starvation by Host Timeout (HTO)	To avoid data loss, card clock out is stopped if FIFO is empty when writing to card, or FIFO is full when reading from card. Whenever card clock is stopped to avoid data loss, data-starvation timeout counter is started with data-timeout value. This interrupt is set if host does not fill data into FIFO during write to

Bits	Interrupt	Description
		card, or does not read from FIFO during read from card before timeout period. Even after timeout, card clock stays in stopped state, with CIU state machines waiting. It is responsibility of host to push or pop data into FIFO upon interrupt, which automatically restarts cclk_out and card state machines. Even if host wants to send stop/abort command, it still needs to ensure it has to push or pop FIFO so that clock starts in order for stop/abort command to send on cmd signal along with data that is sent or received on data line.
9	Data Read Timeout (DRTO)	In Normal functioning mode: Data read timeout (DRTO) Data timeout occurred. Data Transfer Over (DTO) also set if data timeout occurs. In Boot Mode: Boot Data Start (BDS) When set, indicates that Host Controller has started to receive boot data from the card. A write to this register with a value of 1 clears this interrupt.
8	Response Timeout (RTO)	In Normal functioning mode: Response timeout (RTO) Response timeout occurred. Command Done (CD) also set if response timeout occurs. If command involves data transfer and when response times out, no data transfer is attempted by Host Controller. In Boot Mode: Boot Ack Received (BAR) When expect_boot_ack is set, on reception of a boot acknowledge pattern—0-1-0—this interrupt is asserted. A write to this register with a value of 1 clears this interrupt.
7	Data CRC Error (DCRC)	Received Data CRC does not match with locally-generated CRC in CIU. Can also occur if the Write CRC status is incorrectly sampled by the Host.
6	Response CRC Error (RCRC)	Response CRC does not match with locally-generated CRC in CIU.
5	Receive FIFO Data Request (RXDR)	Interrupt set during read operation from card when FIFO level is greater than Receive-Threshold level. Recommendation: In DMA modes, this interrupt should not be enabled. In non-DMA mode: pop RX_WMark + 1 data from FIFO.

4	Transmit FIFO Data Request (TXDR)	Interrupt set during write operation to card when FIFO level reaches less than or equal to Transmit-Threshold level.
		Recommendation:
		In DMA modes, this interrupt should not be enabled.
		In non-DMA mode:
		if (pending_bytes > (FIFO_DEPTH - TX_WMark))
		push (FIFO_DEPTH - TX_WMark) data into FIFO
		else
		push pending_bytes data into FIFO

3	Data Transfer Over (DTO)	Indicates Data transfer completed. Though on detection of errors-Start Bit Error, Data CRC error, and so on, DTO may or may not be set; the application must issue CMD12, which ensures that

Bits	Interrupt	Description
		DTO is set. Recommendation: In non-DMA mode, when data is read from card, on seeing interrupt, host should read any pending data from FIFO. In DMA mode, DMA controllers guarantee FIFO is flushed before interrupt. DTO bit is set at the end of the last data block, even if the device asserts MMC busy after the last data block.
2	Command Done(CD)	Command sent to card and got response from card, even if Response Error or CRC error occurs.
1	Response Error (RE)	Error in received response set if one of following occurs: ● Transmission bit != 0 ● Command index mismatch ● End-bit != 1
0	Card-Detect (CDT)	When one or more cards inserted or removed, this interrupt occurs. Software should read card-detect register to determine current card status. Recommendation: After power-on and before enabling interrupts, software should read card detect register and store it in memory. When interrupt occurs, it should read card detect register and compare it with value stored in memory to determine which card(s) were removed/inserted. Before exiting ISR, software should update memory with new card-detect value.

4. FIFO Controller Unit

The FIFO controller interfaces the external FIFO to the host interface and the card controller unit. When FIFO overrun and under-run conditions occur, the card clock stops in order to avoid data loss.

The FIFO uses a two-clock synchronous read and synchronous write dual-port RAM. One of the ports is connected to the host clock, clk, and the second port is connected to the card clock, cclk_in.

Notes: The FIFO controller does not support simultaneous read/write access from the same port. For debugging purposes, the software may try to write into the FIFO and read back the data; results are indeterminate, since the design does not support read/write access from the same port.

5. Power Control and Card Detection Unit

The register unit has registers that control the power. Power to each card can be selectively turned on or off.

The card detection unit looks for any changes in the card-detect signals for card insertion or card removal. It filters out the debounces associated with mechanical insertion or removal, and generates one interrupt to the host. You can program the debounce filter value.

On power-on, the controller should read in the card_detect port and store the value in the memory. Upon receiving a card-detect interrupt, it should again read the card_detect port and XOR with the previous card-detect status to find out which card has interrupted. If more than one card is simultaneously removed or inserted, there is only one card-detect interrupt; the XOR value indicates which cards have been disturbed. The memory should be updated with the new card-detect value.

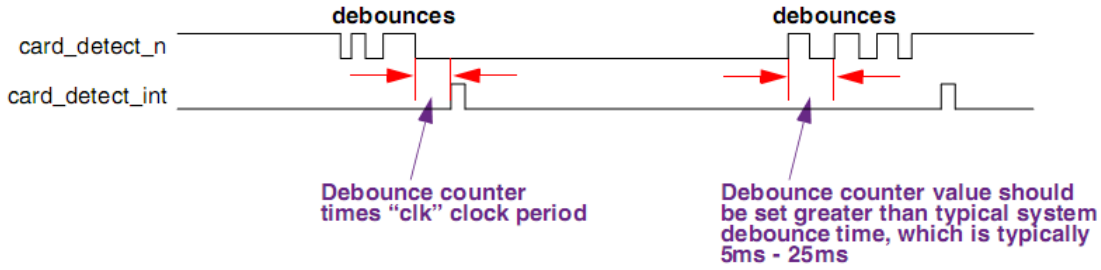


Fig. 10-2 SD/MMC Card-Detect Signal

6. DMA Interface Unit

DMA signals interface the Host Controller to an external DMA controller to reduce the software overhead during FIFO data transfers. The DMA request/acknowledge handshake is used for only data transfers. The DMA interface provides a connection to the DMA Controller.

On seeing the DMA request, the DMA controller initiates accesses through the host interface to read or write into the data FIFO. The Host Controller has FIFO transmit/receive watermark registers that you can set, depending on system latency. The DMA interface asserts the request in the following cases:

- Read from a card when the data FIFO word count exceeds the Rx-Watermark level
- Write to a card when the FIFO word count is less than or equal to the Tx-Watermark level

When the DMA interface is enabled, you can use normal host read/write to access the data FIFO.

10.3.2 Card Interface Unit

The Card Interface Unit (CIU) interfaces with the Bus Interface Unit (BIU) and the devices. The host writes command parameters to the BIU control registers, and these parameters are then passed to the CIU. Depending on control register values, the CIU generates SD/MMC command and data traffic on a selected card bus according to SD/MMC protocol. The Host Controller accordingly controls the command and data path.

The following software restrictions should be met for proper CIU operation:

- Only one data transfer command can be issued at a time.
- During an open-ended card write operation, if the card clock is stopped because the FIFO is empty, the software must first fill the data into the FIFO and start the card clock. It can then issue only a stop/abort command to the card.
- When issuing card reset commands (CMD0, CMD15 or CMD52_reset) while a card data transfer is in progress, the software must set the stop_abort_cmd bit in the Command register so that the Host Controller can stop the data transfer after issuing the card reset command.
- When the data end bit error is set in the SDMMC_RINTSTS register, the Host Controller does not guarantee SDIO interrupts. The software should ignore the SDIO interrupts and issue the stop/abort command to the card, so that the card stops sending the read data.
- If the card clock is stopped because the FIFO is full during a card read, the software should read at least two FIFO locations to start the card clock.

The CIU block consists of the following primary functional blocks:

- Command path
- Data path
- SDIO interrupt control
- Clock control
- Mux/demux unit

1. Command Path

The command path performs the following functions:

- Loads clock parameters
- Loads card command parameters
- Sends commands to card bus (ccmd_out line)
- Receives responses from card bus (ccmd_in line)
- Sends responses to BIU
- Drives the P-bit on command line

A new command is issued to the Host Controller by programming the BIU registers and setting the start_cmd bit in the Command register. The BIU asserts start_cmd, which indicates that a new command is issued to the SD/MMC device. The command path loads this new command (command, command argument, timeout) and sends acknowledge to the BIU by asserting cmd_taken.
Once the new command is loaded, the command path state machine sends a command to the device bus-including the internally generated CRC7-and receives a response, if any. The state machine then sends the received response and signals to the BIU that the command is done, and then waits for eight clocks before loading a new command.

Load Command Parameters

One of the following commands or responses is loaded in the command path:

- New command from BIU – When start_cmd is asserted, then the start_cmd bit is set in the Command register.
- Internally-generated auto-stop command – When the data path ends, the stop command request is loaded.
- IRQ response with RCA 0x000 – When the command path is waiting for an IRQ response from the MMC card and a "send irq response" request is signaled by the BIU, then the send_irq_response bit is set in the control register.

Loading a new command from the BIU in the command path depends on the following Command register bit settings:

- update_clock_registers_only – If this bit is set in the Command register, the command path updates only the clock enable, clock divider, and clock source registers. If this bit is not set, the command path loads the command, command argument, and timeout registers; it then starts processing the new command.
- wait_prvdata_complete – If this bit is set, the command path loads the new command under one of the following conditions:
  - Immediately, if the data path is free (that is, there is no data transfer in progress), or if an open-ended data transfer is in progress (byte_count = 0).
  - After completion of the current data transfer, if a predefined data transfer is in progress.

Send Command and Receive Response

Once a new command is loaded in the command path, update_clock_registers_only bit is unset – the command path state machine sends out a command on the device bus; the command path state machine is illustrated in following figure.

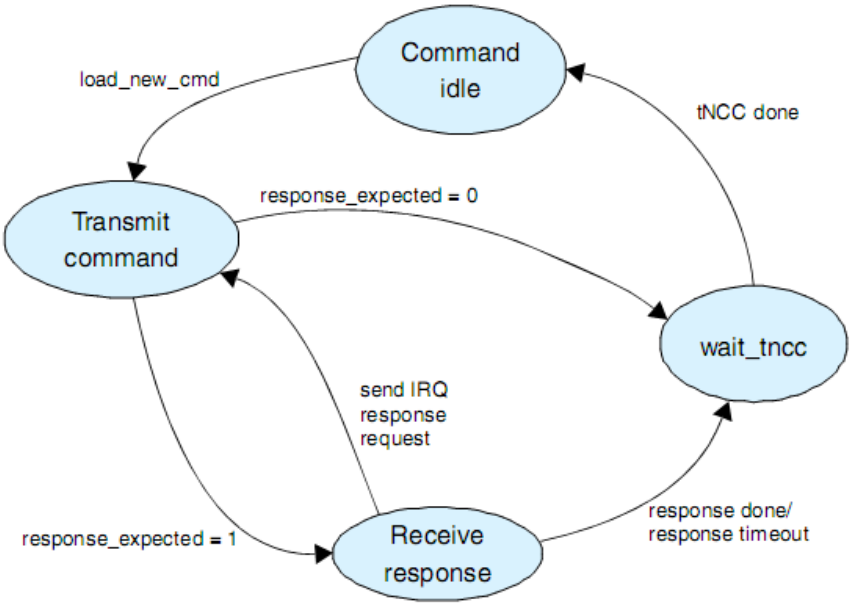


Fig. 10-3 Host Controller Command Path State Machine

The command path state machine performs the following functions, according to Command register bit values:

- send_initialization – Initialization sequence of 80 clocks is sent before sending the command.

- response_expected – Response is expected for the command. After the command is sent out, the command path state machine receives a 48-bit or 136-bit response and sends it to the BIU. If the start bit of the card response is not received within the number of clocks programmed in the timeout register, then the response timeout and command done bit is set in the Raw Interrupt Status register as a signal to the BIU. If the response-expected bit is not set, the command path sends out a command and signals a response done to the BIU; that is, the command done bit is set in the Raw Interrupt Status register.
- response_length – If this bit is set, a 136-bit response is received; if it is not set, a 48-bit response is received.
- check_response_crc – If this bit is set, the command path compares CRC7 received in the response with the internally-generated CRC7. If the two do not match, the response CRC error is signaled to the BIU; that is, the response CRC error bit is set in the Raw Interrupt Status register.

Send Response to BIU

If the response_expected bit is set in the Command register, the received response is sent to the BIU. The Response0 register is updated for a short response, and the Response3, Response2, Response1, and Response0 registers are updated on a long response, after which the Command Done bit is set. If the response is for an auto_stop command sent by the CIU, the response is saved in the Response1 register, after which the Auto Command Done bit is set.

Additionally, the command path checks for the following:

- Transmission bit = 0
- Command index matches command index of the sent command
- End bit = 1 in received card response

The command index is not checked for a 136-bit response or if the check_response_crc bit is unset. For a 136-bit response and reserved CRC 48-bit responses, the command index is reserved-that is, 111111.

Polling Command Completion Signal

The device generates the Command Completion Signal in order to notify the host controller of the normal command completion or command termination.

Command Completion Signal Detection and Interrupt to Host Processor

If the ccs_expected bit is set in the Command register, the Command Completion Signal (CCS) from the device is indicated by setting the Data Transfer Over (DTO) bit in the SDMMC_RINTSTS register. The Host Controller generates a Data Transfer Over (DTO) interrupt if this interrupt is not masked.

Command Completion Signal Timeout

If the command expects a CCS from the device—if the ccs_expected bit is set in the Command register—the command state machine waits for the CCS and remains in a wait_CCSS state. If the device fails to send out the CCS, the host software should implement a timeout mechanism to free the command and data path. The host controller does not implement a hardware timer; it is the responsibility of the host software to maintain a software timer.

In the event of a CCS timeout, the host should issue a CCSD by setting the send_ccsd bit in the SDMMC_CTRL register. The host controller command state machine sends the CCSD to the device and exits to an idle state. After sending the CCSD, the host should also send a CMD12 to the device in order to abort the outstanding command.

Send Command Completion Signal Disable

If the send_ccsd bit is set in the SDMMC_CTRL register, the host sends a Command Completion Signal Disable (CCSD) pattern on the CMD line. The host can send the CCSD while waiting for the CCS or after a CCS timeout happens.

After sending the CCSD pattern, the host sets the Command Done (CD) bit in SDMMC_RINTSTS and also generates an interrupt to the host if the Command Done interrupt is not masked.

2. Data Path

The data path block pops the data FIFO and transmits data on cdata_out during a write data transfer, or it receives data on cdata_in and pushes it into the FIFO during a read data

transfer. The data path loads new data parameters—that is, data expected, read/write data transfer, stream/block transfer, block size, byte count, card type, timeout registers—whenever a data transfer command is not in progress.
If the data_expected bit is set in the Command register, the new command is a data transfer command and the data path starts one of the following:

- Transmit data if the read/write bit = 1
- Data receive if read/write bit = 0

Data Transmit
The data transmit state machine, illustrated in following figure, starts data transmission two clocks after a response for the data write command is received; this occurs even if the command path detects a response error or response CRC error. If a response is not received from the card because of a response timeout, data is not transmitted. Depending upon the value of the transfer_mode bit in the Command register, the data transmit state machine puts data on the card data bus in a stream or in block(s).

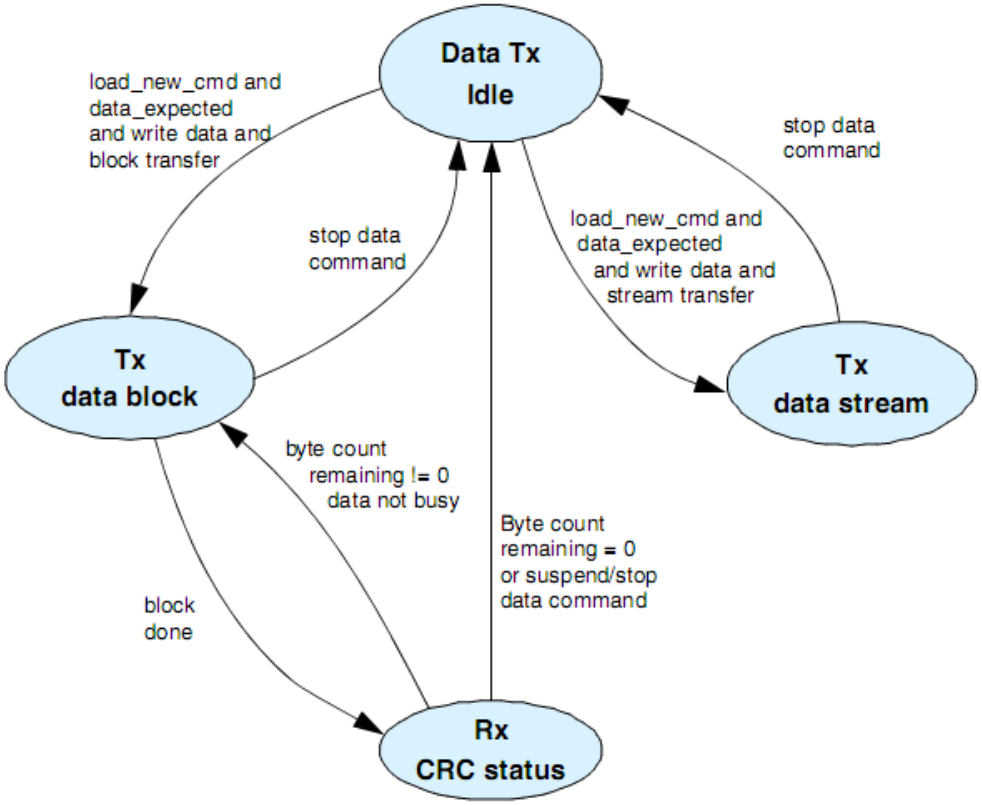


Fig. 10-4 Host Controller Data Transmit State Machine

Stream Data Transmit
If the transfer_mode bit in the Command register is set to 1, it is a stream-write data transfer. The data path pops the FIFO from the BIU and transmits in a stream to the card data bus. If the FIFO becomes empty, the card clock is stopped and restarted once data is available in the FIFO.
If the byte_count register is programmed to 0, it is an open-ended stream-write data transfer. During this data transfer, the data path continuously transmits data in a stream until the host software issues a stop command. A stream data transfer is terminated when the end bit of the stop command and end bit of the data match over two clocks.
If the byte_count register is programmed with a non-zero value and the send_auto_stop bit is set in the Command register, the stop command is internally generated and loaded in the command path when the end bit of the stop command occurs after the last byte of the stream write transfer matches.
This data transfer can also terminate if the host issues a stop command before all the data bytes are transferred to the card bus.

Single Block Data
If the transfer_mode bit in the Command register is set to 0 and the byte_count register value is equal to the value of the block_size register, a single-block write-data transfer

occurs. The data transmit state machine sends data in a single block, where the number of bytes equals the block size, including the internally-generated CRC16.
If the SDMMC_CTYPE register bit for the selected card – indicated by the card_num value in the Command register – is set for a 1-bit, 4-bit, or 8-bit data transfer, the data is transmitted on 1, 4, or 8 data lines, respectively, and CRC16 is separately generated and transmitted for 1, 4, or 8 data lines, respectively.
After a single data block is transmitted, the data transmit state machine receives the CRC status from the card and signals a data transfer to the BIU; this happens when the data-transfer-over bit is set in the SDMMC_RINTSTS register.
If a negative CRC status is received from the card, the data path signals a data CRC error to the BIU by setting the data CRC error bit in the SDMMC_RINTSTS register.
Additionally, if the start bit of the CRC status is not received by two clocks after the end of the data block, a CRC status start bit error is signaled to the BIU by setting the write-no-CRC bit in the SDMMC_RINTSTS register.

Multiple Block Data

A multiple-block write-data transfer occurs if the transfer_mode bit in the Command register is set to 0 and the value in the byte_count register is not equal to the value of the block_size register. The data transmit state machine sends data in blocks, where the number of bytes in a block equals the block size, including the internally-generated CRC16.
If the SDMMC_CTYPE register bit for the selected card – indicated by the card_num value in the Command register – is set to 1-bit, 4-bit, or 8-bit data transfer, the data is transmitted on 1, 4, or 8 data lines, respectively, and CRC16 is separately generated and transmitted on 1, 4, or 8 data lines, respectively.
After one data block is transmitted, the data transmit state machine receives the CRC status from the card. If the remaining byte_count becomes 0, the data path signals to the BIU that the data transfer is done; this happens when the data-transfer-over bit is set in the SDMMC_RINTSTS register.
If the remaining data bytes are greater than 0, the data path state machine starts to transmit another data block.
If a negative CRC status is received from the card, the data path signals a data CRC error to the BIU by setting the data CRC error bit in the SDMMC_RINTSTS register, and continues further data transmission until all the bytes are transmitted.
Additionally, if the CRC status start bit is not received by two clocks after the end of a data block, a CRC status start bit error is signaled to the BIU by setting the write-no-CRC bit in the SDMMC_RINTSTS register; further data transfer is terminated.
If the send_auto_stop bit is set in the Command register, the stop command is internally generated during the transfer of the last data block, where no extra bytes are transferred to the card. The end bit of the stop command may not exactly match the end bit of the CRC status in the last data block.
If the block size is less than 4, 16, or 32 for card data widths of 1 bit, 4 bits, or 8 bits, respectively, the data transmit state machine terminates the data transfer when all the data is transferred, at which time the internally generated stop command is loaded in the command path.
If the byte_count is 0 – the block size must be greater than 0 – it is an open-ended block transfer. The data transmit state machine for this type of data transfer continues the block-write data transfer until the host software issues a stop or abort command.

Data Receive

The data-receive state machine, illustrated in following figure, receives data two clock cycles after the end bit of a data read command, even if the command path detects a response error or response CRC error. If a response is not received from the card because a response timeout occurs, the BIU does not receive a signal that the data transfer is complete; this happens if the command sent by the Host Controller is an illegal operation for the card, which keeps the card from starting a read data transfer.
If data is not received before the data timeout, the data path signals a data timeout to the BIU and an end to the data transfer done. Based on the value of the transfer_mode bit in the Command register, the data-receive state machine gets data from the card data bus in a stream or block(s).

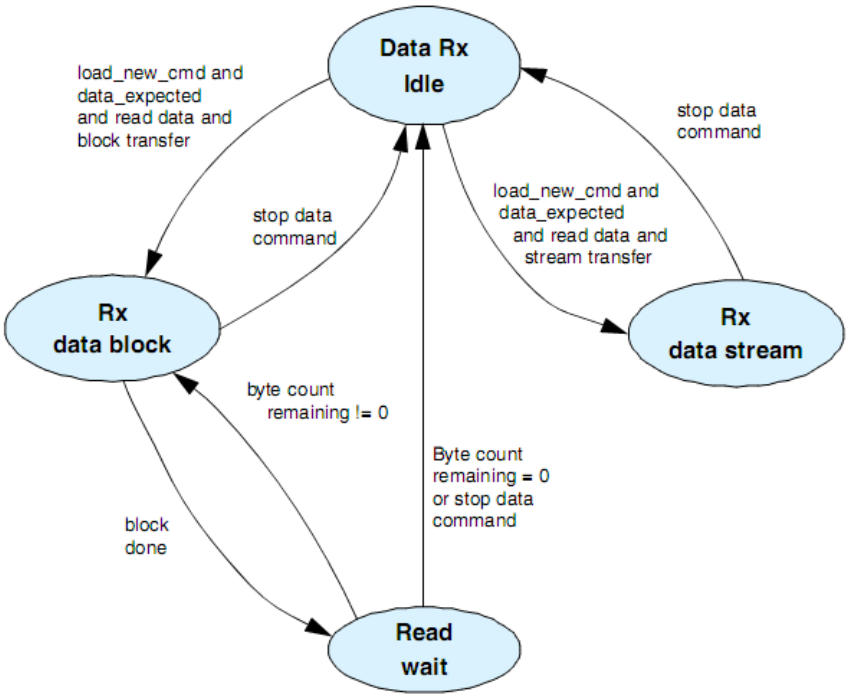


Fig. 10-5 Host Controller Data Receive State Machine

Stream Data Read

A stream-read data transfer occurs if the transfer_mode bit in the Command register equals 1, at which time the data path receives data from the card and pushes it to the FIFO. If the FIFO becomes full, the card clock stops and restarts once the FIFO is no longer full.

An open-ended stream-read data transfer occurs if the byte_count register equals 0. During this type of data transfer, the data path continuously receives data in a stream until the host software issues a stop command. A stream data transfer terminates two clock cycles after the end bit of the stop command.

If the byte_count register contains a non-zero value and the send_auto_stop bit is set in the Command register, a stop command is internally generated and loaded into the command path, where the end bit of the stop command occurs after the last byte of the stream data transfer is received. This data transfer can terminate if the host issues a stop or abort command before all the data bytes are received from the card.

Single-Block Data Read

A single-block read-data transfer occurs if the transfer_mode bit in the Command register is set to 0 and the value of the byte_count register is equal to the value of the block_size register. When a start bit is received before the data times out, data bytes equal to the block size and CRC16 are received and checked with the internally-generated CRC16.

If the SDMMC_CTYPE register bit for the selected card – indicated by the card_num value in the Command register – is set to a 1-bit, 4-bit, or 8-bit data transfer, data is received from 1, 4, or 8 data lines, respectively, and CRC16 is separately generated and checked for 1, 4, or 8 data lines, respectively. If there is a CRC16 mismatch, the data path signals a data CRC error to the BIU. If the received end bit is not 1, the BIU receives an end-bit error.

Multiple-Block Data Read

If the transfer_mode bit in the Command register is set to 0 and the value of the byte_count register is not equal to the value of the block_size register, it is a multiple-block read-data transfer. The data-receive state machine receives data in blocks, where the number of bytes in a block is equal to the block size, including the internally-generated CRC16.

If the SDMMC_CTYPE register bit for the selected card – indicated by the card_num value in the Command register – is set to a 1-bit, 4-bit, or 8-bit data transfer, data is received from 1, 4, or 8 data lines, respectively, and CRC16 is separately generated and checked for 1, 4, or 8 data lines, respectively.

After a data block is received, if the remaining byte_count becomes 0, the data path signals a data transfer to the BIU.

If the remaining data bytes are greater than 0, the data path state machine causes another data block to be received. If CRC16 of a received data block does not match the internally-generated CRC16, a data CRC error to the BIU and data reception continue further data transmission until all bytes are transmitted.
Additionally, if the end of a received data block is not 1, data on the data path signals terminate the bit error to the CIU and the data-receive state machine terminates data reception, waits for data timeout, and signals to the BIU that the data transfer is complete.
If the send_auto_stop bit is set in the Command register, the stop command is internally generated when the last data block is transferred, where no extra bytes are transferred from the card; the end bit of the stop command may not exactly match the end bit of the last data block.
If the requested block size for data transfers to cards is less than 4, 16, or 32 bytes for 1-bit, 4-bit, or 8-bit data transfer modes, respectively, the data-transmit state machine terminates the data transfer when all data is transferred, at which point the internally-generated stop command is loaded in the command path. Data received from the card after that are then ignored by the data path.
If the byte_count is 0—the block size must be greater than 0—it is an open-ended block transfer. For this type of data transfer, the data-receive state machine continues the block-read data transfer until the host software issues a stop or abort command.

Auto-Stop

The Host Controller internally generates a stop command and is loaded in the command path when the send_auto_stop bit is set in the Command register.
The software should set the send_auto_stop bit according to details listed in following table.

Table 10-2 Auto-Stop Generation

Card type	Transfer type	Byte Count	send_auto_stop bit set	Comments
MMC	Stream read	0	No	Open-ended stream
MMC	Stream read	>0	Yes	Auto-stop after all bytes transfer
MMC	Stream write	0	No	Open-ended stream
MMC	Stream write	>0	Yes	Auto-stop after all bytes transfer
MMC	Single-block read	>0	No	Byte count =0 is illegal
MMC	Single-block write	>0	No	Byte count =0 is illegal
MMC	Multiple-block read	0	No	Open-ended multiple block
MMC	Multiple-block read	>0	Yes①	Pre-defined multiple block
MMC	Multiple-block write	0	No	Open-ended multiple block
MMC	Multiple-block write	>0	Yes①	Pre-defined multiple block
SDMEM	Single-block read	>0	No	Byte count =0 is illegal
SDMEM	Single-block write	>0	No	Byte count =0 illegal
SDMEM	Multiple-block read	0	No	Open-ended multiple block
SDMEM	Multiple-block read	>0	Yes	Auto-stop after all bytes transfer
SDMEM	Multiple-block write	0	No	Open-ended multiple block
SDMEM	Multiple-block write	>0	Yes	Auto-stop after all bytes transfer
SDIO	Single-block read	>0	No	Byte count =0 is illegal
SDIO	Single-block write	>0	No	Byte count =0 illegal
SDIO	Multiple-block read	0	No	Open-ended multiple block

Card type	Transfer type	Byte Count	send_auto_stop bit set	Comments
SDIO	Multiple-block read	>0	No	Pre-defined multiple block
SDIO	Multiple-block write	0	No	Open-ended multiple block
SDIO	Multiple-block write	>0	No	Pre-defined multiple block

①:*The condition under which the transfer mode is set to block transfer and byte_count is equal to block size is treated as a single-block data transfer command for both MMC and SD cards. If byte_count = n*block_size (n = 2, 3, …), the condition is treated as a predefined multiple-block data transfer command. In the case of an MMC card, the host software can perform a predefined data transfer in two ways: 1) Issue the CMD23 command before issuing CMD18/CMD25 commands to the card – in this case, issue MD18/CMD25 commands without setting the send_auto_stop bit. 2) Issue CMD18/CMD25 commands without issuing CMD23 command to the card, with the send_auto_stop bit set. In this case, the multiple-block data transfer is terminated by an internally-generated auto-stop command after the programmed byte count.*

The following list conditions for the auto-stop command.

- Stream read for MMC card with byte count greater than 0 – The Host Controller generates an internal stop command and loads it into the command path so that the end bit of the stop command is sent out when the last byte of data is read from the card and no extra data byte is received. If the byte count is less than 6 (48 bits), a few extra data bytes are received from the card before the end bit of the stop command is sent.
- Stream write for MMC card with byte count greater than 0 - The Host Controller generates an internal stop command and loads it into the command path so that the end bit of the stop command is sent when the last byte of data is transmitted on the card bus and no extra data byte is transmitted. If the byte count is less than 6 (48 bits), the data path transmits the data last in order to meet the above condition.
- Multiple-block read memory for SD card with byte count greater than 0 – If the block size is less than 4 (single-bit data bus), 16 (4-bit data bus), or 32 (8-bit data bus), the auto-stop command is loaded in the command path after all the bytes are read. Otherwise, the top command is loaded in the command path so that the end bit of the stop command is sent after the last data block is received.
- Multiple-block write memory for SD card with byte count greater than 0 – If the block size is less than 3 (single-bit data bus), 12 (4-bit data bus), or 24 (8-bit data bus), the auto-stop command is loaded in the command path after all data blocks are transmitted. Otherwise, the stop command is loaded in the command path so that the end bit of the stop command is sent after the end bit of the CRC status is received.
- Precaution for host software during auto-stop – Whenever an auto-stop command is issued, the host software should not issue a new command to the SD/MMC device until the auto-stop is sent by the Host Controller and the data transfer is complete. If the host issues a new command during a data transfer with the auto-stop in progress, an auto-stop command may be sent after the new command is sent and its response is received; this can delay sending the stop command, which transfers extra data bytes. For a stream write, extra data bytes are erroneous data that can corrupt the card data. If the host wants to terminate the data transfer before the data transfer is complete, it can issue a stop or abort command, in which case the Host Controller does not generate an auto-stop command.

3. Non-Data Transfer Commands that Use Data Path

Some non-data transfer commands (non-read/write commands) also use the data path. Following table lists the commands and register programming requirements for them.

Table 10-3 Non-data Transfer Commands and Requirements

Base Address [12:8]	CMD 27	CMD 30	CMD 42	ACMD 13	ACMD 22	ACMD 51
Command register programming						
cmd_index	6'h1B	6'h1E	6'h2A	6'h0D	6'h16	6'h33

Base Address [12:8]	CMD 27	CMD 30	CMD 42	ACMD 13	ACMD 22	ACMD 51
response_expect	1	1	1	1	1	1
rResponse_length	0	0	0	0	0	0
check_response_crc	1	1	1	1	1	1
data_expected	1	1	1	1	1	1
read/write	1	0	1	0	0	0
transfer_mode	0	0	0	0	0	0
send_auto_stop	0	0	0	0	0	0
wait_prevdata_complete	0	0	0	0	0	0
stop_abort_cmd	0	0	0	0	0	0
Command Argument register programming						
	stuff bits	32-bit write protect data address	stuff bits	stuff bits	stuff bits	stuff bits
Block Size register programming						
	16	4	Num_bytes①	64	4	8
Byte Count register programming						
	16	4	Num_bytes①	64	4	8

①: *Num_bytes = No. of bytes specified as per the lock card data structure (Refer to the SD specification and the MMC specification)*

4. SDIO Interrupt Control

Interrupts for SD cards are reported to the BIU by asserting an interrupt signal for two clock cycles. SDIO cards signal an interrupt by asserting cdata_in low during the interrupt period; an interrupt period for the selected card is determined by the interrupt control state machine. An interrupt period is always valid for non-active or non-selected cards, and 1-bit data mode for the selected card. An interrupt period for a wide-bus active or selected card is valid for the following conditions:

- Card is idle
- Non-data transfer command in progress
- Third clock after end bit of data block between two data blocks
- From two clocks after end bit of last data until end bit of next data transfer command

Bear in mind that, in the following situations, the controller does not sample the SDIO interrupt of the selected card when the card data width is 4 bits. Since the SDIO interrupt is level-triggered, it is sampled in a further interrupt period and the host does not lose any SDIO interrupt from the card.

- Read/Write Resume – The CIU treats the resume command as a normal data transfer command. SDIO interrupts during the resume command are handled similarly to other data commands. According to the SDIO specification, for the normal data command the interrupt period ends after the command end bit of the data command; for the resume command, it ends after the response end bit. In the case of the resume command, the Controller stops the interrupt sampling period after the resume command end bit, instead of stopping after the response end bit of the resume command.
- Suspend during read transfer – If the read data transfer is suspended by the host, the host sets the abort_read_data bit in the controller to reset the data state machine. In the CIU, the SDIO interrupts are handled such that the interrupt sampling starts after the abort_read_data bit is set by the host. In this case the controller does not sample SDIO interrupts between the period from response of the suspend command to setting the abort_read_data bit, and starts sampling after setting the abort_read_data bit.

5. Clock Control

The clock control block provides different clock frequencies required for SD/MMC cards. The cclk_in signal is the source clock (cclk_in >= card max operating frequency) for clock divider of the clock control block. This source clock (cclk_in) is used to generate different card clock frequencies (cclk_out). The card clock can have different clock frequencies, since

the card can be a low-speed card or a full-speed card. The Host Controller provides one clock signal (cclk_out).
The clock frequency of a card depends on the following clock control registers:

- Clock Divider register – Internal clock dividers are used to generate different clock frequencies required for card. The division factor for each clock divider can be programmed by writing to the Clock Divider register. The clock divider is an 8-bit value that provides a clock division factor from 1 to 510; a value of 0 represents a clock-divider bypass, a value of 1 represents a divide by 2, a value of 2 represents a divide by 4, and so on.
- Clock Control register – cclk_out can be enabled or disabled for each card under the following conditions:
  - clk_enable – cclk_out for a card is enabled if the clk_enable bit for a card in the Clock Control register is programmed (set to 1) or disabled (set to 0).
  - Low-power mode – Low-power mode of a card can be enabled by setting the low-power mode bit of the Clock Control register to 1. If low-power mode is enabled to save card power, the cclk_out is disabled when the card is idle for at least 8 card clock cycles. It is enabled when a new command is loaded and the command path goes to a non-idle state.

Additionally, cclk_out is disabled when an internal FIFO is full – card read (no more data can be received from card) – or when the FIFO is empty – card write (no data is available for transmission). This helps to avoid FIFO overrun and underrun conditions. It is used by the command and data path to qualify cclk_in for driving outputs and sampling inputs at the programmed clock frequency for the selected card, according to the Clock Divider and Clock Source register values.
Under the following conditions, the card clock is stopped or disabled, along with the active clk_en, for the selected card:

- Clock can be disabled by writing to Clock Enable register (clk_en bit = 1).
- If low-power mode is selected and card is idle, or not selected for 8 clocks.
- FIFO is full and data path cannot accept more data from the card and data transfer is incomplete –to avoid FIFO overrun.
- FIFO is empty and data path cannot transmit more data to the card and data transfer is incomplete – to avoid FIFO underrun.

6. Error Detection

- Response
  - Response timeout – Response expected with response start bit is not received within programmed number of clocks in timeout register.
  - Response CRC error – Response is expected and check response CRC requested; response CRC7 does not match with the internally-generated CRC7.
  - Response error – Response transmission bit is not 0, command index does not match with the command index of the send command, or response end bit is not 1.
- Data transmit
  - No CRC status – During a write data transfer, if the CRC status start bit is not received two clocks after the end bit of the data block is sent out, the data path does the following:
    - Signals no CRC status error to the BIU
    - Terminates further data transfer
    - Signals data transfer done to the BIU
  - Negative CRC – If the CRC status received after the write data block is negative (that is, not 010), a data CRC error is signaled to the BIU and further data transfer is continued.
  - Data starvation due to empty FIFO – If the FIFO becomes empty during a write data transmission, or if the card clock is stopped and the FIFO remains empty for data timeout clocks, then a data-starvation error is signaled to the BIU and the data path continues to wait for data in the FIFO.
- Data receive
  - Data timeout – During a read-data transfer, if the data start bit is not received before the number of clocks that were programmed in the timeout register, the

data path does the following:
- Signals data-timeout error to the BIU
- Terminates further data transfer
- Signals data transfer done to BIU

- Data start bit error – During a 4-bit or 8-bit read-data transfer, if the all-bit data line does not have a start bit, the data path signals a data start bit error to the BIU and waits for a data timeout, after which it signals that the data transfer is done.
- Data CRC error – During a read-data-block transfer, if the CRC16 received does not match with the internally generated CRC16, the data path signals a data CRC error to the BIU and continues further data transfer.
- Data end-bit error – During a read-data transfer, if the end bit of the received data is not 1, the data path signals an end-bit error to the BIU, terminates further data transfer, and signals to the BIU that the data transfer is done.
- Data starvation due to FIFO full – During a read data transmission and when the FIFO becomes full, the card clock is stopped. If the FIFO remains full for data timeout clocks, a data starvation error is signaled to the BIU (Data Starvation by Host Timeout bit is set in SDMMC_RINTSTS register) and the data path continues to wait for the FIFO to start to empty.

10.4 Register Description

10.4.1 Registers Summary

Name	Offset	Size	Reset Value	Description
SDMMC_CTRL	0x0000	W	0x01000000	Control register
SDMMC_PWREN	0x0004	W	0x00000000	Power-enable register
SDMMC_CLKDIV	0x0008	W	0x00000000	Clock-divider register
SDMMC_CLKSRC	0x000c	W	0x00000000	SD clock source register
SDMMC_CLKENA	0x0010	W	0x00000000	Clock-enable register
SDMMC_TMOUT	0x0014	W	0xffffff40	Time-out register
SDMMC_CTYPE	0x0018	W	0x00000000	Card-type register
SDMMC_BLKSIZ	0x001c	W	0x00000200	Block-size register
SDMMC_BYTCNT	0x0020	W	0x00000200	Byte-count register
SDMMC_INTMASK	0x0024	W	0x00000000	Interrupt-mask register
SDMMC_CMDARG	0x0028	W	0x00000000	Command-argument register
SDMMC_CMD	0x002c	W	0x00000000	Command register
SDMMC_RESP0	0x0030	W	0x00000000	Response-0 register
SDMMC_RESP1	0x0034	W	0x00000000	Response-1 register
SDMMC_RESP2	0x0038	W	0x00000000	Response-2 register
SDMMC_RESP3	0x003c	W	0x00000000	Response-3 register
SDMMC_MINTSTS	0x0040	W	0x00000000	Masked interrupt-status register
SDMMC_RINTSTS	0x0044	W	0x00000000	Raw interrupt-status register
SDMMC_STATUS	0x0048	W	0x00000406	Status register
SDMMC_FIFOTH	0x004c	W	0x00000000	FIFO threshold register
SDMMC_CDETECT	0x0050	W	0x00000000	Card-detect register
SDMMC_WRTPRT	0x0054	W	0x00000000	Write-protect register
SDMMC_TCBCNT	0x005c	W	0x00000000	Transferred CIU card byte count
SDMMC_TBBCNT	0x0060	W	0x00000000	Transferred host/DMA to/from BIU-FIFO byte count
SDMMC_DEBNCE	0x0064	W	0x00ffffff	Card detect debounce register

Name	Offset	Size	Reset Value	Description
SDMMC_USRID	0x0068	W	0x07967797	User ID register
SDMMC_VERID	0x006c	W	0x5342270a	Version ID register
SDMMC_HCON	0x0070	W	0x00000000	Hardware configuration register
SDMMC_UHS_REG	0x0074	W	0x00000000	UHS-1 register
SDMMC_RST_n	0x0078	W	0x00000001	Hardware reset register
SDMMC_PLDMND	0x0084	W	0x00000000	Poll demand register
SDMMC_CARDTHRCTL	0x0100	W	0x00000000	Card read threshold enable register
SDMMC_BACK_END_POWER	0x0104	W	0x00000000	Back-end power register
SDMMC_EMMC_DDR_REG	0x010c	W	0x00000000	eMMC4.5 DDR start bit detection control register
SDMMC_FIFO_BASE	0x0200	W	0x00000000	FIFO base address register

*Notes:Size:**B**- Byte (8 bits) access, **HW**- Half WORD (16 bits) access, **W**-WORD (32 bits) access*

10.4.2 Detail Register Description

SDMMC_CTRL

Address: Operational Base + offset (0x0000)

Control register

Bit	Attr	Reset Value	Description
31:26	RO	0x0	reserved
25	RW	0x0	use_internal_dmac Present only for the Internal DMAC configuration; else, it is reserved. 0: The host performs data transfers through the slave interface 1: Internal DMAC used for data transfer
24:12	RO	0x0	reserved
11	RW	0x0	ceata_device_interrupt_status 0: Interrupts not enabled in CE-ATA device (nIEN = 1 in ATA control register) 1: Interrupts are enabled in CE-ATA device (nIEN = 0 in ATA control register) Software should appropriately write to this bit after power-on reset or any other reset to CE-ATA device. After reset, usually CE-ATA device interrupt is disabled (nIEN = 1). If the host enables CE-ATA device interrupt, then software should set this bit.

Bit	Attr	Reset Value	Description
10	RW	0x0	send_auto_stop_ccsd 0: Clear bit if Mobile Storage Host Controller does not reset the bit. 1: Send internally generated STOP after sending CCSD to CE-ATA device. NOTE: Always set send_auto_stop_ccsd and send_ccsd bits together send_auto_stop_ccsd should not be set independent of send_ccsd. When set, Mobile Storage Host Controller automatically sends internally- generated STOP command (CMD12) to CE-ATA device. After sending internally-generated STOP command, Auto Command Done (ACD) in SDMMC_RINTSTS is set and generates interrupt to host if Auto Command Done interrupt is not masked. After sending the CCSD, Mobile Storage Host Controller automatically clears send_auto_stop_ccsd bit.
9	RW	0x0	send_ccsd 0: Clear bit if Mobile Storage Host Controller does not reset the bit. 1: Send Command Completion Signal Disable (CCSD) to CE-ATA device When set, Mobile Storage Host Controller sends CCSD to CE-ATA device. Software sets this bit only if current command is expecting CCS (that is, RW_BLK) and interrupts are enabled in CE-ATA device. Once the CCSD pattern is sent to device, Mobile Storage Host Controller automatically clears send_ccsd bit. It also sets Command Done (CD) bit in SDMMC_RINTSTS register and generates interrupt to host if Command Done interrupt is not masked. NOTE: Once send_ccsd bit is set, it takes two card clock cycles to drive the CCSD on the CMD line. Due to this, during the boundary conditions it may happen that CCSD is sent to the CE-ATA device, even if the device signalled CCS
8	RW	0x0	abort_read_data 0: no change 1: after suspend command is issued during read-transfer, software polls card to find when suspend happened. Once suspend occurs, software sets bit to reset data state-machine, which is waiting for next block of data. Bit automatically clears once data state machine resets to idle. Used in SDIO card suspend sequence.

Bit	Attr	Reset Value	Description
7	RW	0x0	send_irq_response 0: no change 1: send auto IRQ response Bit automatically clears once response is sent. To wait for MMC card interrupts, host issues CMD40, and SDMMC Controller waits for interrupt response from MMC card(s). In meantime, if host wants SDMMC Controller to exit waiting for interrupt state, it can set this bit, at which time SDMMC Controller command state-machine sends CMD40 response on bus and returns to idle state.
6	RW	0x0	read_wait 0: clear read wait 1: assert read wait For sending read-wait to SDIO cards
5	RW	0x0	dma_enable 0: disable DMA transfer mode 1: enable DMA transfer mode Even when DMA mode is enabled, host can still push/pop data into or from FIFO; this should not happen during the normal operation. If there is simultaneous FIFO access from host/DMA, the data coherency is lost. Also, there is no arbitration inside SDMMC Controller to prioritize simultaneous host/DMA access.
4	RW	0x0	int_enable Global interrupt enable/disable bit: 0: disable interrupts 1: enable interrupts The int port is 1 only when this bit is 1 and one or more unmasked interrupts are set.
3	RO	0x0	reserved
2	W1C	0x0	dma_reset 0: no change 1: reset internal DMA interface control logic To reset DMA interface, firmware should set bit to 1. This bit is auto-cleared after two AHB clocks.
1	W1C	0x0	fifo_reset 0: no change 1: reset to data FIFO To reset FIFO pointers To reset FIFO, firmware should set bit to 1. This bit is auto-cleared after completion of reset operation

Bit	Attr	Reset Value	Description
0	W1C	0x0	controller_reset 0: no change 1: reset SDMMC controller To reset controller, firmware should set bit to 1. This bit is auto-cleared after two AHB and two cclk_in clock cycles. This resets: a. BIU/CIU interface b. CIU and state machines c. abort_read_data, send_irq_response, and read_wait bits of Control register d. start_cmd bit of Command register Does not affect any registers or DMA interface, or FIFO or host interrupts

SDMMC_PWREN
Address: Operational Base + offset (0x0004)
Power-enable register

Bit	Attr	Reset Value	Description
31:1	RO	0x0	reserved
0	RW	0x0	power_enable Power on/off switch for the card. Once power is turned on, firmware should wait for regulator/switch ramp-up time before trying to initialize card. 0: power off 1: power on Bit values output to card_power_en port.

SDMMC_CLKDIV
Address: Operational Base + offset (0x0008)
Clock-divider register

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:0	RW	0x00	clk_divider0 Clock divider-0 value. Clock division is 2*n. For example, value of 0 means divide by 2*0 = 0 (no division, bypass), value of 1 means divide by 2*1 = 2, value of "ff" means divide by 2*255 = 510, and so on.

SDMMC_CLKSRC
Address: Operational Base + offset (0x000c)
SD clock source register

Bit	Attr	Reset Value	Description
31:2	RO	0x0	reserved

Bit	Attr	Reset Value	Description
1:0	RW	0x0	clk_source Clock divider source for up to 16 SD cards supported. Each card has two bits assigned to it. For example, bits[1:0] assigned for card-0, which maps and internally routes clock divider[3:0] outputs to cclk_out[15:0] pins, depending on bit value. 00: Clock divider 0 The cclk_out is always from clock divider 0, and this register is not implemented.

SDMMC_CLKENA
Address: Operational Base + offset (0x0010)
Clock-enable register

Bit	Attr	Reset Value	Description
31:17	RO	0x0	reserved
16	RW	0x0	cclk_low_power Low-power control for SD card clock and MMC card clock supported. 0: non-low-power mode 1: low-power mode; stop clock when card in IDLE (should be normally set to only MMC and SD memory cards; for SDIO cards, if interrupts must be detected, clock should not be stopped).
15:1	RO	0x0	reserved
0	RW	0x0	cclk_enable Clock-enable control for SD card clock and MMC card clock supported. 0: clock disabled 1: clock enabled

SDMMC_TMOUT
Address: Operational Base + offset (0x0014)
Time-out register

Bit	Attr	Reset Value	Description
31:8	RW	0xffffff	data_timeout Value for card Data Read Timeout; same value also used for Data Starvation by Host timeout. Value is in number of card output clocks cclk_out of selected card. Note: The software timer should be used if the timeout value is in the order of 100 ms. In this case, read data timeout interrupt needs to be disabled.
7:0	RW	0x40	response_timeout Response timeout value. Value is in number of card output clocks –cclk_out.

SDMMC_CTYPE

Address: Operational Base + offset (0x0018)
Card-type register

Bit	Attr	Reset Value	Description
31:17	RO	0x0	reserved
16	RW	0x0	card_width_8 Indicates if card is 8-bit: 0: non 8-bit mode 1: 8-bit mode
15:1	RO	0x0	reserved
0	RW	0x0	card_width Indicates if card is 1-bit or 4-bit: 0: 1-bit mode 1: 4-bit mode

SDMMC_BLKSIZ
Address: Operational Base + offset (0x001c)
Block-size register

Bit	Attr	Reset Value	Description
31:16	RO	0x0	reserved
15:0	RW	0x0200	block_size Block size

SDMMC_BYTCNT
Address: Operational Base + offset (0x0020)
Byte-count register

Bit	Attr	Reset Value	Description
31:0	RW	0x00000200	byte_count Number of bytes to be transferred; should be integer multiple of Block Size for block transfers. For undefined number of byte transfers, byte count should be set to 0. When byte count is set to 0, it is responsibility of host to explicitly send stop/abort command to terminate data transfer.

SDMMC_INTMASK
Address: Operational Base + offset (0x0024)
Interrupt-mask register

Bit	Attr	Reset Value	Description
31:25	RO	0x0	reserved
24	RW	0x0	sdio_int_mask Mask SDIO interrupts. When masked, SDIO interrupt detection for that card is disabled. A 0 masks an interrupt, and 1 enables an interrupt.
23:17	RO	0x0	reserved
16	RW	0x0	data_nobusy_int_mask 0: data no busy interrupt not masked 1: data no busy interrupt masked

Bit	Attr	Reset Value	Description
15:0	RW	0x0000	int_mask Bits used to mask unwanted interrupts. Value of 0 masks interrupt; value of 1 enables interrupt. [15]: End-bit error (read)/Write no CRC (EBE) [14]: Auto command done (ACD) [13]: Start-bit error (SBE) [12]: Hardware locked write error (HLE) [11]: FIFO underrun/overrun error (FRUN) [10]: Data starvation-by-host timeout (HTO) /Volt_switch_int [9]: Data read timeout (DRTO) [8]: Response timeout (RTO) [7]: Data CRC error (DCRC) [6]: Response CRC error (RCRC) [5]: Receive FIFO data request (RXDR) [4]: Transmit FIFO data request (TXDR) [3]: Data transfer over (DTO) [2]: Command done (CD) [1]: Response error (RE) [0]: Card detect (CD)

SDMMC_CMDARG
Address: Operational Base + offset (0x0028)
Command-argument register

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	cmd_arg Value indicates command argument to be passed to card.

SDMMC_CMD
Address: Operational Base + offset (0x002c)
Command register

Bit	Attr	Reset Value	Description
31	RW	0x0	start_cmd Start command. Once command is taken by CIU, bit is cleared. When bit is set, host should not attempt to write to any command registers. If write is attempted, hardware lock error is set in raw interrupt register. Once command is sent and response is received from SD_MMC cards, Command Done bit is set in raw interrupt register.
30	RO	0x0	reserved

Bit	Attr	Reset Value	Description
29	RW	0x0	use_hold_reg Use Hold Register 0: CMD and DATA sent to card bypassing HOLD Register 1: CMD and DATA sent to card through the HOLD Register Note: a. Set to 1'b1 for SDR12 and SDR25 (with non-zero phase-shifted cclk_in_drv); zero phase shift is not allowed in these modes. b. Set to 1'b0 for SDR50, SDR104, and DDR50 (with zero phase-shifted cclk_in_drv). c. Set to 1'b1 for SDR50, SDR104, and DDR50 (with non-zero phase-shifted cclk_in_drv).
28	RW	0x0	volt_switch Voltage switch bit. 0: no voltage switching 1: voltage switching enabled; must be set for CMD11 only
27	RW	0x0	boot_mode Boot Mode. 0: mandatory Boot operation 1: alternate Boot operation
26	RW	0x0	disable_boot Disable Boot. When software sets this bit along with start_cmd, CIU terminates the boot operation. Do NOT set disable_boot and enable_boot together.
25	RW	0x0	expect_boot_ack Expect Boot Acknowledge. When Software sets this bit along with enable_boot, CIU expects a boot acknowledge start pattern of 0-1-0 from the selected card.
24	RW	0x0	enable_boot Enable Boot—this bit should be set only for mandatory boot mode.When Software sets this bit along with start_cmd, CIU starts the boot sequence for the corresponding card by asserting the CMD line low. Do NOT set disable_boot and enable_boot together.
23	RW	0x0	ccs_expected 0: Interrupts are not enabled in CE-ATA device (nIEN = 1 in ATA control register), or command does not expect CCS from device 1: Interrupts are enabled in CE-ATA device (nIEN = 0), and RW_BLK command expects command completion signal from CE-ATA device. If the command expects Command Completion Signal (CCS) from the CE-ATA device, the software should set this control bit. Mobile Storage Host Controller sets Data Transfer Over (DTO) bit in SDMMC_RINTSTS register and generates interrupt to host if Data Transfer Over interrupt is not masked.

Bit	Attr	Reset Value	Description
22	RW	0x0	read_ceata_device 0: Host is not performing read access (RW_REG or RW_BLK) towards CE-ATA device 1: Host is performing read access (RW_REG or RW_BLK) towards CE-ATA device Software should set this bit to indicate that CE-ATA device is being accessed for read transfer. This bit is used to disable read data timeout indication while performing CE-ATA read transfers. Maximum value of I/O transmission delay can be no less than 10 seconds. Mobile Storage Host Controller should not indicate read data timeout while waiting for data from CE-ATA device.
21	RW	0x0	update_clock_registers_only 0: normal command sequence 1: do not send commands, just update clock register value into card clock domain Following register values transferred into card clock domain: SDMMC_CLKDIV, SDMMC_CLRSRC, SDMMC_CLKENA. Changes card clocks (change frequency, truncate off or on, and set low-frequency mode); provided in order to change clock frequency or stop clock without having to send command to cards. During normal command sequence, when update_clock_registers_only = 0, following control registers are transferred from BIU to CIU: SDMMC_CMD, SDMMC_CMDARG, SDMMC_TMOUT, SDMMC_CTYPE, SDMMC_BLKSIZ, SDMMC_BYTCNT. CIU uses new register values for new command sequence to card. When bit is set, there are no Command Done interrupts because no command is sent to SD_MMC_CEATA cards.
20:16	RO	0x0	reserved
15	RW	0x0	send_initialization 0: do not send initialization sequence (80 clocks of 1) before sending this command 1: send initialization sequence before sending this command After power on, 80 clocks must be sent to card for initialization before sending any commands to card. Bit should be set while sending first command to card so that controller will initialize clocks before sending command to card. This bit should not be set for either of the boot modes (alternate or mandatory).

Bit	Attr	Reset Value	Description
14	RW	0x0	stop_abort_cmd 0: neither stop nor abort command to stop current data transfer in progress. If abort is sent to function-number currently selected or not in data-transfer mode, then bit should be set to 0. 1: stop or abort command intended to stop current data transfer in progress. When open-ended or predefined data transfer is in progress, and host issues stop or abort command to stop data transfer, bit should be set so that command/data state-machines of CIU can return correctly to idle state. This is also applicable for Boot mode transfers. To Abort boot mode, this bit should be set along with CMD[26] = disable_boot.
13	RW	0x0	wait_prvdata_complete 0: send command at once, even if previous data transfer has not completed 1: wait for previous data transfer completion before sending command The wait_prvdata_complete = 0 option typically used to query status of card during data transfer or to stop current data transfer; card_number should be same as in previous command.
12	RW	0x0	send_auto_stop 0: no stop command sent at end of data transfer 1: send stop command at end of data transfer When set, SDMMC Controller sends stop command to SD_MMC cards at end of data transfer. a. when send_auto_stop bit should be set, since some data transfers do not need explicit stop commands b. open-ended transfers that software should explicitly send to stop command Additionally, when "resume" is sent to resume –suspended memory access of SD-Combo card –bit should be set correctly if suspended data transfer needs send_auto_stop. Don't care if no data expected from card.
11	RW	0x0	transfer_mode 0: block data transfer command 1: stream data transfer command Don't care if no data expected.
10	RW	0x0	wr 0: read from card 1: write to card Don't care if no data expected from card.
9	RW	0x0	data_expected 0: no data transfer expected (read/write) 1: data transfer expected (read/write)

Bit	Attr	Reset Value	Description
8	RW	0x0	check_response_crc 0: do not check response CRC 1: check response CRC Some of command responses do not return valid CRC bits. Software should disable CRC checks for those commands in order to disable CRC checking by controller
7	RW	0x0	response_length 0: short response expected from card 1: long response expected from card
6	RW	0x0	response_expect 0: no response expected from card 1: response expected from card
5:0	RW	0x00	cmd_index Command index

SDMMC_RESP0
Address: Operational Base + offset (0x0030)
Response-0 register

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	response0 Bit[31:0] of response

SDMMC_RESP1
Address: Operational Base + offset (0x0034)
Response-1 register

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	Response1 Register represents bit[63:32] of long response. When CIU sends auto-stop command, then response is saved in register. Response for previous command sent by host is still preserved in Response 0 register. Additional auto-stop issued only for data transfer commands, and response type is always "short" for them.

SDMMC_RESP2
Address: Operational Base + offset (0x0038)
Response-2 register

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	response2 Bit[95:64] of long response

SDMMC_RESP3
Address: Operational Base + offset (0x003c)
Response-3 register

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	response3 Bit[127:96] of long response

SDMMC_MINTSTS
Address: Operational Base + offset (0x0040)
Masked interrupt-status register

Bit	Attr	Reset Value	Description
31:25	RO	0x0	reserved
24	RO	0x0	sdio_interrupt Interrupt from SDIO card; SDIO interrupt for card enabled only if corresponding sdio_int_mask bit is set in Interrupt mask register (mask bit 1 enables interrupt; 0 masks interrupt). 0: no SDIO interrupt from card 1: SDIO interrupt from card
23:17	RO	0x0	reserved
16	RW	0x0	data_nobusy_int_status Data no busy Interrupt Status
15:0	RO	0x0000	int_status Interrupt enabled only if corresponding bit in interrupt mask register is set. [15]: End-bit error (read)/Write no CRC (EBE) [14]: Auto command done (ACD) [13]: Start-bit error (SBE) [12]: Hardware locked write error (HLE) [11]: FIFO underrun/overrun error (FRUN) [10]: Data starvation-by-host timeout (HTO) /Volt_switch_int [9]: Data read timeout (DRTO) [8]: Response timeout (RTO) [7]: Data CRC error (DCRC) [6]: Response CRC error (RCRC) [5]: Receive FIFO data request (RXDR) [4]: Transmit FIFO data request (TXDR) [3]: Data transfer over (DTO) [2]: Command done (CD) [1]: Response error (RE) [0]: Card detect (CD)

SDMMC_RINTSTS
Address: Operational Base + offset (0x0044)
Raw interrupt-status register

Bit	Attr	Reset Value	Description
31:25	RO	0x0	reserved

Bit	Attr	Reset Value	Description
24	RO	0x0	sdio_interrupt Interrupt from SDIO card; Writes to these bits clear them. Value of 1 clears bit and 0 leaves bit intact. 0: no SDIO interrupt from card 1: SDIO interrupt from card
23:17	RO	0x0	reserved
16	RW	0x0	data_nobusy_int_status Data no busy interrupt status
15:0	RO	0x0000	int_status Writes to bits clear status bit. Value of 1 clears status bit, and value of 0 leaves bit intact. Bits are logged regardless of interrupt mask status. [15]: End-bit error (read)/Write no CRC (EBE) [14]: Auto command done (ACD) [13]: Start-bit error (SBE) [12]: Hardware locked write error (HLE) [11]: FIFO underrun/overrun error (FRUN) [10]: Data starvation-by-host timeout (HTO) /Volt_switch_int [9]: Data read timeout (DRTO) [8]: Response timeout (RTO) [7]: Data CRC error (DCRC) [6]: Response CRC error (RCRC) [5]: Receive FIFO data request (RXDR) [4]: Transmit FIFO data request (TXDR) [3]: Data transfer over (DTO) [2]: Command done (CD) [1]: Response error (RE) [0]: Card detect (CD)

SDMMC_STATUS
Address: Operational Base + offset (0x0048)
Status register

Bit	Attr	Reset Value	Description
31	RO	0x0	dma_req DMA request signal state
30	RO	0x0	dma_ack DMA acknowledge signal state
29:17	RO	0x0000	fifo_count Number of filled locations in FIFO
16:11	RO	0x00	response_index Index of previous response, including any auto-stop sent by core
10	RO	0x1	data_state_mc_busy Data transmit or receive state-machine is busy

Bit	Attr	Reset Value	Description
9	RO	0x0	data_busy Inverted version of raw selected card_data[0] 0: card data not busy 1: card data busy default value is 1 or 0 depending on cdata_in
8	RO	0x0	data_3_status Raw selected card_data[3]; checks whether card is present 0: card not present 1: card present default value is 1 or 0 depending on cdata_in
7:4	RO	0x0	command_fsm_states Command FSM states: 0: idle 1: send init sequence 2: Tx cmd start bit 3: Tx cmd tx bit 4: Tx cmd index + arg 5: Tx cmd crc7 6: Tx cmd end bit 7: Rx resp start bit 8: Rx resp IRQ response 9: Rx resp tx bit 10: Rx resp cmd idx 11: Rx resp data 12: Rx resp crc7 13: Rx resp end bit 14: Cmd path wait NCC 15: Wait; CMD-to-response turnaround The command FSM state is represented using 19 bits. The SDMMC_STATUS[7:4] has 4 bits to represent the command FSM states. Using these 4 bits, only 16 states can be represented. Thus three states cannot be represented in the SDMMC_STATUS[7:4] register. The three states that are not represented in the SDMMC_STATUS Register[7:4] are: a. Bit 16 –Wait for CCS b. Bit 17 –Send CCSD c. Bit 18 –Boot Mode Due to this, while command FSM is in "Wait for CCS state" or "Send CCSD" or "Boot Mode", the SDMMC_STATUS register indicates status as 0 for the bit field [7:4].
3	RO	0x0	fifo_full FIFO is full status
2	RO	0x1	fifo_empty FIFO is empty status

Bit	Attr	Reset Value	Description
1	RO	0x1	fifo_tx_watermark FIFO reached Transmit watermark level; not qualified with data transfer
0	RO	0x0	fifo_rx_watermark FIFO reached Receive watermark level; not qualified with data transfer

SDMMC_FIFOTH
Address: Operational Base + offset (0x004c)
FIFO threshold register

Bit	Attr	Reset Value	Description
31	RO	0x0	reserved

Bit	Attr	Reset Value	Description
30:28	RW	0x0	dma_mutiple_transaction_size Burst size of multiple transaction; should be programmed same as DMA controller multiple-transaction-size SRC/DEST_MSIZE. 0: 1 transfers 1: 4 2: 8 3: 16 4: 32 5: 64 6: 128 7: 256 The unit for transfer is the H_DATA_WIDTH parameter. A single transfer (dw_dma_single assertion in case of Non DW DMA interface) would be signalled based on this value. Value should be sub-multiple of (RX_WMark + 1)* (F_DATA_WIDTH/H_DATA_WIDTH) and (FIFO_DEPTH - TX_WMark)* (F_DATA_WIDTH/ H_DATA_WIDTH) For example, if FIFO_DEPTH = 16, FDATA_WIDTH == H_DATA_WIDTH Allowed combinations for MSize and TX_WMark are: MSize = 1, TX_WMARK = 1-15 MSize = 4, TX_WMark = 8 MSize = 4, TX_WMark = 4 MSize = 4, TX_WMark = 12 MSize = 8, TX_WMark = 8 MSize = 8, TX_WMark = 4 Allowed combinations for MSize and RX_WMark are: MSize = 1, RX_WMARK = 0-14 MSize = 4, RX_WMark = 3 MSize = 4, RX_WMark = 7 MSize = 4, RX_WMark = 11 MSize = 8, RX_WMark = 7 Recommended: MSize = 8, TX_WMark = 8, RX_WMark = 7

Bit	Attr	Reset Value	Description
27:16	RW	0x000	rx_wmark FIFO threshold watermark level when receiving data to card. When FIFO data count reaches greater than this number, DMA/FIFO request is raised. During end of packet, request is generated regardless of threshold programming in order to complete any remaining data. In non-DMA mode, when receiver FIFO threshold (RXDR) interrupt is enabled, then interrupt is generated instead of DMA request. During end of packet, interrupt is not generated if threshold programming is larger than any remaining data. It is responsibility of host to read remaining bytes on seeing Data Transfer Done interrupt. In DMA mode, at end of packet, even if remaining bytes are less than threshold, DMA request does single transfers to flush out any remaining bytes before Data Transfer Done interrupt is set. 12 bits-1 bit less than FIFO-count of status register, which is 13 bits. Limitation: RX_WMark <= FIFO_DEPTH-2 Recommended: (FIFO_DEPTH/2) - 1; (means greater than (FIFO_DEPTH/2) - 1) *NOTE: In DMA mode during CCS time-out, the DMA does not generate the request at the end of packet, even if remaining bytes are less than threshold. In this case, there will be some data left in the FIFO. It is the responsibility of the application to reset the FIFO after the CCS timeout.*
15:12	RO	0x0	reserved
11:0	RW	0x000	tx_wmark FIFO threshold watermark level when transmitting data to card. When FIFO data count is less than or equal to this number, DMA/FIFO request is raised. If Interrupt is enabled, then interrupt occurs. During end of packet, request or interrupt is generated, regardless of threshold programming. In non-DMA mode, when transmit FIFO threshold (TXDR) interrupt is enabled, then interrupt is generated instead of DMA request. During end of packet, on last interrupt, host is responsible for filling FIFO with only required remaining bytes (not before FIFO is full or after CIU completes data transfers, because FIFO may not be empty). In DMA mode, at end of packet, if last transfer is less than burst size, DMA controller does single cycles until required bytes are transferred. 12 bits -1 bit less than FIFO-count of status register, which is 13 bits. Limitation: TX_WMark >= 1; Recommended: FIFO_DEPTH/2; (means less than or equal to FIFO_DEPTH/2)

SDMMC_CDETECT
Address: Operational Base + offset (0x0050)
Card-detect register

Bit	Attr	Reset Value	Description
31:1	RO	0x0	reserved
0	RO	0x0	card_detect_n Value on card_detect_n input ports; read-only bits. 0 represents presence of card.

SDMMC_WRTPRT
Address: Operational Base + offset (0x0054)
Write-protect register

Bit	Attr	Reset Value	Description
31:1	RO	0x0	reserved
0	RW	0x0	write_protect Value on card_write_prt input port. 1 represents write protection.

SDMMC_TCBCNT
Address: Operational Base + offset (0x005c)
Transferred CIU card byte count

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	trans_card_byte_count Number of bytes transferred by CIU unit to card. In 32-bit or 64-bit AMBA data-bus-width modes, register should be accessed in full to avoid read-coherency problems. In 16-bit AMBA data-bus-width mode, internal 16-bit coherency register is implemented. User should first read lower 16 bits and then higher 16 bits. When reading lower 16 bits, higher 16 bits of counter are stored in temporary register. When higher 16 bits are read, data from temporary register is supplied. Both SDMMC_TCBCNT and SDMMC_TBBCNT share same coherency register. When AREA_OPTIMIZED parameter is 1, register should be read only after data transfer completes; during data transfer, register returns 0.

SDMMC_TBBCNT
Address: Operational Base + offset (0x0060)
Transferred host/DMA to/from BIU-FIFO byte count

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	trans_fifo_byte_count Number of bytes transferred between Host/DMA memory and BIU FIFO. In 32-bit or 64-bit AMBA data-bus-width modes, register should be accessed in full to avoid read-coherency problems. In 16-bit AMBA data-bus-width mode, internal 16-bit coherency register is implemented. User should first read lower 16 bits and then higher 16 bits. When reading lower 16 bits, higher 16 bits of counter are stored in temporary register. When higher 16 bits are read, data from temporary register is supplied. Both SDMMC_TCBCNT and SDMMC_TBBCNT share same coherency register.

SDMMC_DEBNCE
Address: Operational Base + offset (0x0064)
Card detect debounce register

Bit	Attr	Reset Value	Description
31:24	RO	0x0	reserved
23:0	RW	0xffffff	debounce_count Number of host clocks (clk) used by debounce filter logic; typical debounce time is 5-25 ms.

SDMMC_USRID
Address: Operational Base + offset (0x0068)
User ID register

Bit	Attr	Reset Value	Description
31:0	RW	0x07967797	usrid User identification register. The default value is determined by Configuration Value.

SDMMC_VERID
Address: Operational Base + offset (0x006c)
Version ID register

Bit	Attr	Reset Value	Description
31:0	RO	0x5342270a	verid Version identification register; register value is hard-wired. Can be read by firmware to support different versions of core.

SDMMC_HCON
Address: Operational Base + offset (0x0070)
Hardware configuration register

Bit	Attr	Reset Value	Description

31:0	RO	0x00000000	HCON Configuration Dependent. Hardware configurations selected by user before synthesizing core. Register values can be used to develop configuration-independent software drivers. [0]: CARD_TYPE 0: MMC_ONLY 1: SD_MMC [5:1]: NUM_CARDS - 1 [6]: H_BUS_TYPE 0: APB 1: AHB [9:7]: H_DATA_WIDTH 0: 16 bits 1: 32 bits 2: 64 bits others: reserved [15:10]: H_ADDR_WIDTH 0 to 7: reserved 8: 9 bits 9: 10 bits … 31: 32 bits 32 to 63: reserved [17:16]: DMA_INTERFACE 0: none 1: DMA 1 2: DMA 2 3: DMA 3 [20:18]: GE_DMA_DATA_WIDTH 0: 16 bits 1: 32 bits 2: 64 bits others: reserved [21]: FIFO_RAM_INSIDE 0: outside 1: inside [22]: IMPLEMENT_HOLD_REG 0: no hold register 1: hold register [23]: SET_CLK_FALSE_PATH 0: no false path 1: false path set [25:24]: NUM_CLK_DIVIDER-1 [26]: AREA_OPTIMIZED 0: no area optimization 1: Area optimization

SDMMC_UHS_REG
Address: Operational Base + offset (0x0074)
UHS-1 register

Bit	Attr	Reset Value	Description
31:17	RO	0x0	reserved
16	RW	0x0	ddr_reg DDR mode. Determines the voltage fed to the buffers by an external voltage regulator. 0: non-DDR mode 1: DDR mode SDMMC_UHS_REG[16] should be set for card.
15:0	RO	0x0	reserved

SDMMC_RST_n
Address: Operational Base + offset (0x0078)
Hardware reset register

Bit	Attr	Reset Value	Description
31:1	RO	0x0	reserved
0	RW	0x1	card_reset Hardware reset. 0: active mode 1: reset These bits cause the cards to enter pre-idle state, which requires them to be re-initialized. CARD_RESET[0] should be set to 1'b1 to reset card.

SDMMC_PLDMND
Address: Operational Base + offset (0x0084)
Poll demand register

Bit	Attr	Reset Value	Description
31:0	WO	0x00000000	PD Poll Demand. If the OWN bit of a descriptor is not set, the FSM goes to the Suspend state. The host needs to write any value into this register for the IDMAC FSM to resume normal descriptor fetch operation. This is a write only register.

SDMMC_CARDTHRCTL
Address: Operational Base + offset (0x0100)
Card read threshold enable register

Bit	Attr	Reset Value	Description
31:28	RO	0x0	reserved
27:16	RW	0x000	CardRdThreshold Card Read Threshold size
15:2	RO	0x0	reserved

Bit	Attr	Reset Value	Description
1	RW	0x0	BsyClrIntEn Busy Clear Interrupt generation: 0: Busy Clear Interrupt disabled 1: Busy Clear Interrupt enabled Note: The application can disable this feature if it does not want to wait for a Busy Clear Interrupt. For example, in a multi-card scenario, the application can switch to the other card without waiting for a busy to be completed. In such cases, the application can use the polling method to determine the status of busy. By default this feature is disabled and backward-compatible to the legacy drivers where polling is used.
0	RW	0x0	CardRdThrEn Card Read Threshold Enable. 0: Card Read Threshold disabled 1: Card Read Threshold enabled. Host Controller initiates Read Transfer only if CardRdThreshold amount of space is available in receive FIFO.

SDMMC_BACK_END_POWER
Address: Operational Base + offset (0x0104)
Back-end power register

Bit	Attr	Reset Value	Description
31:1	RO	0x0	reserved
0	RW	0x0	back_end_power Back end power 0: Off 1: Back-end Power supplied to card application

SDMMC_EMMC_DDR_REG
Address: Operational Base + offset (0x010c)
eMMC4.5 DDR start bit detection control register

Bit	Attr	Reset Value	Description
31:1	RO	0x0	reserved
0	RW	0x0	HALF_START_BIT Control for start bit detection mechanism inside Mobile Storage Host Controller based on duration of start bit; each bit refers to one slot. For eMMC 4.5, start bit can be: 0: Full cycle (HALF_START_BIT = 0) 1: Less than one full cycle (HALF_START_BIT = 1) Set HALF_START_BIT=1 for eMMC 4.5 and above; set to 0 for SD applications.

SDMMC_FIFO_BASE
Address: Operational Base + offset (0x0200)
FIFO base address register

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	fifo_base_addr FIFO base address

10.5 Interface Description

The interface and IOMUX setting for SDMMC, SDIO, EMMC are shown as follows.

Table 10-4 SDMMC Interface Description

Module Pin	Direction	Pad Name	IOMUX Setting
sdmmc_cclk	O	IO_MMC0clkout_GPIO1c0	GRF_GPIO1C_IOMUX[1:0] =2'b01
sdmmc_ccmd	I/O	IO_MMC0cmd_GPIO1b7	GRF_GPIO1B_IOMUX[15:14] =2'b01
sdmmc_cdata0	I/O	IO_MMC0d0_UART2tx_GPIO1c2	GRF_GPIO1C_IOMUX[5:4] =2'b01
sdmmc_cdata1	I/O	IO_MMC0d1_UART2rx_GPIO1c3	GRF_GPIO1C_IOMUX[7:6] =2'b01
sdmmc_cdata2	I/O	IO_MMC0d2_JTAGtck1_GPIO1c4	GRF_GPIO1C_IOMUX[9:8] =2'b01
sdmmc_cdata3	I/O	IO_MMC0d3_JTAGtms1_GPIO1c5	GRF_GPIO1C_IOMUX[11:10] =2'b01
sdmmc_cdetectn	I	IO_MMC0detn_GPIO1c1	GRF_GPIO1C_IOMUX[3:2] =2'b01
sdmmc_wprt	I	IO_MMC0wrprt_GPIO1a7	GRF_GPIO1A_IOMUX[15:14] =2'b01
sdmmc_pwren	O	IO_MMC0pwren_GPIO1b6	GRF_GPIO1B_IOMUX[13:12] =2'b01

Notes: I=input, O=output, I/O=input/output, bidirectional

Table 10-5 SDIO Interface Description

Module Pin	Direction	Pad Name	IOMUX Setting
sdio_cclk	O	IO_I2Smclk_MMC1clkout_XIN32k_GPIO1a0	GRF_GPIO1A_IOMUX[1:0] =2'b10
sdio_ccmd	I/O	IO_I2C1tpsda_MMC1cmd_GPIO0a3	GRF_GPIO0A_IOMUX[7:6] =2'b10
sdio_cdata0	I/O	IO_I2Ssclk_MMC1d0_PMICsleep1_GPIO1a1	GRF_GPIO1A_IOMUX[3:2] =2'b10
sdio_cdata1	I/O	IO_I2Slrckrx_MMC1d1_GPIO1a2	GRF_GPIO1A_IOMUX[5:4] =2'b10
sdio_cdata2	I/O	IO_I2Ssdo_MMC1d2_GPIO1a4	GRF_GPIO1A_IOMUX[9:8] =2'b10
sdio_cdata3	I/O	IO_I2Ssdi_MMC1d3_GPIO1a5	GRF_GPIO1A_IOMUX[11:10] =2'b10
sdio_pwren	O	IO_MMC1pwren_GPIO0d6	GRF_GPIO0D_IOMUX[13:12] =2'b01

Notes: I=input, O=output, I/O=input/output, bidirectional

Table 10-6 EMMC Interface Description

Module Pin	Direction	Pad Name	IOMUX Setting
emmc_cclk	O	IO_NANDdqs_EMMCclkout_GPIO2a7	GRF_GPIO2A_IOMUX[15:14] =2'b10
emmc_ccmd	I/O	IO_NANDcs2_EMMCcmd_GPIO1c6	GRF_GPIO1C_IOMUX[13:12] =2'b10 & GRF_SOC_CON1[6]=1'b0
		IO_NANDrdy_EMMCcmd1_SFCclk_GPIO2a4	GRF_GPIO2A_IOMUX[9:8] =2'b10 & GRF_SOC_CON1[6]=1'b1
emmc_cdata0	I/O	IO_NANDd0_EMMCd0_SFCd0_GPIO1d0	GRF_GPIO1D_IOMUX[1:0] =2'b10
emmc_cdata1	I/O	IO_NANDd1_EMMCd1_SFCd1_GPIO1d1	GRF_GPIO1D_IOMUX[3:2] =2'b10
emmc_cdata2	I/O	IO_NANDd2_EMMCd2_SFCd2_GPIO1d2	GRF_GPIO1D_IOMUX[5:4] =2'b10
emmc_cdata3	I/O	IO_NANDd3_EMMCd3_SFCd3_GPIO1d3	GRF_GPIO1D_IOMUX[7:6] =2'b10
emmc_cdata4	I/O	IO_NANDd4_EMMCd4_SPI1rxd1_GPIO1d4	GRF_GPIO1D_IOMUX[9:8] =2'b10
emmc_cdata5	I/O	IO_NANDd5_EMMCd5_SPI1txd1_GPIO1d5	GRF_GPIO1D_IOMUX[11:10] =2'b10
emmc_cdata6	I/O	IO_NANDd6_EMMCd6_SPI1csn0_GPIO1d6	GRF_GPIO1D_IOMUX[13:12] =2'b10
emmc_cdata7	I/O	IO_NANDd7_EMMCd7_SPI1csn1_GPIO1d7	GRF_GPIO1D_IOMUX[15:14] =2'b10
emmc_rstn	O	IO_NANDcs3_EMMCrstnout_GPIO1c7	GRF_GPIO1C_IOMUX[15:14] =2'b10
emmc_pwren	O	IO_NANDwp_EMMCpwren_GPIO2a5	GRF_GPIO2A_IOMUX[11:10] =2'b10

Notes: I=input, O=output, I/O=input/output, bidirectional

10.6 Application Notes

10.6.1 Card-Detect and Write-Protect Mechanism

Following figure illustrates how the SD/MMC card detection and write-protect signals are connected. Most of the SD/MMC sockets have card-detect pins. When no card is present, card_detect_n is 1 due to the pull-up. When the card is inserted, the card-detect pin is shorted to ground, which makes card_detect_n go to 0. Similarly in SD cards, when the write-protect switch is toward the left, it shorts the write_protect port to ground.

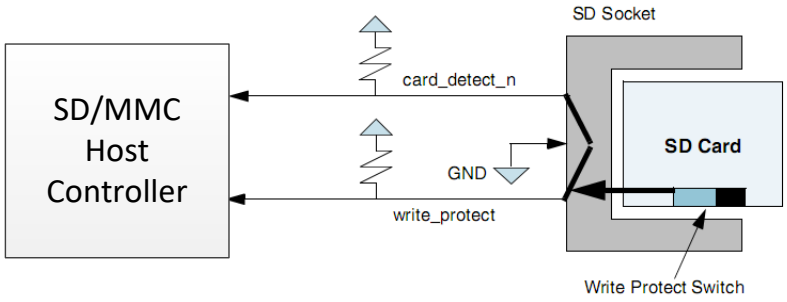


Fig. 10-6 SD/MMC Card-Detect and Write-Protect

10.6.2 SD/MMC Termination Requirement

Following Figure illustrates the SD/MMC termination requirements, which is required to pull up ccmd and cdata lines on the device bus. The recommended specification for pull-up on the ccmd line (Rcmd) is 4.7K - 100K for MMC, and 10K - 100K for an SD. The recommended pull-up on the cdata line (Rdat) is 50K - 100K.

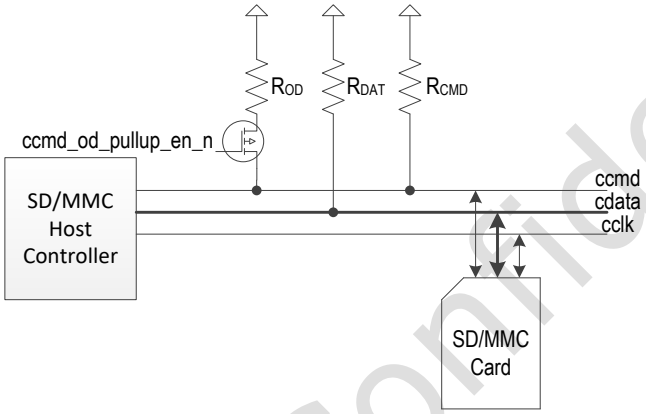


Fig. 10-7 SD/MMC Card Termination

1. Rcmd and Rod Calculation

The SD/MMC card enumeration happens at a very low frequency – 100-400KHz. Since the MMC bus is a shared bus between multiple cards, during enumeration open-drive mode is used to avoid bus conflict. Cards that drive 0 win over cards that drive "z". The pull-up in the command line pulls the bus to 1 when all cards drive "z". During normal data transfer, the host chooses only one card and the card driver switches to push-pull mode.

For example, if enumeration is done at 400KHz and the total bus capacitance is 200 pf, the pull-up needed during enumeration is:

2.2 RC = rise-time = 1/400KHz

R = 1/(2.2 * C * 100KHz)

= 1/(2.2 x 200 x 10**-12 x 400 x 10**3)

= 1/(17.6 x 10**-5)

= 5.68K

The ROD and RCMD should be adjusted in such a way that the effective pull-up is at the maximum 5.68K during enumeration. If there are only a few cards in the bus, a fixed RCMD resister is sufficient and there is no need for an additional ROD pull-up during enumeration. You should also ensure the effective pull-up will not violate the Iol rating of the drivers.

In SD mode, since each card has a separate bus, the capacitance is less, typically in the order of 20-30pf (host capacitance + card capacitance + trace + socket capacitance). For example, if enumeration is done at 400KHz and the total bus capacitance is 20pf, the pull-up needed during enumeration is:

2.2 RC = rise-time = 1/400KHz

R = 1/(2.2 * C * 100KHz)

= 1/(2.2 x 20 x 10**-12 x 400 x 10**3)

= 1/(1.76 x 10**-5)

= 56.8K

Therefore, a fixed 56.8K permanent Rcmd is sufficient in SD mode to enumerate the cards.

The driver of the SD/MMC on the "command" port needs to be only a push-pull driver. During enumeration, the SD/MMC emulates an open-drain driver by driving only a 0 or a"z" by controlling the ccmd_out and ccmd_out_en signals.

10.6.3 Software/Hardware Restriction

Before issuing a new data transfer command, the software should ensure that the card is not busy due to any previous data transfer command. Before changing the card clock frequency, the software must ensure that there are no data or command transfers in progress.

If the card is enumerated in SDR50, or DDR50 mode, then the application must program the use_hold_reg bit[29] in the CMD register to 1'b0 (phase shift of cclk_in_drv = 0) or 1'b1 (phase shift of cclk_in_drv>0). If the card is enumerated in SDR12 or SDR25 mode, the application must program the use_hold_reg bit[29] in the SDMMC_CMD register to 1'b1.

This programming should be done for all data transfer commands and non-data commands that are sent to the card. When the use_hold_reg bit is programmed to 1'b0, the Host Controller bypasses the Hold Registers in the transmit path. The value of this bit should not be changed when a Command or Data Transfer is in progress. For more details on using use_hold_reg and the implementation requirements for meeting the Card input hold time, refer to "Recommended Usage" in following table.

Table 10-7 Recommended Usage of use_hold_reg

No.	Speed Mode	use_hold_reg	cclk_in (MHz)	clk_in_drv (MHz)	clk_divider	Phase shift
1	SDR104	1'b0	200	200	0	0
2	SDR104	1'b1	200	200	0	Tunable> 0
3	SDR50	1'b0	100	100	0	0
4	SDR50	1'b1	100	100	0	Tunable> 0
5	DDR50 (8bit)	1'b0	100	100	1	0
6	DDR50 (8bit)	1'b1	100	100	1	Tunable> 0
7	DDR50 (4bit)	1'b0	50	50	0	0
8	DDR50 (4bit)	1'b1	50	50	0	Tunable> 0
9	SDR25	1'b1	50	50	0	Tunable> 0
10	SDR12	1'b1	50	50	1	Tunable> 0

To avoid glitches in the card clock outputs, the software should use the following steps when changing the card clock frequency:

1) Before disable the clocks, ensure that the card is not busy due to any previous data command. To determine this, check for 0 in bit9 of SDMMC_STATUS register.

2) Update the Clock Enable register to disable all clocks. To ensure completion of any previous command before this update, send a command to the CIU to update the clock registers by setting:

- start_cmd bit
- "update clock registers only" bits
- "wait_previous data complete" bit

Wait for the CIU to take the command by polling for 0 on the start_cmd bit.

3) Set the start_cmd bit to update the Clock Divider and/or Clock Source registers, and send a command to the CIU in order to update the clock registers; wait for the CIU to take the command.

4) Set start_cmd to update the Clock Enable register in order to enable the required clocks and send a command to the CIU to update the clock registers; wait for the CIU to take the command.

In non-DMA mode, while reading from a card, the Data Transfer Over (SDMMC_RINTSTS[3]) interrupt occurs as soon as the data transfer from the card is over. There still could be some data left in the FIFO, and the RX_WMark interrupt may or may not occur, depending on the remaining bytes in the FIFO. Software should read any

remaining bytes upon seeing the Data Transfer Over (DTO) interrupt. While using the external DMA interface for reading from a card, the DTO interrupt occurs only after all the data is flushed to memory by the DMA interface unit.
While writing to a card in external DMA mode, if an undefined-length transfer is selected by setting the Byte Count Register to 0, the DMA logic will likely request more data than it will send to the card, since it has no way of knowing at which point the software will stop the transfer. The DMA request stops as soon as the DTO is set by the CIU.
If the software issues a controller_reset command by setting control register bit[0] to 1, all the CIU state machines are reset; the FIFO is not cleared. The DMA sends all remaining bytes to the host. In addition to a card-reset, if a FIFO reset is also issued, then:

- Any pending DMA transfer on the bus completes correctly
- DMA data read is ignored
- Write data is unknown(x)

Additionally, if dma_reset is also issued, any pending DMA transfer is abruptly terminated. When the DMA is used, the DMA controller channel should also be reset and reprogrammed.
If any of the previous data commands do not properly terminate, then the software should issue the FIFO reset in order to remove any residual data, if any, in the FIFO. After asserting the FIFO reset, you should wait until this bit is cleared.
One data-transfer requirement between the FIFO and host is that the number of transfers should be a multiple of the FIFO data width (32bits). For example, you want to write only 15 bytes to an SD/MMC card (SDMMC_BYTCNT), the host should write 16 bytes to the FIFO or program the DMA to do 16-byte transfers. The software can still program the Byte Count register to only 15, at which point only 15 bytes will be transferred to the card. Similarly, when 15 bytes are read from a card, the host should still read all 16 bytes from the FIFO. It is recommended that you not change the FIFO threshold register in the middle of data transfers.

10.6.4 Programming Sequence

1. Initialization

Following figure illustrates the initialization flow.

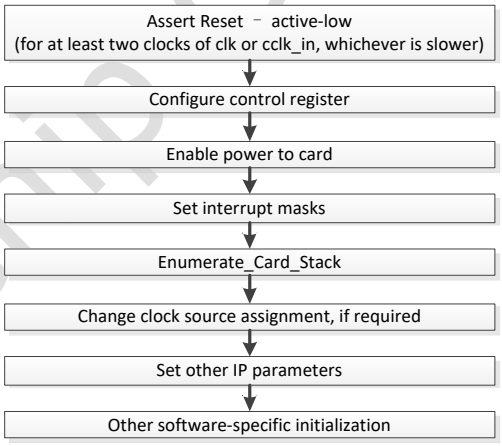


Fig. 10-8 Host Controller Initialization Sequence

Once the power and clocks are stable, reset_n should be asserted(active-low) for at least two clocks of clk or cclk_in, whichever is slower. The reset initializes the registers, ports, FIFO-pointers, DMA interface controls, and state-machines in the design. After power-on reset, the software should do the following:

1) Configure control register – For MMC mode, enable the open-drain pullup by setting enable_OD_pullup(bit24) in the SDMMC_CTRL register.
2) Enable power to cards – Before enabling the power, confirm that the voltage setting to the voltage regulators is correct. Enable power to the connected cards by setting the corresponding bit to 1 in the Power Enable register. Wait for the power ramp-up time.
3) Set masks for interrupts by clearing appropriate bits in the Interrupt Mask register. Set the global int_enable bit of the Control register. It is recommended that you write 0xffff_ffff to the Raw Interrupt register in order to clear any pending interrupts before setting the int_enable bit.

4) Enumerate card stack – Each card is enumerated according to card type; for details, refer to "Enumerated Card Stack". For enumeration, you should restrict the clock frequency to 400KHz.
5) Changing clock source assignment – set the card frequency using the clock-divider and clock-source registers; for details, refer to "Clock Programming". MMC cards operate at a maximum of 20MHz (at maximum of 52MHz in high-speed mode). SD mode operates at a maximum of 25MHz (at maximum of 50MHz in high-speed mode).
6) Set other parameters, which normally do not need to be changed with every command, with a typical value such as timeout values in cclk_out according to SD/MMC specifications.

- ResponseTimeOut = 0x64
- DataTimeOut = highest of one of the following:
- (10*((TAAC*Fop)+(100*NSAC))
- Host FIFO read/write latency from FIFO empty/full
- Set the debounce value to 25ms (default: 0x0fffff) in host clock cycle units in the SDMMC_DEBNCE register.
- FIFO threshold value in bytes in the SDMMC_FIFOTH register.

2. Enumerated Card Stack

The card stack does the following:

- Enumerates all connected cards
- Sets the RCA for the connected cards
- Reads card-specific information
- Stores card-specific information locally

Enumeration depends on the operating mode of the SD/MMC card; the card type is first identified and the appropriate card enumeration routine is called.

1) Check if the card is connected.
2) Clear the card type register to set the card width as a single bit. For the given card number, clear the corresponding bits in the card_type register. Clear the register bit for a 1-bit, 4-bit bus width. For example, for card number=1, clear bit 0 and bit 16 of the card_type register.
3) Set clock frequency to FOD=400KHz, maximum – Program clock divider0 (bits 0-7 in the SDMMC_CLKDIV register) value to one-half of the cclk_in frequency divided by 400 KHz. For example, if cclk_in is 20MHz, then the value is 20, 000/(2*400)=25.
4) Identify the card type; that is, SD, MMC, or SDIO.
   a. Send CMD5 first. If a response is received, then the card is SDIO
   b. If not, send CMD8 with the following Argument
      Bit[31:12] = 20'h0 //reserved bits
      Bit[11:8] = 4'b0001 //VHS value
      Bit[7:0] = 8'b10101010 //Preferred Check Pattern by SD2.0
   c. If Response is received the card supports High Capacity SD2.0 then send ACMD41 with the following Argument
      Bit[31] = 1'b0; //Reserved bits
      Bit[30] = 1'b1; //High Capacity Status
      Bit[29:24] = 6'h0; //Reserved bits
      Bit[23:0] = Supported Voltage Range
   d. If Response is received for ACMD41 then the card is SD. Otherwise the card is MMC.
   e. If response is not received for initial CMD8 then card does not support High Capacity SD2.0, then issue CMD0 followed by ACMD41 with the following Argument
      Bit[31] = 1'b0; //Reserved bits
      Bit[30] = 1'b0; //High Capacity Status
      Bit[29:24] = 6'h0; //Reserved bits
      Bit[23:0] = Supported Voltage Range
5) Enumerate the card according to the card type.
6) Use a clock source with a frequency = Fod (that is, 400KHz) and use the following enumeration command sequence:

- SD card – Send CMD0, CMD8, ACMD41, CMD2, CMD3.
- MMC – Send CMD0, CMD1, CMD2, CMD3.

3. Power Control

You can implement power control using the following registers, along with external circuitry:

- Control register bits card_voltage_a and card_voltage_b – Status of these bits is reflected at the IO pins. The bits can be used to generate or control the supply voltage that the memory cards require.
- Power enable register – Control power to individual cards.

Programming these two register depends on the implemented external circuitry. While turning on or off the power enable, you should confirm that power supply settings are correct. Power to all cards usually should be disabled while switching off the power.

4. Clock Programming

The Host Controller supports one clock sources. The clock to an individual card can be enabled or disabled. Registers that support this are:

- SDMMC_CLKDIV – Programs individual clock source frequency. SDMMC_CLKDIV limited to 0 or 1 is recommended.
- SDMMC_CLKSRC – Assign clock source for each card.
- SDMMC_CLKENA – Enables or disables clock for individual card and enables low-power mode, which automatically stops the clock to a card when the card is idle for more than 8 clocks.

The Host Controller loads each of these registers only when the start_cmd bit and the update_clk_regs_only bit in the SDMMC_CMD register are set. When a command is successfully loaded, the Host Controller clears this bit, unless the Host Controller already has another command in the queue, at which point it gives an HLE(Hardware Locked Error). Software should look for the start_cmd and the Update_clk_regs_only bits, and should also set the wait_prvdata_complete bit to ensure that clock parameters do not change during data transfer. Note that even though start_cmd is set for updating clock registers, the Host Controller does not raise a command_done signal upon command completion.

The following shows how to program these registers:

1) Confirm that no card is engaged in any transaction; if there is a transaction, wait until it finishes.
2) Stop all clocks by writing 0 to the SDMMC_CLKENA register. Set the start_cmd, Update_clk_regs_only, and wait_prvdata_complete bits in the SDMMC_CMD register. Wait until start_cmd is cleared or an HLE is set; in case of an HLE, repeat the command.
3) Program the SDMMC_CLKDIV and SDMMC_CLKSRC registers, as required. Set the start_cmd, Update_clk_regs_only, and wait_prvdata_complete bits in the SDMMC_CMD register. Wait until start_cmd is cleared or an HLE is set; in case of an HLE, repeat the command.
4) Re-enable all clocks by programming the SDMMC_CLKENA register. Set the start_cmd, Update_clk_regs_only, and wait_prvdata_complete bits in the SDMMC_CMD register. Wait until start_cmd is cleared or an HLE is set; in case of an HLE, repeat the command.

5. No-Data Command With or Without Response Sequence

To send any non-data command, the software needs to program the SDMMC_CMD register @0x2C and the SDMMC_CMDARG register @0x28 with appropriate parameters. Using these two registers, the Host Controller forms the command and sends it to the command bus. The Host Controller reflects the errors in the command response through the error bits of the SDMMC_RINTSTS register.

When a response is received – either erroneous or valid – the Host Controller sets the command_done bit in the SDMMC_RINTSTS register. A short response is copied in Response Register0, while along response is copied to all four response registers @0x30, 0x34, 0x38, and 0x3C. The Response3 register bit 31 represents the MSB, and the Response0 register bit 0 represents the LSB of a long response.

For basic commands or non-data commands, follow these steps:

1) Program the Command register @0x28 with the appropriate command argument parameter.
2) Program the Command register @0x2C with the settings in following table.

Table 10-8 Command Settings for No-Data Command

Parameter	Value	Description
Default		

Parameter	Value	Description
start_cmd	1	-
use_hold_reg	1/0	Choose value based on speed mode being used;ref to "use_hold_reg" on SDMMC_CMD register
update_clk_regs_only	0	No clock parameters update command
data_expected	0	No data command
card number	0	Actual card number(one controller only connect one card, the num is No. 0)
cmd_index	command-index	-
send_initialization	0	Can be 1, but only for card reset commands, such as CMD0
stop_abort_cmd	0	Can be 1 for commands to stop data transfer, such as CMD12
response_length	0	Can be 1 for R2(long) response
response_expect	1	Can be 0 for commands with no response; for example, CMD0, CMD4, CMD15, and so on
User-selectable		
wait_prvdata_complete	1	Before sending command on command line, host should wait for completion of any data command in process, if any (recommended to always set this bit, unless the current command is to query status or stop data transfer when transfer is in progress)
check_response_crc	1	If host should crosscheck CRC of response received

1) Wait for command acceptance by host. The following happens when the command is loaded into the Host Controller:
   - Host Controller accepts the command for execution and clears the start_cmd bit in the SDMMC_CMD register, unless one command is in process, at which point the Host Controller can load and keep the second command in the buffer.
   - If the Host Controller is unable to load the command – that is, a command is already in progress, a second command is in the buffer, and a third command is attempted – then it generates an HLE (hardware-locked error).
2) Check if there is an HLE.
3) Wait for command execution to complete. After receiving either a response from a card or response timeout, the Host Controller sets the command_done bit in the SDMMC_RINTSTS register. Software can either poll for this bit or respond to a generated interrupt.
4) Check if response_timeout error, response_CRC error, or response error is set. This can be done either by responding to an interrupt raised by these errors or by polling bits 1, 6, and 8 from the SDMMC_RINTSTS register @0x44. If no response error is received, then the response is valid. If required, the software can copy the response from the response registers @0x30-0x3C.

Software should not modify clock parameters while a command is being executed.

6. Data Transfer Commands

Data transfer commands transfer data between the memory card and the Host Controller. To send a data command, the Host Controller needs a command argument, total data size, and block size. Software can receive or send data through the FIFO.
Before a data transfer command, software should confirm that the card is not busy and is in a transfer state, which can be done using the CMD13 and CMD7 commands, respectively. For the data transfer commands, it is important that the same bus width that is programmed in the card should be set in the card type register @0x18.
The Host Controller generates an interrupt for different conditions during data transfer, which are reflected in the SDMMC_RINTSTS register @0x44 as:

1) Data_Transfer_Over (bit 3) – When data transfer is over or terminated. If there is a response timeout error, then the Host Controller does not attempt any data transfer and the "Data Transfer Over" bit is never set.
2) Transmit_FIFO_Data_request (bit 4) – FIFO threshold for transmitting data was reached; software is expected to write data, if available, in FIFO.
3) Receive_FIFO_Data_request (bit 5) – FIFO threshold for receiving data was reached; software is expected to read data from FIFO.
4) Data starvation by Host timeout (bit 10) – FIFO is empty during transmission or is full during reception. Unless software writes data for empty condition or reads data for full condition, the Host Controller cannot continue with data transfer. The clock to the card has been stopped.
5) Data read timeout error (bit 9) – Card has not sent data within the timeout period.
6) Data CRC error (bit 7) – CRC error occurred during data reception.
7) Start bit error (bit 13) – Start bit was not received during data reception.
8) End bit error (bit 15) – End bit was not received during data reception or for a write operation; a CRC error is indicated by the card.

Conditions 6, 7, and 8 indicate that the received data may have errors. If there was a response timeout, then no data transfer occurred.

7. Single-Block or Multiple-Block Read

Steps involved in a single-block or multiple-block read are:

1) Write the data size in bytes in the SDMMC_BYTCNT register @0x20.
2) Write the block size in bytes in the SDMMC_BLKSIZ register @0x1C. The Host Controller expects data from the card in blocks of size SDMMC_BLKSIZ each.
3) Program the SDMMC_CMDARG register @0x28 with the data address of the beginning of a data read.
4) Program the Command register with the parameters listed in following table. For SD and MMC cards, use CMD17 for a single-block read and CMD18 for a multiple-block read. For SDIO cards, use CMD53 for both single-block and multiple-block transfers.

Table 10-9 Command Setting for Single or Multiple-Block Read

Parameter	Value	Description
Default		
start_cmd	1	-
use_hold_reg	1/0	Choose value based on speed mode being used;ref to "use_hold_reg" on SDMMC_CMD register
update_clk_regs_only	0	No clock parameters update command
card number	0	Actual card number(one controller only connect one card, the num is No.0)
send_initialization	0	Can be 1, but only for card reset commands, such as CMD0
stop_abort_cmd	0	Can be 1 for commands to stop data transfer, such as CMD12
send_auto_stop	0/1	-

Parameter	Value	Description
transfer_mode	0	Block transfer
read_write	0	Read from card
data_expected	1	Data command
response_length	0	Can be 1 for R2(long) response
response_expect	1	Can be 0 for commands with no response; for example, CMD0, CMD4, CMD15, and so on
User-selectable		
cmd_index	command-index	-
wait_prvdata_complete	1	0- Sends command immediately 1- Sends command after previous data transfer ends
check_response_crc	1	0- Host Controller should not check response CRC 1- Host Controller should check response CRC

After writing to the SDMMC_CMD register, the Host Controller starts executing the command; when the command is sent to the bus, the command_done interrupt is generated.

- Software should look for data error interrupts; that is, bits 7, 9, 13, and 15 of the SDMMC_RINTSTS register. If required, software can terminate the data transfer by sending a STOP command.
- Software should look for Receive_FIFO_Data_request and/or data starvation by host timeout conditions. In both cases, the software should read data from the FIFO and make space in the FIFO for receiving more data.
- When a Data_Transfer_Over interrupt is received, the software should read the remaining data from the FIFO.

8. Single-Block or Multiple-Block Write

Steps involved in a single-block or multiple-block write are:

1) Write the data size in bytes in the SDMMC_BYTCNT register @0x20.
2) Write the block size in bytes in the SDMMC_BLKSIZ register @0x1C; the Host Controller sends data in blocks of size SDMMC_BLKSIZ each.
3) Program SDMMC_CMDARG register @0x28 with the data address to which data should be written.
4) Write data in the FIFO; it is usually best to start filling data the full depth of the FIFO.
5) Program the Command register with the parameters listed in following table.

Table 10-10 Command Settings for Single or Multiple-Block Write

Parameter	Value	Description
Default		
start_cmd	1	-
use_hold_reg	1/0	Choose value based on speed mode being used;ref to "use_hold_reg" on SDMMC_CMD register
update_clk_regs_only	0	No clock parameters update command
card number	0	Actual card number(one controller only connect one card, the num is No. 0)
send_initialization	0	Can be 1, but only for card reset commands, such as CMD0

Parameter	Value	Description
stop_abort_cmd	0	Can be 1 for commands to stop data transfer, such as CMD12
send_auto_stop	0/1	-
transfer_mode	0	Block transfer
read_write	1	Write to card
data_expected	1	Data command
response_length	0	Can be 1 for R2(long) response
response_expect	1	Can be 0 for commands with no response; for example, CMD0, CMD4, CMD15, and so on
User-selectable		
cmd_index	command-index	-
wait_prvdata_complete	1	0- Sends command immediately 1- Sends command after previous data transfer ends
check_response_crc	1	0- Host Controller should not check response CRC 1- Host Controller should check response CRC

After writing to the SDMMC_CMD register, Host Controller starts executing a command; when the command is sent to the bus, a command_done interrupt is generated.

- Software should look for data error interrupts; that is, for bits 7, 9, and 15 of the SDMMC_RINTSTS register. If required, software can terminate the data transfer by sending the STOP command.
- Software should look for Transmit_FIFO_Data_Request and/or timeout conditions from data starvation by the host. In both cases, the software should write data into the FIFO.
- When a Data_Transfer_Over interrupt is received, the data command is over. For an open-ended block transfer, if the byte count is 0, the software must send the STOP command. If the byte count is not 0, then upon completion of a transfer of a given number of bytes, the Host Controller should send the STOP command, if necessary. Completion of the AUTO-STOP command is reflected by the Auto_command_done interrupt – bit 14 of the SDMMC_RINTSTS register. A response to AUTO_STOP is stored in SDMMC_RESP1 @0x34.

9. Stream Read

A stream read is like the block read mentioned in "Single-Block or Multiple-Block Read", except for the following bits in the Command register:

transfer_mode = 1; //Stream transfer
cmd_index = CMD20;

A stream transfer is allowed for only a single-bit bus width.

10. Stream Write

A stream write is exactly like the block write mentioned in "Single-Block or Multiple-Block Write", except for the following bits in the Command register:

transfer_mode = 1;//Stream transfer
cmd_index = CMD11;

In a stream transfer, if the byte count is 0, then the software must send the STOP command. If the byte count is not 0, then when a given number of bytes completes a transfer, the Host Controller sends the STOP command. Completion of this AUTO_STOP command is reflected by the Auto_command_done interrupt. A response to an AUTO_STOP is stored in the SDMMC_RESP1 register@0x34.

A stream transfer is allowed for only a single-bit bus width.

11. Packed Commands

In order to reduce overhead, read and write commands can be packed in groups of commands—either all read or all write—that transfer the data for all commands in the group in one transfer on the bus.

Packed commands can be of two types:

- Packed Write: CMD23 →CMD25
- Packed Read: CMD23 → CMD25 → CMD23 → CMD18

Packed commands are put in packets by the application software and are transparent to the core.

12. Sending Stop or Abort in Middle of Transfer

The STOP command can terminate a data transfer between a memory card and the Controller, while the ABORT command can terminate an I/O data transfer for only the SDIO_IOONLY and SDIO_COMBO cards.

- Send STOP command – Can be sent on the command line while a data transfer is in progress; this command can be sent at any time during a data transfer.

You can also use an additional setting for this command in order to set the Command register bits (5-0) to CMD12 and set bit 14 (stop_abort_cmd) to 1. If stop_abort_cmd is not set to 1, the Controller does not know that the user stopped a data transfer. Reset bit 13 of the Command register (wait_prvdata_complete) to 0 in order to make the Controller send the command at once, even though there is a data transfer in progress.

- Send ABORT command – Can be used with only an SDIO_IOONLY or SDIO_COMBO card. To abort the function that is transferring data, program the function number in ASx bits (CCCR register of card, address 0x06, bits (0-2) using CMD52.

13. Suspend or Resume Sequence

In an SDIO card, the data transfer between an I/O function and the Controller can be temporarily halted using the SUSPEND command; this may be required in order to perform a high-priority data transfer with another function. When desired, the data transfer can be resumed using the RESUME command.

The following functions can be implemented by programming the appropriate bits in the CCCR register (Function 0) of the SDIO card. To read from or write to the CCCR register, use the CMD52 command.

- SUSPEND data transfer – Non-data command

1) Check if the SDIO card supports the SUSPEND/RESUME protocol; this can be done through the SBS bit in the CCCR register @0x08 of the card.
2) Check if the data transfer for the required function number is in process; the function number that is currently active is reflected in bits 0-3 of the CCCR register @0x0D. Note that if the BS bit (address 0xc::bit 0) is 1, then only the function number given by the FSx bits is valid.
3) To suspend the transfer, set BR (bit 2) of the CCCR register @0x0C.
4) Poll for clear status of bits BR (bit 1) and BS (bit 0) of the CCCR @0x0C. The BS (Bus Status) bit is 1 when the currently-selected function is using the data bus; the BR (Bus Release) bit remains 1 until the bus release is complete. When the BR and BS bits are 0, the data transfer from the selected function has been suspended.

- RESUME data transfer – This is a data command

1) Check that the card is not in a transfer state, which confirms that the bus is free for data transfer.
2) If the card is in a disconnect state, select it using CMD7. The card status can be retrieved in response to CMD52/CMD53 commands.
3) Check that a function to be resumed is ready for data transfer; this can be confirmed by reading the RFx flag in CCCR @0x0F. If RF = 1, then the function is ready for data transfer.
4) To resume transfer, use CMD52 to write the function number at FSx bits (0-3) in the CCCR register @0x0D. Form the command argument for CMD52 and write it in SDMMC_CMDARG @0x28.
5) Write the block size in the SDMMC_BLKSIZ register @0x1C; data will be transferred in units of this block size.
6) Write the byte count in the SDMMC_BYTCNT register @0x20. This is the total size of the

data; that is, the remaining bytes to be transferred. It is the responsibility of the software to handle the data.
7) Program Command register; similar to a block transfer.
8) When the Command register is programmed, the command is sent and the function resumes data transfer. Read the DF flag (Resume Data Flag). If it is 1, then the function has data for the transfer and will begin a data transfer as soon as the function or memory is resumed. If it is 0, then the function has no data for the transfer.
9) If the DF flag is 0, then in case of a read, the Host Controller waits for data. After the data timeout period, it gives a data timeout error.

14. Read_Wait Sequence

Read_wait is used with only the SDIO card and can temporarily stall the data transfer—either from function or memory—and allow the host to send commands to any function within the SDIO device. The host can stall this transfer for as long as required. The Host Controller provides the facility to signal this stall transfer to the card. The steps for doing this are:

1) Check if the card supports the read_wait facility; read SRW (bit 2) of the CCCR register @0x08. If this bit is 1, then all functions in the card support the read_wait facility. Use CMD52 to read this bit.
2) If the card supports the read_wait signal, then assert it by setting the read_wait (bit 6) in the SDMMC_CTRL register @0x00.
3) Clear the read_wait bit in the SDMMC_CTRL register.

15. Controller/DMA/FIFO Reset Usage

- Controller reset – Resets the controller by setting the controller_reset bit (bit 0) in the SDMMC_CTRL register; this resets the CIU and state machines, and also resets the BIU-to-CIU interface. Since this reset bit is self-clearing, after issuing the reset, wait until this bit is cleared.
- FIFO reset - Resets the FIFO by setting the fifo_reset bit (bit 1) in the SDMMC_CTRL register; this resets the FIFO pointers and counters of the FIFO. Since this reset bit is self-clearing, after issuing the reset, wait until this bit is cleared.

In external DMA transfer mode, even when the FIFO pointers are reset, if there is a DMA transfer in progress, it could push or pop data to or from the FIFO; the DMA itself completes correctly. In order to clear the FIFO, the software should issue an additional FIFO reset and clear any FIFO underrun or overrun errors in the SDMMC_RAWINTS register caused by the DMA transfers after the FIFO was reset.

16. Card Read Threshold

When an application needs to perform a Single or Multiple Block Read command, the application must program the SDMMC_CARDTHRCTL register with the appropriate Card Read Threshold size (CardRdThreshold) and set the Card Read Threshold Enable (CardRdThrEnable) bit to 1'b1. This additional programming ensures that the Host controller sends a Read Command only if there is space equal to the CardRDThreshold available in the Rx FIFO. This in turn ensures that the card clock is not stopped in the middle a block of data being transmitted from the card. The Card Read Threshold can be set to the block size of the transfer, which guarantees that there is a minimum of one block size of space in the RxFIFO before the controller enables the card clock.The Card Read Threshold is required when the Round Trip Delay is greater than 0.5cclk_in period.

17. Error Handling

The Host Controller implements error checking; errors are reflected in the SDMMC_RAWINTS register@0x44 and can be communicated to the software through an interrupt, or the software can poll for these bits. Upon power-on, interrupts are disabled (int_enable in the SDMMC_CTRL register is 0), and all the interrupts are masked (bits 0-31 of the SDMMC_INTMASK register; default is 0).

Error handling:

- Response and data timeout errors – For response timeout, software can retry the command. For data timeout, the Host Controller has not received the data start bit – either for the first block or the intermediate block – within the timeout period, so software can either retry the whole data transfer again or retry from a specified block onwards. By reading the contents of the SDMMC_TCBCNT later, the software can decide

how many bytes remain to be copied.
- Response errors – Set when an error is received during response reception. In this case, the response that copied in the response registers is invalid. Software can retry the command.
- Data errors – Set when error in data reception are observed; for example, data CRC, start bit not found, end bit not found, and so on. These errors could be set for any block-first block, intermediate block, or last block. On receipt of an error, the software can issue a STOP or ABORT command and retry the command for either whole data or partial data.
- Hardware locked error – Set when the Host Controller cannot load a command issued by software. When software sets the start_cmd bit in the SDMMC_CMD register, the Host Controller tries to load the command. If the command buffer is already filled with a command, this error is raised. The software then has to reload the command.
- FIFO underrun/overrun error – If the FIFO is full and software tries to write data in the FIFO, then an overrun error is set. Conversely, if the FIFO is empty and the software tries to read data from the FIFO, an underrun error is set. Before reading or writing data in the FIFO, the software should read the fifo_empty or fifo_full bits in the SDMMC_STATUS register.
- Data starvation by host timeout – Raised when the Host Controller is waiting for software intervention to transfer the data to or from the FIFO, but the software does not transfer within the stipulated timeout period. Under this condition and when a read transfer is in process, the software should read data from the FIFO and create space for further data reception. When a transmit operation is in process, the software should fill data in the FIFO in order to start transferring data to the card.
- CRC Error on Command – If a CRC error is detected for a command, the CE-ATA device does not send a response, and a response timeout is expected from the Host Controller. The ATA layer is notified that an MMC transport layer error occurred.

Notes: During a multiple-block data transfer, if a negative CRC status is received from the device, the data path signals a data CRC error to the BIU by setting the data CRC error bit in the SDMMC_RINTSTS register. It then continues further data transmission until all the bytes are transmitted.

10.6.5 Voltage Switching

The Host Controller supports SD 3.0 Ultra High Speed (UHS-1) and is capable of voltage switching in SD-mode, which can be applied to SD High-Capacity (SDHC) and SD Extended Capacity (SDXC) cards. UHS-1 supports only 4-bit mode.

However, whether the IO voltage of 1.8v supported or not is depended on the SoC design.

SD 3.0 UHS-1 supports the following transfer speed modes for UHS-50 and/or UHS-104 cards:
- DS – default-speed up to 25MHz, 3.3V signaling
- HS – high-speed up to 50MHz, 3.3V signaling
- SDR12 – SDR up to SDR 25MHz, 1.8V signaling
- SDR25 – SDR up to 50MHz, 1.8V signaling
- SDR50 – SDR up to 100MHz, 1.8V signaling
- DDR50 – DDR up to 50MHz, 1.8V signaling

Voltage selection can be done in only SD mode. The first CMD0 selects the bus mode-either SD mode or SPI mode. The card must be in SD mode in order for 1.8V signaling mode to apply, during which time the card cannot be switched to SPI mode or 3.3V signaling without a power cycle.

If the System BIOS in an embedded system already knows that it is connected to an SD 3.0 card, then the driver programs the Controller to initiate ACMD41. The software knows from the response of ACMD41 whether or not the card supports voltage switching to 1.8V.
- If bit 32 of ACMD41 response is 1'b1: card supports voltage switching and next command-CMD11-invokes voltage switching sequence. After CMD11 is started, the software must program the IO voltage selection register based on the soc architecture.
- If bit 32 of ACMD41 response is 1'b0: card does not support voltage switching and CMD11 should not be started.

If the card and host controller accept voltage switching, then they support UHS-1 modes of data transfer. After the voltage switch to 1.8V, SDR12 is the default speed.

Since the UHS-1 can be used in only 4-bit mode, the software must start ACMD6 and

change the card data width to 4-bit mode; ACMD6 is driven in any of the UHS-1 speeds. If the host wants to select the DDR mode of data transfer, then the software must program the DDR_REG register in the CSR space with the appropriate card number.
To choose from any of the SDR or DDR modes, appropriate values should be programmed in the SDMMC_CLKDIV register.

1. Voltage Switch Operation

The Voltage Switch operation must be performed in SD mode only.

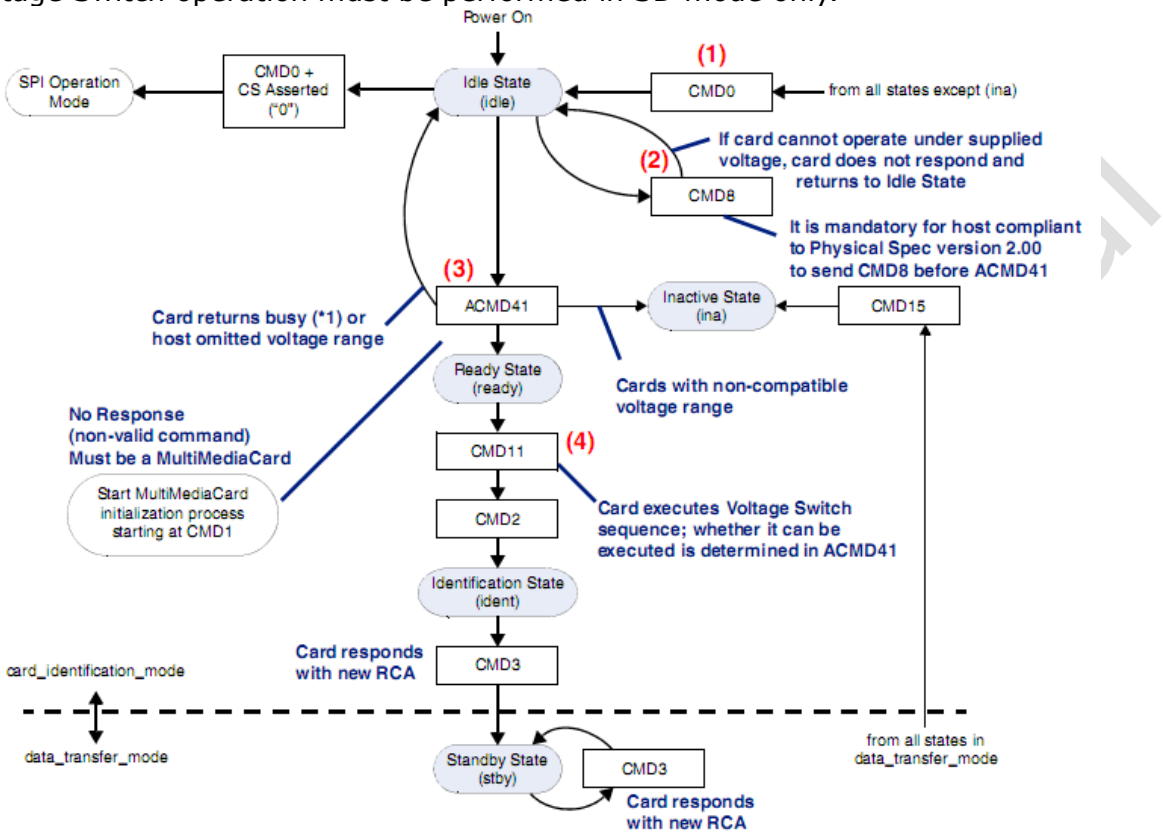


Fig. 10-9 Voltage Switching Command Flow Diagram

The following outlines the steps for the voltage switch programming sequence

1) Software Driver starts CMD0, which selects the bus mode as SD.
2) After the bus is in SD card mode, CMD8 is started in order to verify if the card is compatible with the SD Memory Card Specification, Version 2. 00. CMD8 determines if the card is capable of working within the host supply voltage specified in the VHS (19:16) field of the CMD; the card supports the current host voltage if a response to CMD8 is received.
3) ACMD 41 is started. The response to this command informs the software if the card supports voltage switching; bits 38, 36, and 32 are checked by the card argument of ACMD41; refer to following figure.

47	46	45-40	39	38	37	36	35-33	32	31-16	15-08	07-01	00
S	D	Index	Busy 31	HCS 30	(FB) 29	XPC 28	Reserved 27-25	S18R 24	OCR 23-08	Reserved 07-00	CRC7	E
0	1	101001	0	X	0	X	000	X	xxxxh	0000000	xxxxxxx	1

Host Capacity Support
0b: SDSC-only Host
1b: SDHC or SDXC supported

SCXC Power Control
0b: Power saving
1b: Maximum performance

S18R: Switching to 1.8V Request
0b: Use current signal voltage
1b: Switch to 1.8V signal voltage

Fig. 10-10 ACMD41 Argument

- Bit 30 informs the card if host supports SDHC/SDXC or not; this bit should be set to 1'b1.
- Bit 28 can be either 1 or 0.
- Bit 24 should be set to 1'b1, indicating that the host is capable of voltage switching; refer to following figure.

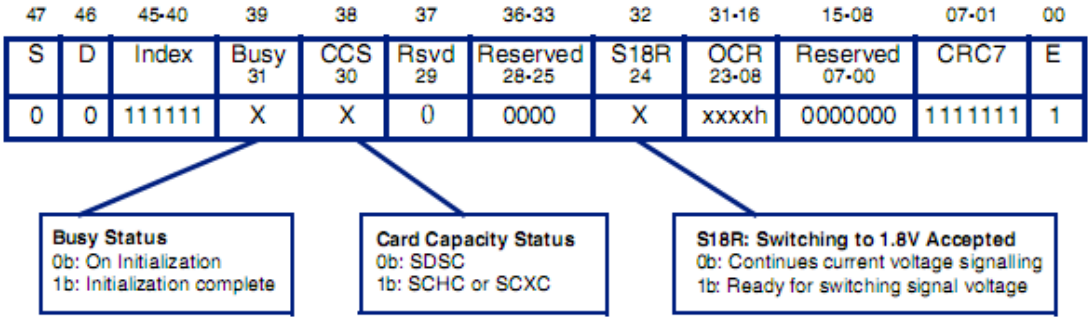


Fig. 10-11 ACMD41 Response(R3)

- Bit 30 – If set to 1'b1, card supports SDHC/SDXC; if set to 1'b0, card supports only SDSC
- Bit 24 – If set to 1'b1, card supports voltage switching and is ready for the switch
- Bit 31 – If set to 1'b1, initialization is over; if set to 1'b0, means initialization in process

4) If the card supports voltage switching, then the software must perform the steps discussed for either the "Voltage Switch Normal Scenario" or the "Voltage Switch Error Scenario".

2. Voltage Switch Normal Scenario

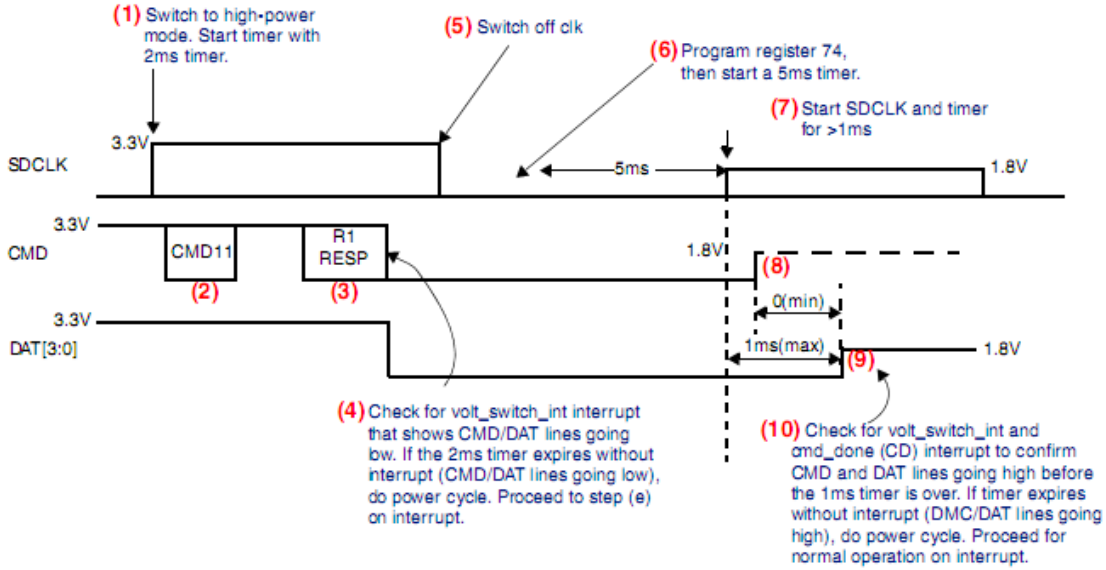


Fig. 10-12 Voltage Switch Normal Scenario

1) The host programs CLKENA—cclk_low_power register—with zero (0) for the corresponding card, which makes the host controller move to high-power mode. The application should start a timer with a recommended value of 2ms; this value of 2 ms is determined as below:Total clk required for CMD11 = 48 clks
   Total clk required for RESP R1 = 48 clks
   Maximum clk delay between MCD11 end to start of SDMMC_RESP1 = 60 clks
   Total = 48+48 + 60 = 160
   Minimum frequency during enumeration is 100 KHz; that is, 10us
   Total time = 160 * 10us = 1600us = 1. 6ms ~ 2ms
2) The host issues CMD11 to start the voltage switch sequence. Set bit 28 to 1'b1 in CMD when setting CMD11; for more information on setting bits, refer to "Boot Operation".
3) The card returns R1 response; the host controller does not generate cmd_done interrupt on receiving R1 response.
4) The card drives CMD and DAT [3:0] to low immediately after the response. The host controller generates interrupt (VOLT_SWITCH_INT) once the CMD or DAT [3:0] line goes low. The application should wait for this interrupt. If the 2ms timer expires without an interrupt (CMD/DAT lines going low), do a power cycle.

Note: Before doing a power cycle, switch off the card clock by programming SDMMC_CLKENA register

Proceed to step (5) on getting an interrupt (VOLT_SWITCH_INT).

Note: This interrupt must be cleared once this interrupt is received. Additionally, this interrupt should not be masked during the voltage switch sequence.

If the timer expires without interrupt (CMD/DAT lines going low), perform a power cycle. Proceed to step (5) on interrupt.

1) Program the SDMMC_CLKENA, cclk_enable register, with 0 for the corresponding card; the host stops supplying SDCLK.
2) Program Voltage register to the required values for the corresponding card. The application should start a timer > 5ms.
3) After the 5ms timer expires, the host voltage regulator is stable. Program SDMMC_CLKENA, cclk_enable register, with 1 for the corresponding card; the host starts providing SDCLK at 1. 8V; this can be at zero time after Voltage register has been programmed. When the SDMMC_CLKENA register is programmed, the application should start another timer > 1ms.
4) By detecting SDCLK, the card drives CMD to high at 1. 8V for at least one clock and then stops driving (tri-state); CMD is triggered by the rising edge of SDCLK (SDR timing).
5) If switching to 1. 8V signaling is completed successfully, the card drives DAT [3:0] to high at 1. 8V for at least one clock and then stops driving (tri-state); DAT [3:0] is triggered by the rising edge of SDCLK (SDR timing). DAT[3:0] must be high within 1ms from the start of SDCLK.
6) The host controller generates a voltage switch interrupt (VOLT_SWITCH_INT) and a command done (CD) interrupt once the CMD and DAT[3:0] lines go high. The application should wait for this interrupt to confirm CMD and DAT lines going high before the 1ms timer is done.

If the timer expires without the voltage switch interrupt (VOLT_SWITCH_INT), a power cycle should be performed. Program the SDMMC_CLKENA register to stop the clock for the corresponding card number. Wait for the cmd_done (CD) interrupt. Proceed for normal operation on interrupt. After the sequence is completed, the host and the card start communication in SDR12 timing.

3. Voltage Switch Error Scenario

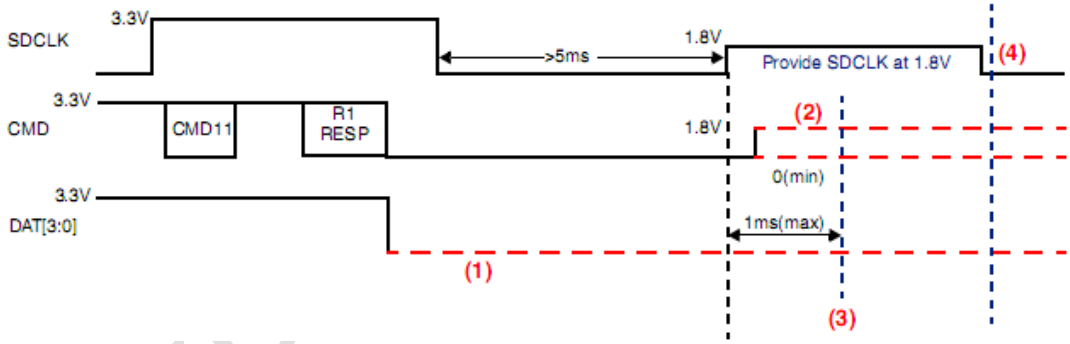


Fig. 10-13 Voltage Switch Error Scenario

1) If the interrupt (VOLT_SWITCH_INT) does not come, then the 2 ms timer should time out and a power cycle should be initiated.

Note: Before performing a power cycle, switch off the card clock by programming SDMMC_CLKENA register; no cmd_done (CD) interrupt is generated.

Additionally, if the card detects a voltage error at any point in between steps (5) and (7) in the card keeps driving DAT[3:0] to low until card power off.

2) CMD can be low or tri-state.
3) The host controller generates a voltage switch interrupt once the CMD and DAT[3:0] lines go high. The application should check for an interrupt to confirm CMD and DAT lines going high before the 1 ms timer is done.

If the 1 ms timer expires without interrupt (VOLT_SWITCH_INT) and cmd_done (CD), a power cycle should be performed. Program the SDMMC_CLKENA register to stop SDCLK of the corresponding card. Wait for the cmd_done interrupt. Proceed for normal operation on interrupt.

4) If DAT[3:0] is low, the host drives SDCLK to low and then stops supplying the card power.

Note: The card checks voltages of its own regulator output and host signals to ensure they are less than 2. 5V. Errors are indicated by (1) and (2).

- If voltage switching is accepted by the card, the default speed is SDR12.

- Command Done is given:
  - If voltage switching is properly done, CMD and DAT line goes high.
  - If switching is not complete, the 1ms timer expires, and the card clk is switched off.

Note: No other CMD should be driven before the voltage switching operation is completed and Command Done is received.

- The application should use CMD6 to check and select the particular function; the function appropriate-speed should be selected.

After the function switches, the application should program the correct value in the SDMMC_CLKDIV register, depending on the function chosen. Additionally, if Function 0x4 of the Access mode is chosen—that is, DDR50, then the application should also program 1'b1 in DDR_REG for the card number that has been selected for DDR50 mode.

10.6.6 Back-End Power

Each device needs one bit to control the back-end power supply for an embedded device; this bit does not control the VDDH of the host controller. A SDMMC_BACK_END_POWER register enables software programming for back-end power. The value on this register is output to the back_end_power signal, which can be used to switch power on and off the embedded device.

10.6.7 DDR Operation

1. 4-bit DDR Programming Sequence

DDR programming should be done only after the voltage switch operation has completed. The following outlines the steps for the DDR programming sequence:

1) Once the voltage switch operation is complete, the user must program voltage selection register to the required values for the corresponding card.

- To start a card to work in DDR mode, the application must program a bit of the newly defined UHS_REG[16] register with a value of 1'b1.
- The bit that the user programs depends on which card is to be accessed in DDR mode.

2) To move back to SDR mode, a power cycle should be run on the card—putting the card in SDR12 mode—and only then should SDMMC_UHS_REG[16] be set back to 1'b0 for the appropriate card.

2. 8-bit DDR Programming Sequence

The following outlines the steps for the 8-bit DDR programming sequence:

1) The cclk_in signal should be twice the speed of the required cclk_out. Thus, if the cclk_out signal is required to be 50 MHz, the cclk_in signal should be 100 MHz.
2) The SDMMC_CLKDIV register should always be programmed with a value higher than zero (0); that is, a clock divider should always be used for 8-bit DDR mode.
3) The application must program the SDMMC_UHS_REG[16] register (DDR_REG bits) by assigning it with a value of 1 for the bit corresponding to the card number; this causes the selected card to start working in DDR mode.
4) Depending on the card number, the SDMMC_CTYPE [31:16] bits should be set in order to make the host work in the 8-bit mode.

3. eMMC4.5 DDR START Bit

The eMMC4.5 changes the START bit definition in the following manner:

1) Receiver samples the START bit on the rising edge.
2) On the next rising edge after sampling the START bit, the receiver must sample the data.
3) Removes requirement of the START bit and END bit to be high for one full cycle.

Notes: The Host Controller does not support a START bit duration higher than one clock cycle. START bit durations of one or less than one clock cycle are supported and can be defined at the time of startup by programming the SDMMC_EMMC_DDR_REG register.

Following figure illustrates cases for the definition change of the START bit with eMMC4.5; it also illustrates how some of these cases can fail in sampling when higher-value delays are considered for I/O PADs.

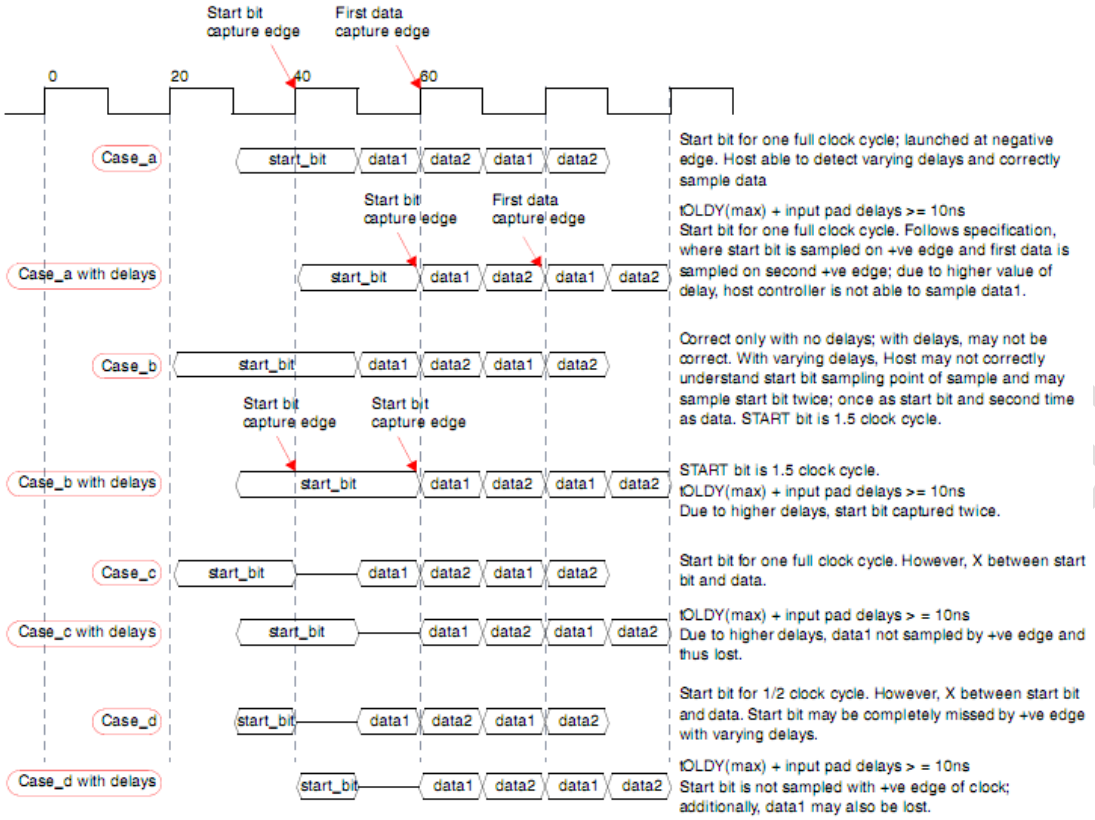


Fig. 10-14 CASES for eMMC 4.5 START bit

4. Reset Command/Moving from DDR50 to SDR12

To reset the mode of operation from DDR50 to SDR12, the following sequence of operations has to be done by the application:

1) Issue CMD0.

When CMD0 is received, the card changes from DDR50 to SDR12.

2) Program the SDMMC_CLKDIV register with an appropriate value.
3) Set DDR_REG to 0.

Note: The Voltage register should not be programmed to 0 while switching from DDR50 to SDR12, since the card is still operating in 1.8V mode after receiving CMD0.

10.6.8 H/W Reset Operation

When the RST_n signal goes low, the card enters a pre-idle state from any state other than the inactive state.

H/W Reset Programming Sequence

The following outlines the steps for the H/W reset programming sequence:

1) Program CMD12 to end any transfer in process.
2) Wait for DTO, even if no response is sent back by the card.
3) Set the following resets:

- DMA reset–SDMMC_CTRL[2]
- FIFO reset –SDMMC_CTRL[1] bits

Note: The above steps are required only if a transfer is in process.

4) Program the SDMMC_RST_n register with a value of 0; this can be done at any time when the card is connected to the controller. This programming asserts the RST_n signal and resets the card.
5) Wait for minimum of 1 µs or cclk_in period, whichever is greater
6) After a minimum of 1 µs, the application should program a value of 0 into the SDMMC_RST_n register. This de-asserts the RST_n signal and takes the card out of reset.
7) The application can program a new CMD only after a minimum of 200 µs after the de-assertion of the RST_n signal, as per the MMC 4.41 standard.

Note: For backward compatibility, the RST_n signal is temporarily disabled in the card by default. The host may need to set the signal as either permanently enabled or permanently disabled before it uses the card.

10.6.9 FBE Scenarios

An FBE occurs due to an AHB error response on the AHB bus. This is a system error, so the

software driver should not perform any further programming to the Host. The only recovery mechanism from such scenarios is to do one of the following:

- Issue a hard reset by asserting the reset_n signal
- Do a program controller reset by writing to the SDMMC_CTRL[0] register

1. FIFO Overflow and Underflow

During normal data transfer conditions, FIFO overflow and underflow will not occur. However if there is a programming error, then FIFO overflow/underflow can result. For example, consider the following scenarios.

- For transmit: PBL=4, Tx watermark = 1. For the above programming values, if the FIFO has only one location empty, it issues a dma_req to IDMAC FSM. Due to PBL value=4, the IDMAC FSM performs 4 pushes into the FIFO. This will result in a FIFO overflow interrupt.
- For receive: PBL=4, Rx watermark = 1. For the above programming values, if the FIFO has only one location filled, it issues a dma_req to IDMAC FSM. Due to PBL value=4, the IDMAC FSM performs 4 pops to the FIFO. This will result in a FIFO underflow interrupt.

The driver should ensure that the number of bytes to be transferred as indicated in the descriptor should be a multiple of 4bytes with respect to H_DATA_WIDTH=32. For example, if the SDMMC_BYTCNT = 13, the number of bytes indicated in the descriptor should be 16 for H_DATA_WIDTH=32.

2. Programming of PBL and Watermark Levels

The DMAC performs data transfers depending on the programmed PBL and threshold values.

Table 10-11 PBL and Watermark Levels

PBL (Number of transfers)	Tx/Rx Watermark Value
1	greater than or equal to 1
4	greater than or equal to 4
8	greater than or equal to 8
16	greater than or equal to 16
32	greater than or equal to 32
64	greater than or equal to 64
128	greater than or equal to 128
256	greater than or equal to 256

10.6.10 Variable Delay/Clock Generation

Variable delay mechanism for the cclk_in_drv is optional, but it can be useful in order to meet a range of hold-time requirements across modes. Variable delay mechanism for the cclk_in_sample is mandatory and is required to achieve the correct sampling point for data. cclk_in/cclk_in_sample/ cclk_in_drv is generated by Clock Generation Unit (CLKGEN) with variable delay mechanism, which includes Phase Shift Unit and Delay Line Unit selectable. The Phase Shift Unit can shift cclk_in_sample/cclk_in_drv by 0/90/180/270-degree relative to cclk_in, controlled by sample_degree/drv_degree.

The Delay Line Unit can shift cclk_in_sample/cclk_in_drv in the unit of 40ps~80ps for every delay element. The delay unit number is determined by sample_delaynum/drv_delaynum, and enabled by sample_sel/drv_sel.

cclk_in is generated by cclkin divided by 2. cclk_in_drv and cclk_in_sample clocks are phase-shifted with delayed versions of cclk_in. All clocks are recommended to have a 50% duty cycle; DDR modes must have 50% duty cycles.

The architecture is as follows.

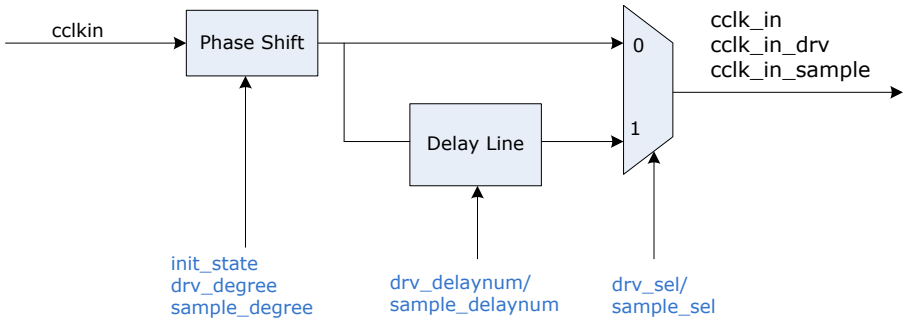


Fig. 10-15 Clock Generation Unit

The control signals for different Host Controller instance are shown as follows:

Table 10-12 Configuration for SDMMC Clock Generation

Signal Name	Source	Def	Description
init_state	CRU_SDMMC_CON0[0]	0	Soft initial state for phase shift.
drv_degree [1:0]	CRU_SDMMC_CON0[2:1]	2	Phase shift for cclk_in_drv. 0: 0-degree 1: 90-degree 2: 180-degree 3: 270-degree
drv_delaynum [7:0]	CRU_SDMMC_CON0[10:3]	0	Element number in delay line for cclk_in_drv
drv_sel	CRU_SDMMC_CON0[11]	0	cclk_in_drv source selection: 0: use clock after phase_shift 1: use clock after phase_shift and delay line
sample_degree [1:0]	CRU_SDMMC_CON1[1:0]	0	Phase shift for cclk_in_sample. 0: 0-degree 1: 90-degree 2: 180-degree 3: 270-degree
sample_delaynum[7:0]	CRU_SDMMC_CON1[9:2]	0	Element number in delay line for cclk_in_sample
sample_sel	CRU_SDMMC_CON1[10]	0	cclk_in_sample source selection: 0: use clock after phase_shift 1: use clock after phase_shift and delay line

Table 10-13 Configuration for SDIO Clock Generation

Signal Name	Source	Def	Description
init_state	CRU_SDIO0_CON0[0]	0	Soft initial state for phase shift.
drv_degree [1:0]	CRU_SDIO0_CON0[2:1]	2	Phase shift for cclk_in_drv. 0: 0-degree 1: 90-degree 2: 180-degree 3: 270-degree
drv_delaynum [7:0]	CRU_SDIO0_CON0[10:3]	0	Element number in delay line for cclk_in_drv
drv_sel	CRU_SDIO0_CON0[11]	0	cclk_in_drv source selection: 0: use clock after phase_shift 1: use clock after phase_shift and delay line
sample_degree [1:0]	CRU_SDIO0_CON1[1:0]	0	Phase shift for cclk_in_sample. 0: 0-degree 1: 90-degree 2: 180-degree 3: 270-degree
sample_delaynu	CRU_SDIO0_CON1[9:2]	0	Element number in delay line for

Signal Name	Source	Def	Description
m[7:0]			cclk_in_sample
sample_sel	CRU_SDIO0_CON1[10]	0	cclk_in_sample source selection: 0: use clock after phase_shift 1: use clock after phase_shift and delay line

Table 10-14 Configuration for EMMC Clock Generation

Signal Name	Source	Def	Description
init_state	CRU_EMMC_CON0[0]	0	Soft initial state for phase shift.
drv_degree [1:0]	CRU_EMMC_CON0[2:1]	2	Phase shift for cclk_in_drv. 0: 0-degree 1: 90-degree 2: 180-degree 3: 270-degree
drv_delaynum [7:0]	CRU_EMMC_CON0[10:3]	0	Element number in delay line for cclk_in_drv
drv_sel	CRU_EMMC_CON0[11]	0	cclk_in_drv source selection: 0: use clock after phase_shift 1: use clock after phase_shift and delay line
sample_degree [1:0]	CRU_EMMC_CON1[1:0]	0	Phase shift for cclk_in_sample. 0: 0-degree 1: 90-degree 2: 180-degree 3: 270-degree
sample_delaynum[7:0]	CRU_EMMC_CON1[9:2]	0	Element number in delay line for cclk_in_sample
sample_sel	CRU_EMMC_CON1[10]	0	cclk_in_sample source selection: 0: use clock after phase_shift 1: use clock after phase_shift and delay line

The following outlines the steps for clock generation sequence:

1) Assert init_state to soft reset the CLKGEN.
2) Configure drv_degree/sample_degree.
3) If fine adjustment required, delay line can be used by configuring drv_delaynum/sample_delaynum and drv_sel/sample_sel.
4) Dis-assert init_state to start CLKGEN.

10.6.11 Variable Delay Tuning

Tuning is defined by SD and MMC cards to determine the correct sampling point required for the host, especially for the speed modes SDR104 and HS200 where the output delays from the cards can be up to 2 UI. Tuning is required for other speed modes-such as DDR50-even though the output delay from the card is less than one cycle.
Command for tuning is different for different cards.

- SD Memory Card:
  - CMD19 – SD card for SDR50 and SDR104 speed modes. Tuning data is defined by card specifications.
  - CMD6 – SD card for speed modes not supporting CMD19. Tuning data is the 64byte SD status.
- Multimedia Card:
  - CMD21 – MMC card for HS200 speed mode. Tuning data is defined by card specifications.
  - CMD8 – MMC card for speed modes not supporting CMD21. Tuning data is 512 byte ExtCSD data.

The following is the procedure for variable delay tuning:

1) Set a phase shift of 0-degree on cclk_in_sample.
2) Send the Tuning command to the card; the card in turn sends an R1 response on the

CMD line and tuning data on the DAT line.

3) If the host sees any of the errors—start bit error, data crc error, end bit error, data read time-out, response crc error, response error—then the sampling point is incorrect.
4) Send CMD12 to bring the host controller state machines to idle.

- The card may treat CMD12 as an invalid command because the card has successfully sent the tuning data, and it cannot send a response.
- The host controller may generate a response time-out interrupt that must be cleared by software.

5) Repeat steps 2) to 4) by increasing the phase shift value or delay element number on cclk_in_sample until the correct sampling point is received such that the host does not see any of the errors.
6) Mark this phase shift value as the starting point of the sampling window.
7) Repeat steps 2 to 4 by increasing the phase shift value or delay element number on cclk_in_sample until the host sees the errors starting to come again or the phase shift value reaches 360-degree.
8) Mark the last successful phase shift value as the ending point of the sampling window.

A window is established where the tuning block is matched. For example, for a scenario where the tuning block is received correctly for a phase shift window of 90-degree and 180-degree, then an appropriate sampling point is established as 135-degree. Once a sampling point is established, no errors should be visible in the tuning block.

10.6.12 Package Command

In order to reduce overhead, read and write commands can be packed in groups of commands-either all read or all write-that transfer the data for all commands in the group in one transfer on the bus.

Packed commands can be of two types:

- Packed Write: CMD23 → CMD25
- Packed Read: CMD23 → CMD25 → CMD23 → CMD18

Packed commands are put in packets by the application software and are transparent to the core. For more information on packed commands, refer to the eMMC specification.

10.6.13 Card Detection Method

There are many methods for SDMMC card detection.

- Method1: Using SDMMC_CDETECT register, which is value on card_detect_n input port. 0 represents presence of card.
- Method2: Using card detection unit, outputting host interrupt. The card detection unit looks for any changes in the card-detect signals for card insertion or card removal. It filters out the debounces associated with mechanical insertion or removal, and generates one interrupt to the host. You can program the debounce filter value in SDMMC_DEBNCE[23:0]. Following figure illustrates the timing for card-detect signals.

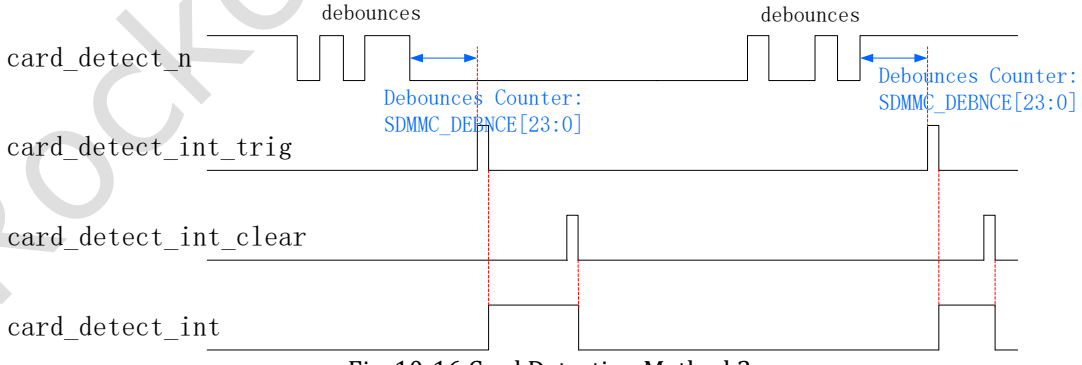


Fig. 10-16 Card Detection Method 2

- Method3: Using card detection unit in GRF, outputting sdmmc_detect_dual edge_int, only available for SDMMC. Similar to Method2, except that the debounce is configurable; and the insertion/removal detection interrupt can be enabled or cleared respectively. The detailed register information is:

Table 10-15 Register for SDMMC Card Detection Method 3

Signal Name	Source	Default	Description
sd_detectn_rise_edge_irq_en	GRF_SOC_CON0[2]	0	sdmmc detect_n signal rise edge interrupt enable.

Signal Name	Source	Default	Description
			1'b1: enable 1'b0: disable
sd_detectn_rise_edge_irq_pd	GRF_SOC_CON0[0]	0	sdmmc detect_n rise edge interrupt pending status. Write 1 to clear the status.
sd_detectn_fall_edge_irq_en	GRF_SOC_CON0[3]	0	sdmmc detect_n signal fall edge interrupt enable. 1'b1: enable 1'b0: disable
sd_detectn_fall_edge_irq_pd	GRF_SOC_CON0[1]	0	sdmmc detect_n fall edge interrupt pending status. Write 1 to clear the status.
grf_filter_cnt_sel	GRF_SOC_CON0[5:4]	0	the counter select for sd card detect filter: 2'b00: 5ms 2'b01: 15ms 2'b10: 35ms 2'b11: 50ms
sd_detectn_rise_edge_irq_en	GRF_SOC_CON0[2]	0	sdmmc detect_n signal rise edge interrupt enable. 1'b1: enable 1'b0: disable
sd_detectn_rise_edge_irq_pd	GRF_SOC_CON0[0]	0	sdmmc detect_n rise edge interrupt pending status. Write 1 to clear the status.

- Method4: Using card_detect_n for interrupt source, connecting to *IRQ* (IRQ_ID[78]) directly.

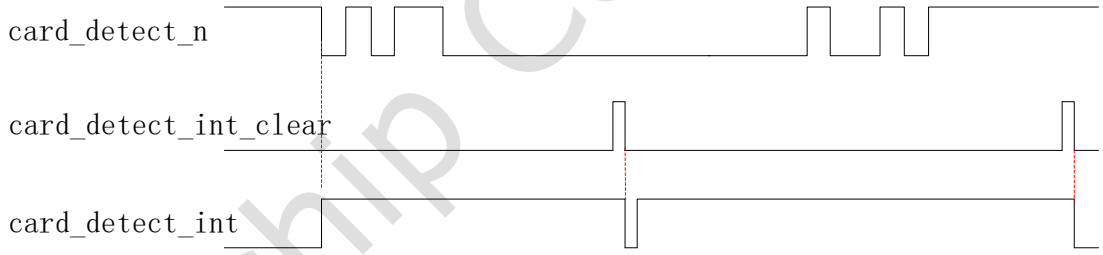


Fig. 10-17 Card Detection Method 4

10.6.14 SDMMC IOMUX With JTAG

The IO for sdmmc_cdata2/sdmmc_cdata3 is shared with jtag_tck/jtag_tms. The condition of usage for SDMMC or JTAG usage is as follows.

- If GRF_SOC_CON0[8](grf_force_jtag) is equal to 1 and sdmmc card is not detected within detection time(in the unit of XIN24M clock), the GPIOs are used for JTAG.
- Otherwise, the GPIOs' usage is defined by IOMUX configuration.

Rockchip
PX3 SE
Technical Reference Manual
Part2

Revision 1.0
July. 2017

Revision History

Date	Revision	Description
2017-7-21	1.0	Initial Release

Chapter 11 Audio codec

11.1 Overview

The Audio Codec is a capless, low power, high resolution, stereo CODEC solution that employs Sigma-Delta noise-shaping technique.The ADC with 24bit resolution, DAC with 24bit resolution and power amplifier are integrated.

Key Features

- 24 bit DAC with 95dB SNR
- Support DC-coupled capless headphone output
- Support 16Ω to 32Ω headphone output and speaker output
- 24 bit ADC with 92dB SNR
- Support single-ended and differential microphone input and line input
- Automatic Level Control (ALC) for smooth audio recording
- Support Mono, Stereo, 5.1 and 7.1 HiFi channel performance
- Programmable input and output analog gains
- Digital interpolation and decimation filter integrated
- Sampling rate of 8/12/16/24/32/44.1/48/96kHz

11.2 Block Diagram

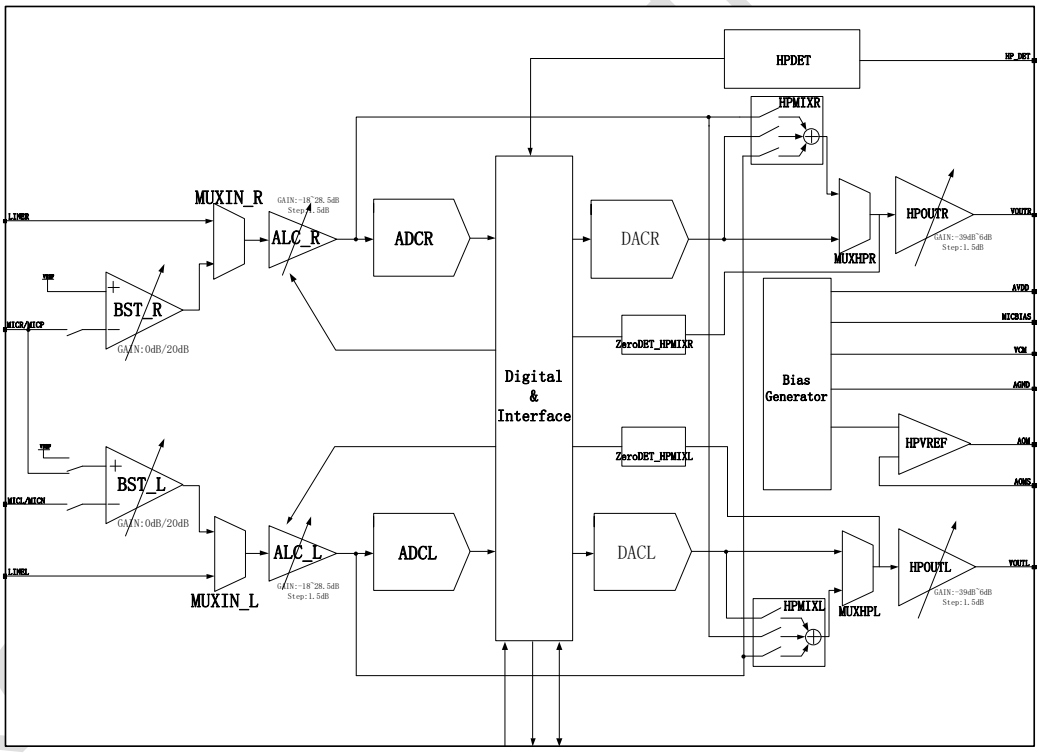


Fig. 11-1 Audio Codec Block Diagram

11.3 Electrical Specification

Test conditions: AVDD = 3.3V, DVDD = 1.1V, T_A = 25℃, 1KHz Sine Input, Fs = 48KHz.

Parameter	Symbol	Condition	Min	Typ	Max	Unit
Operating Condition						
Analog Supply	AVDD		2.97	3.3	3.63	V
Digital Supply	DVDD		0.99	1.1	1.21	V
Junction Temperature	T_J		-40		125	℃

Microphone Bias						
Bias Voltage	V_{MICB}		0.5* AVDD		0.85* AVDD	V
Bias Current	I_{MICB}				3	mA
Microphone Gain Boost PGA						
Programmable Gain	G_{BST}		0		20	dB
Gain Step Size				20		dB
Input Resistance	R_{IN}	G_{BST}=0dB		83		KΩ
		G_{BST}=20dB		15		KΩ
Input Capacitance	C_{IN}			10		pF
ALC PGA						
Programmable Gain	G_{ALC}		-18		28.5	dB
Gain Step Size				1.5		dB
ADC Input Path (Microphone or Line input to ADC)						
Signal to Noise Ratio	SNR	A-weighted		95		dB
Total Harmonic Distortion	THD	-3dBFS input		-83		dB
Power Supply Rejection	PSRR	1KHz		80		dB
ADC						
Signal to Noise Ratio	SNR	A-weighted		92		dB
Total Harmonic Distortion	THD	-3dBFS input		-81		dB
Channel Separation				80		dB
DAC						
Signal to Noise Ratio	SNR	A-weighted		95		dB
Total Harmonic Distortion	THD	-3dBFS output 10KΩ load		-84		dB
Channel Separation				85		dB
Output Driver						
Programmable Gain	G_{DRV}		-39		6	dB
Gain Step Size				1.5		dB
Output Resistance	R_{OUT}			1		KΩ
Output Capacitance	C_{OUT}			20		pF
Power Supply Rejection	PSRR	1KHz		70		dB
Line Output						
Signal to Noise Ratio	SNR	A-weighted		93		dB
Total Harmonic Distortion	THD	-3dBFS output 10KΩ load		-84		dB
Headphone Output						
Signal to Noise Ratio	SNR	A-weighted		92		dB
Total Harmonic Distortion	THD	16Ω load P_O=20mW		-70		dB
		32Ω load P_O=20mW		-75		dB
Power Consumption						
Standby				0.05		mA
Stereo Recording				6.5		mA
Stereo Playback		Quiescent output		11		mA

11.4 Function description

11.4.1 Digital Interface

The Codec has the I2S PCM interface of audio data stream in for DAC and out for ADC, both of which can be configured in master or slave mode. Different audio data formats are available for different operating modes, which are demonstrated in below table.

Table 11-1 Supported Data Formats in Different Modes

Data Formats	ADC		DAC	
	Master	Slave	Master	Slave
Left Justified	√	×	√	√
Right Justified	√	√	√	√
I²S	√	√	√	√
DSP/PCM mode A	√	√	√	√
DSP/PCM mode B	√	×	√	√

I2S_PCM interface supports five audio data formats: Left Justified mode, Right Justified mode, I²S mode, DSP/PCM mode A and mode B. They are valid when the device operates as a master or slave.

For Left Justified mode, the data format is illustrated in following figure.The MSB is valid at the first rising edge of sck after ws transition is done. The other valid bits up to the LSB are transmitted sequentially. Due to varied word length, different sck frequency and sample rate, some unused sck cycles may appear before every ws transition, which means the data in this period is invalid.

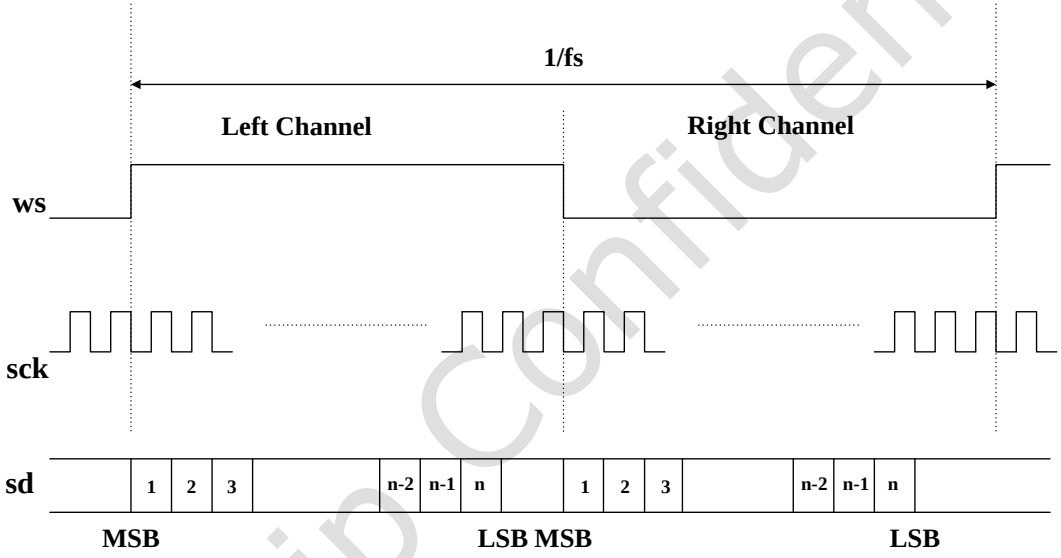


Fig. 11-2 Left Justified Mode (assuming n-bit word length)

For Right Justified mode, the data format is shown in in following figure. The LSB becomes valid at the last rising edge of sck before ws transition is done. As the MSB is transmitted first, the other valid bits up to the MSB are followed in order. Due to varied word length, different sck frequency and sample rate, some unused sck cycles may exist after every ws transition, which means the data in this period is invalid.

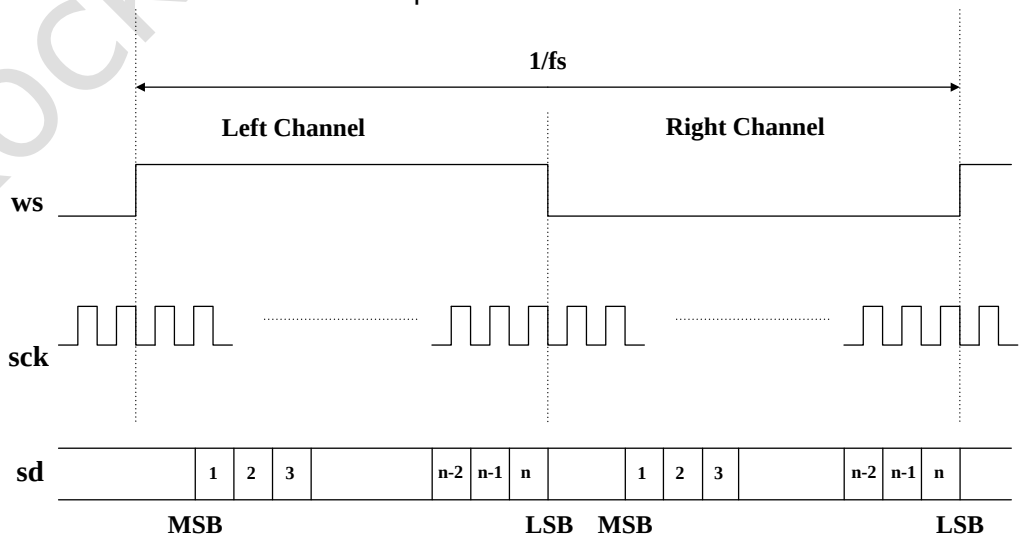


Fig. 11-3 Right Justified Mode (assuming n-bit word length)

For I2S mode, the data format is depicted in following figure. The MSB becomes available at the second rising edge of sck when ws transition is done. The other valid bits up to the LSB are transmitted in order. Due to varied word length, different sck frequency and sample rate, some unused sck cycles may appear between the LSB of the current sample and the MSB of the next one, which means the data in this period can be ignored.

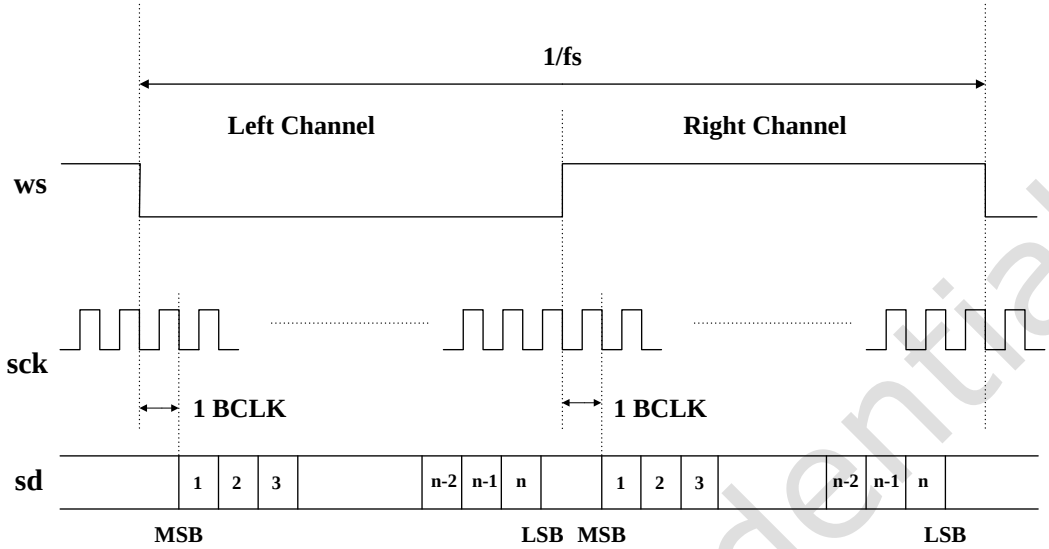


Fig. 11-4 I2S Mode (assuming n-bit word length)

For DSP/PCM mode, the left channel data is transmitted first, followed by right channel data. For DSP/PCM mode A/B, the MSB is available at the second and first rising edge of sck after the rising edge of ws respectively, as shown in Fig.11-5 and Fig.11-6. Based on word length, sck frequency and sample rate, there may be some invalid data between the LSB of the right channel data and the next sample.

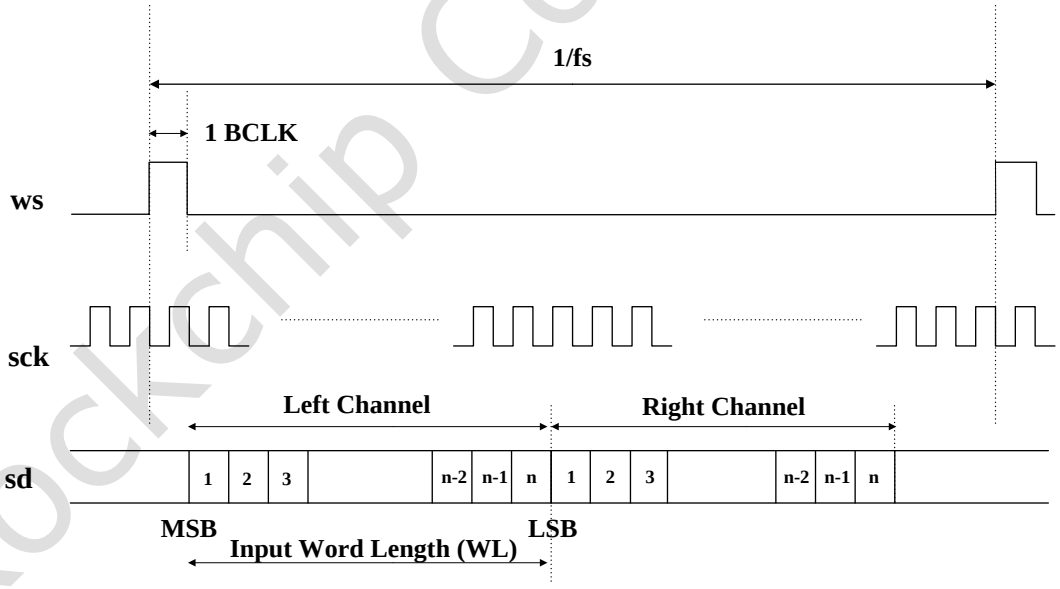


Fig. 11-5 DSP/PCM Mode A (assuming n-bit word length)

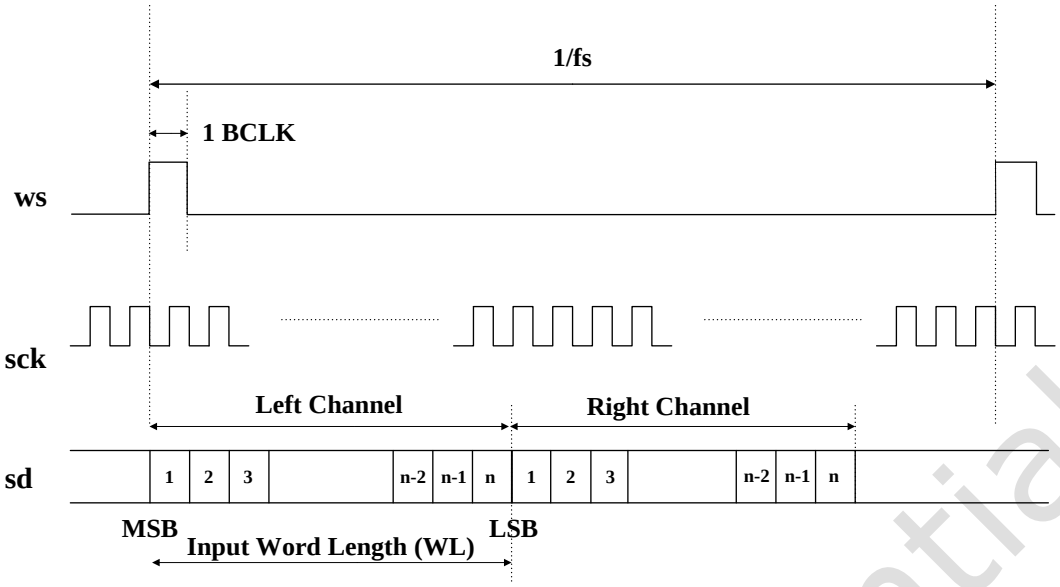


Fig. 11-6 DSP/PCM Mode B (assuming n-bit word length)

11.4.2 Analog Interface

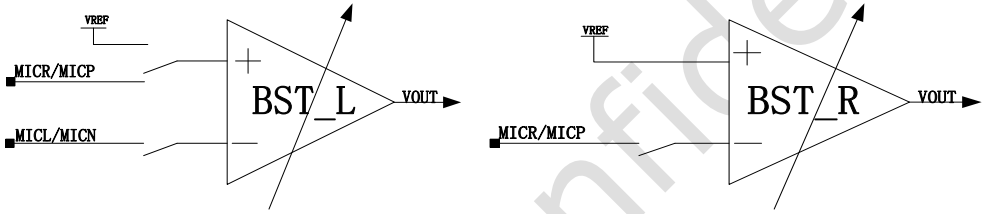


Fig. 11-7 MicroPhone Input

There are two microphone input channels, left and right channel. In left channel, there are two differential inputs, and they can be configured as either single-ended input or differential inputs by the microphone PGA (BST_L). In right channel, there is only one input, and it is configured as single-ended input by the microphone PGA (BST_R).

In left channel, microphone inputs are MICL and MICR. When working in single-ended configuration, the input signal should be input through MICL.

In right channel, microphone input is MICR.

Microphone PGA has two gains to amplify the input signal, that is, 0dB and +20dB.

There are two line input channels, INL and INR. They are input to left and right channel MUX (MUXIN_L and MUXIN_R), respectively. In each channel, the input MUX can choose line input or microphone PGA output as the input of ALC PGA.

Automatic Level Control (ALC) function is included to adjust the signal level, which is input into ADC. ALC will measure the signal magnitude and compare it to defined threshold. Then it will adjust the ALC controlled PAG (ALC_L and ALC_R) gain according to the comparison result.

The programmable gain range of ALC controlled PAG is from -18dB to +28.5dB. The tuning step is 1.5dB.

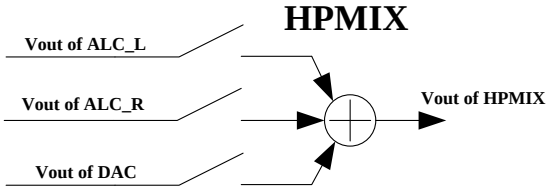


Fig. 11-8 output mixer

DAC output and ADC input can be mixed by output mixer. There are two output channel mixers, HPMIXL and HPMIXR.

In HPMIXL mixer, output of left channel DAC, output of left channel ALC PGA and output of right channel ALC PGA can be mixed. In HPMIXR mixer, output of right channel DAC, output of left channel ALC PGA and output of right channel ALC PGA can be mixed.

This Codec supports two headphone output configurations. The headphone output can drive 16Ω or 32Ω headphone load either through DC-blocking capacitor or DC-coupled capless configuration.

In the configuration using DC-blocking capacitor, shown in following figure, the headphone ground is connected to the real ground. The capacitance and the load resistance determine the lower cut-off frequency. For instance, if 16Ω headphone and 100uF DC-blocking capacitor are used, the lower cut-off frequency is:

$$f = \frac{1}{2\pi RC} = \frac{1}{2\pi \times 16 \times 100 \times 10^{-6}} = 99.5Hz$$

The DC-blocking capacitor can be increased to lower the cut-off frequency for better bass response.

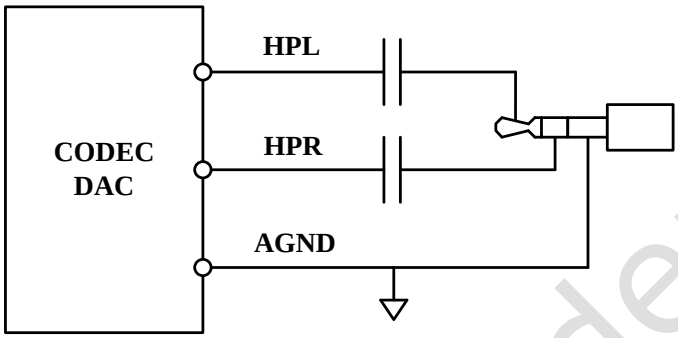


Fig. 11-9 DC-blocking capacitor

In the DC-coupled capless configuration, shown in following figure, the headphone ground is connected to a virtual ground, AOM. AOM is a DC output driver with a DC voltage of AVDD/2, that is, half of the analog supply. The requirement for DC-blocking capacitor is removed, which can save the cost.

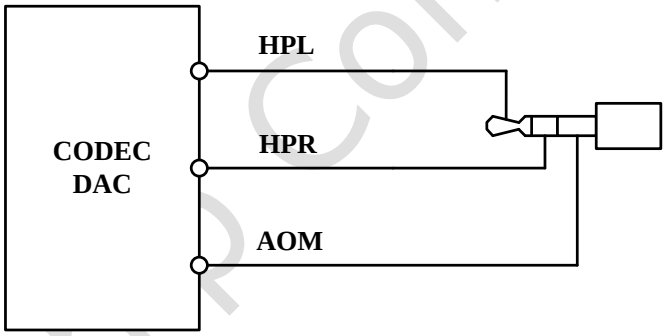


Fig. 11-10 DC-coupled capless

The headphone driver can choose mixer output or DAC output as input. It has a gain range from -39dB to +6dB with a tuning step of 1.5dB.

Microphone bias output is used to bias external microphones. The bias voltage can varies from 0.5*AVDD to 0.85* AVDD with a step of 0.05*AVDD.

11.4.3 Interface Relationship

In broadcasting application, the I2S/PCM1/2 controller is used as a transmitter and audio CODEC is used as a receiver. In recording application, the I2S/PCM1/2 controller is used as a receiver and audio CODEC is used as a transmitter. Either the I2S/PCM1/2 controller or the audio CODEC can act as a master or a slave, but if one is master, the other must be slave.

Fig.11-11 and Fig.11-12 illustrate the relationship between I2S interface and the parallel audio data in ADC and DAC channels.

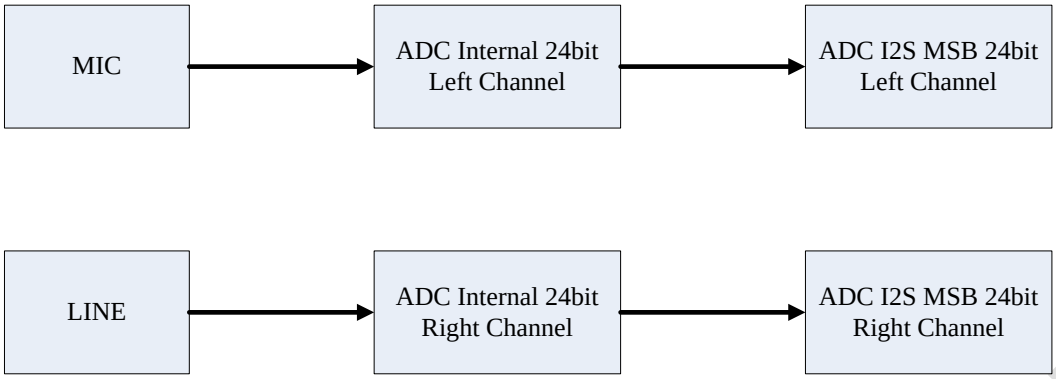


Fig. 11-11 ADC Channels Relationship

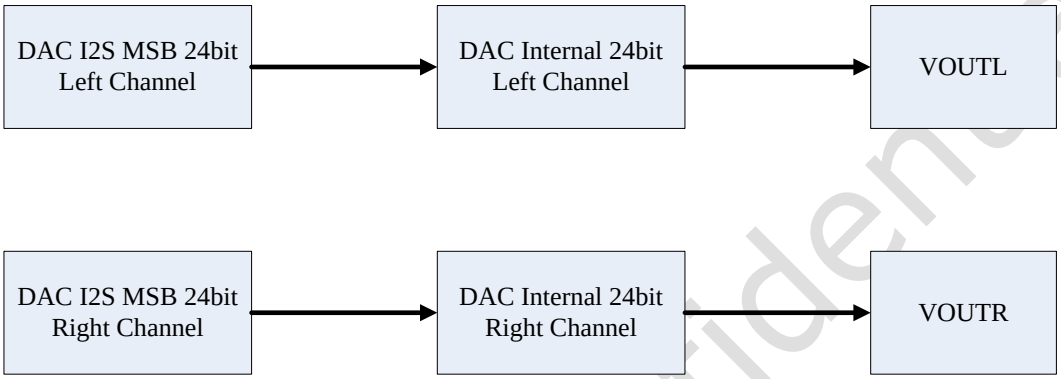


Fig. 11-12 DAC Channels Relationship

Chapter 12 USB2.0 Host

12.1 Overview

USB2.0 host controller supports fully USB2.0 functions with one EHCI host controller and one OHCI host controller, and each host controller has one USB port. OHCI host controller only supports full-speed and low-speed mode and is used for full-speed devices and low-speed devices. EHCI only supports high-speed mode and is used for high-speed devices. OHCI host controller and EHCI host controller share the same USB port. USB host controller will automatically select the owner (OHCI or EHCI) of this USB port depending on the speed mode of attached devices.

USB2.0 Host Controller supports the following features:

- Compatible Specifications
  - Universal Serial Bus Specification, Revision 2.0
  - Enhanced Host Controller Interface Specification (EHCI), Revision 1.0
  - Open Host Controller Interface Specification (OHCI), Revision 1.0a
- Support High-speed (480Mbps), Full-speed (12Mbps) and Low-speed (1.5Mbps)

12.2 Block Diagram

USB2.0 Host Controller comprises with:

- EHCI Host Controller: Perform High-speed transactions
- OHCI Host Controller: Perform full/low-speed transactions
- Port Routing Control: Select EHCI Host Controller or OHCI Host Controller

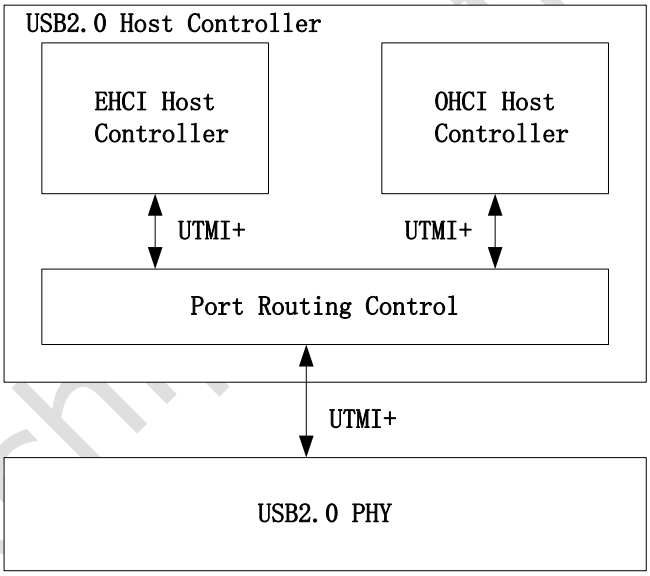


Fig. 12-1 USB2.0 Host Controller Block Diagram

12.3 Function Description

12.3.1 EHCI Host Controller

It performs descriptors and data read or write from or to system memory and packs or un-pack USB transactions from or to UTMI+ interface defined in EHCI specification for high-speed data transmission.

12.3.2 OHCI Host Controller

It performs descriptors and data read/write from/to system memory and packs or un-pack USB transactions from or to UTMI+ interface defined in OHCI specification for full-speed or low-speed data transmission.

12.3.3 Port Routing Control

As part of logic in the EHCI host controller, it is used to automatically select EHCI or OHCI host controller to serve the attached device depending on the speed of the attached device.

12.4 Register Description

12.4.1 Internal Address Mapping

Slave address can be divided into different length for different usage, which is shown as follows.

Table 12-1 USB2.0 Host Controller Address Mapping

Base Address[16]	Device	Address Length	Offset Address Range
1'b0	EHCI	128K BYTE	0x00000 ~ 0x1ffff
1'b1	OHCI	128K BYTE	0x20000 ~ 0x3ffff

EHCI and OHCI register definitions, please refer to Enhanced Host Controller Interface Specification (EHCI), Revision 1.0 and Open Host Controller Interface Specification (OHCI), Revision 1.0a.

12.5 Interface Description

Table 12-2 USB2.0 PHY Interface Description

Module Pin	Direction	Pad Name	Descriptions
USB1PN	I/O	IO_USB1_PN	USB2.0 PHY Host Port PN
USB1PP	I/O	IO_USB1_PP	USB2.0 PHY Host Port PP
USBRBIAS	I/O	IO_USB0_RBIAS	USB2.0 PHY Shared RBIAS

12.6 Application Notes

12.6.1 Special Setting

Set USB2.0 host controller master secure setting (see Chapter SGRF) before initialization.

12.6.2 Program flow

Please refer to Enhanced Host Controller Interface Specification (EHCI), Revision 1.0 and Open Host Controller Interface Specification (OHCI), Revision 1.0a.

12.6.3 Relative GRF Registers

GRF_USBPHY_CON0 ~ GRF_USBPHY_CON15 contain 256 bit registers to configure USB PHY. These bits are used to adjust DP/DM SI.

GRF_UOC1_CON1 ~ GRF_UOC1_CON5 contain some bit registers to configure USB HOST Controller. These bits are used to control UTMI+ and AHB interface.

GRF_SOC_STATUS0 and GRF_UOC_STATUS0 contain some bit status of USB HOST Controller.

Please refer to "Chapter GRF" of part1 for more details.

Chapter 13 USB OTG2.0

13.1 Overview

USB OTG 2.0 is a Dual-Role Device controller, which supports both device and host functions and is fully compliant with OTG Supplement to USB2.0 specification and support high-speed (480Mbps), full-speed (12Mbps), low-speed (1.5Mbps) transfer.
USB OTG 2.0 is optimized for portable electronic devices, point-to-point applications (no hub, direct connection to device) and multi-point applications to devices.

13.1.1 Features

- Compliant with the OTG Supplement to the USB2.0 Specification
- Operates in High-Speed and Full-Speed mode
- Support 9 channels in host mode
- 9 Device mode endpoints in addition to control endpoint 0, 4 IN, 3 OUT and 2 IN/OUT
- Built-in one 1024x35 bits FIFO
- Internal DMA with scatter/gather function
- Supports packet-based, dynamic FIFO memory allocation for endpoints for flexible, efficient use of RAM

13.2 Block Diagram

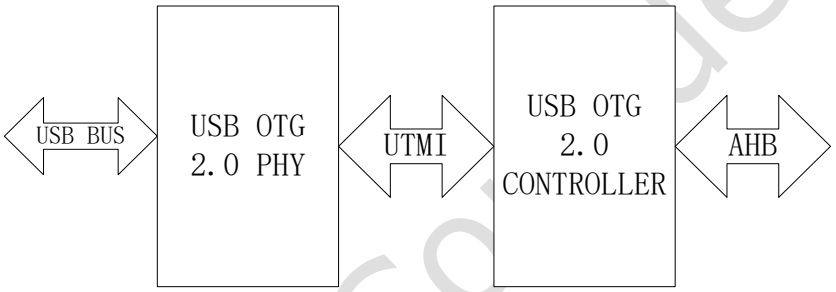


Fig. 13-1 USB OTG 2.0 Architecture

The Fig.1-1 shows the architecture of USB OTG 2.0. It is broken up into two separate units: USB OTG 2.0 controller and USB OTG 2.0 PHY. The two units are interconnected with UTMI interface.

13.3 USB OTG2.0 Controller

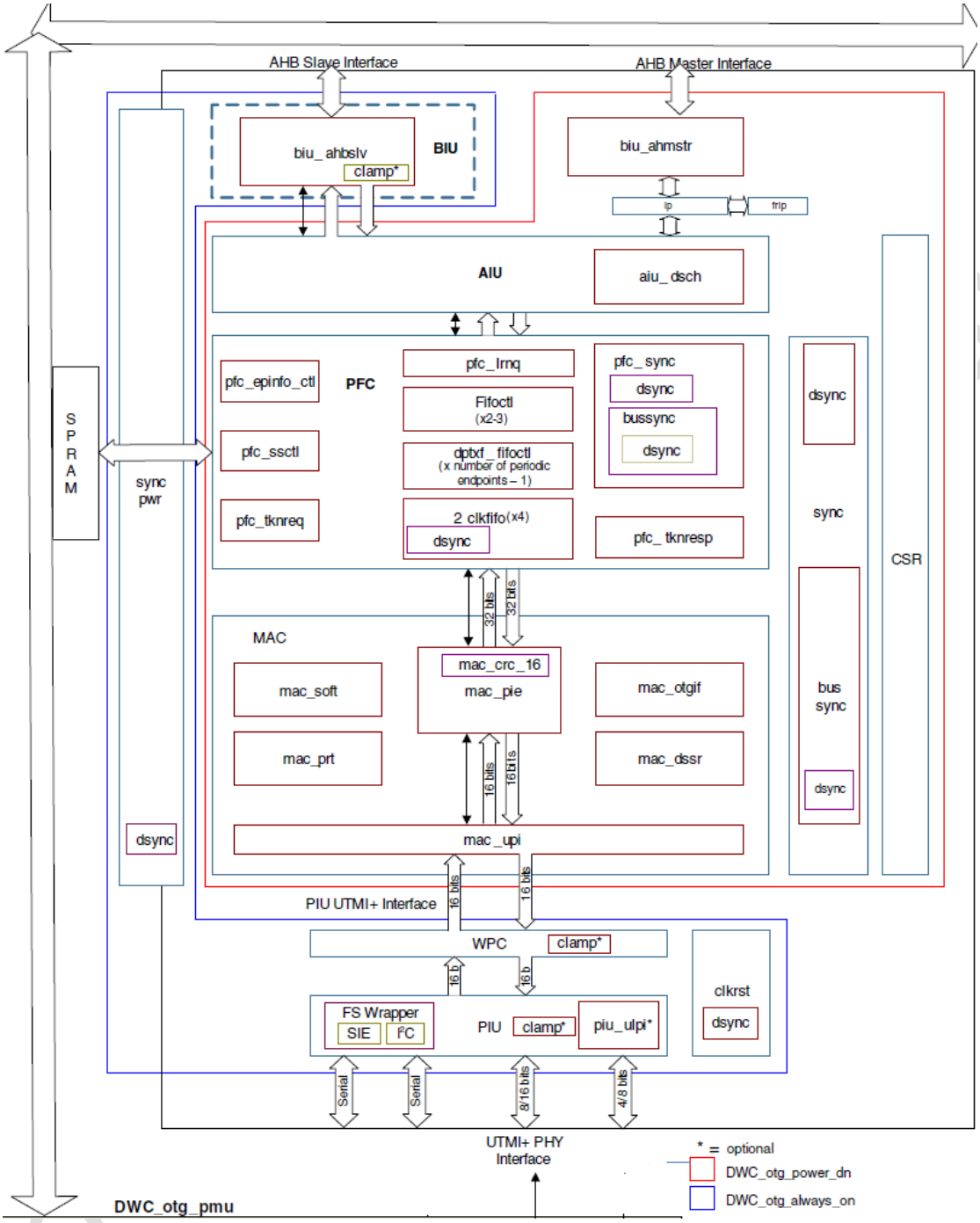


Fig. 13-2 USB OTG2.0 Controller Architecture

1). AHB Slave Bus Interface Unit (BIUS)

The AHB Slave interface unit converts AHB cycles to CSR write/read, Data-FIFO read/write, and DFIFO push/pop signals.

2) Control and Status Registers (CSR)

The CSR block resides in the AHB clock domain, and contains all registers except the Power and Clock Gating Control Register (PCGCCTL) and bits 31:29 of the Core Interrupt register (GINTSTS).

3) Application Interface Unit (AIU)

The application Interface Unit (AIU) consists of the following interfaces:

AHB Master

AHB Slave

Packet FIFO Controller

Control and Status registers

4) DMA Scheduler (DSCH)

This block is used only in DMA mode. It controls the transfer of data packets between the system memory and the USB OTG 2.0 Controller for both Internal and External DMA.
5) Packet FIFO Controller (PFC)
Several FIFOs are used in Device and Host modes to store data inside the core before transmitting it on either the AHB or the USB. PFC connects the Data FIFO interface to an industry-standard, single-port synchronous SRAM. Address, write data, and control outputs are driven late by the USB OTG 2.0 Controller, but in time to meet the SRAM setup requirements. Input read data is expected late from the SPRAM and registered inside the core before being used.

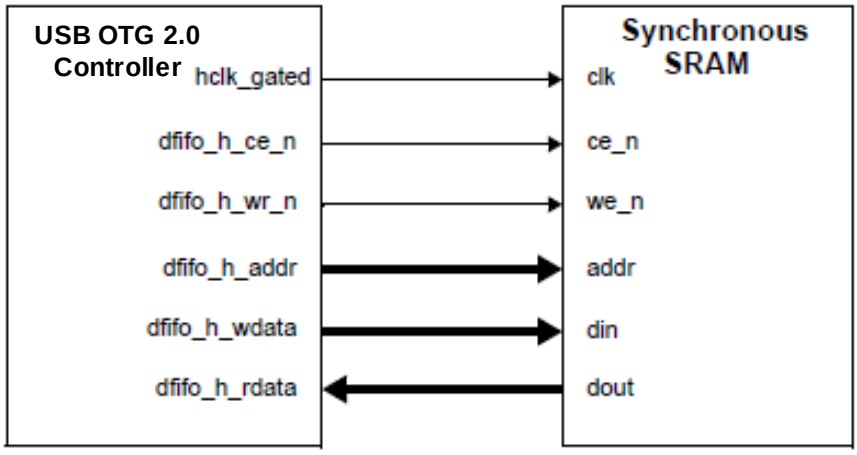


Fig. 13-3 DFIFO single-port synchronous SRAM interface

6) Media Access Controller (MAC)
The Media Access Controller (MAC) module handles USB transactions, and device, host, and OTG protocols.
7) PHY Interface Unit (PIU)
The core uses 16-bit UTMI+ Interface.
8) Wakeup and Power Controller (WPC)
When the USB is suspended or the session is not valid, the PHY is driven into Suspend mode and the PHY clock is stopped to reduce PHY and the core power consumption. To reduce power consumption further, the core also supports AHB clock gating.

13.3.1 Host Architecture

The host uses one transmit FIFO for all non-periodic OUT transactions and one transmit FIFO for all periodic OUT transactions. These transmit FIFOs are used as transmit buffers to hold the data (payload of the transmit packet) to be transmitted over USB.
The host pipes the USB transactions through Request queues (one for periodic and one for non-periodic). Each entry in the Request - queue holds the IN or OUT channel number along with other information to perform a transaction on the USB. The order in which the requests are written into the queue determines the sequence of transactions on the USB. The host processes the periodic Request queue first, followed by the non-periodic Request queue, at the beginning of each (micro) frame.
The host uses one Receive-FIFO for all periodic and non-periodic transactions. The FIFO is used as a Receive-buffer to hold the received data (payload of the received packet) from the USB until it is transferred to the system memory. The status of each packet received also goes into the FIFO. The status entry holds the IN channel number along with other information, such as received byte count and validity status, to perform a transaction on the AHB.

13.3.2 Device Architecture

The core uses Dedicated Transmit FIFO Operation. In this mode, there are individual transmit FIFOs for each IN endpoint.
The OTG device uses a single receive FIFO to receive the data for all the OUT endpoints. The receive FIFO holds the status of the received data packet, such as byte count, data PID and the validity of the received data. The DMA or the application reads the data out of the receive FIFO as it is received.

13.3.3 FIFO Mapping

- FIFO mapping in Host mode.

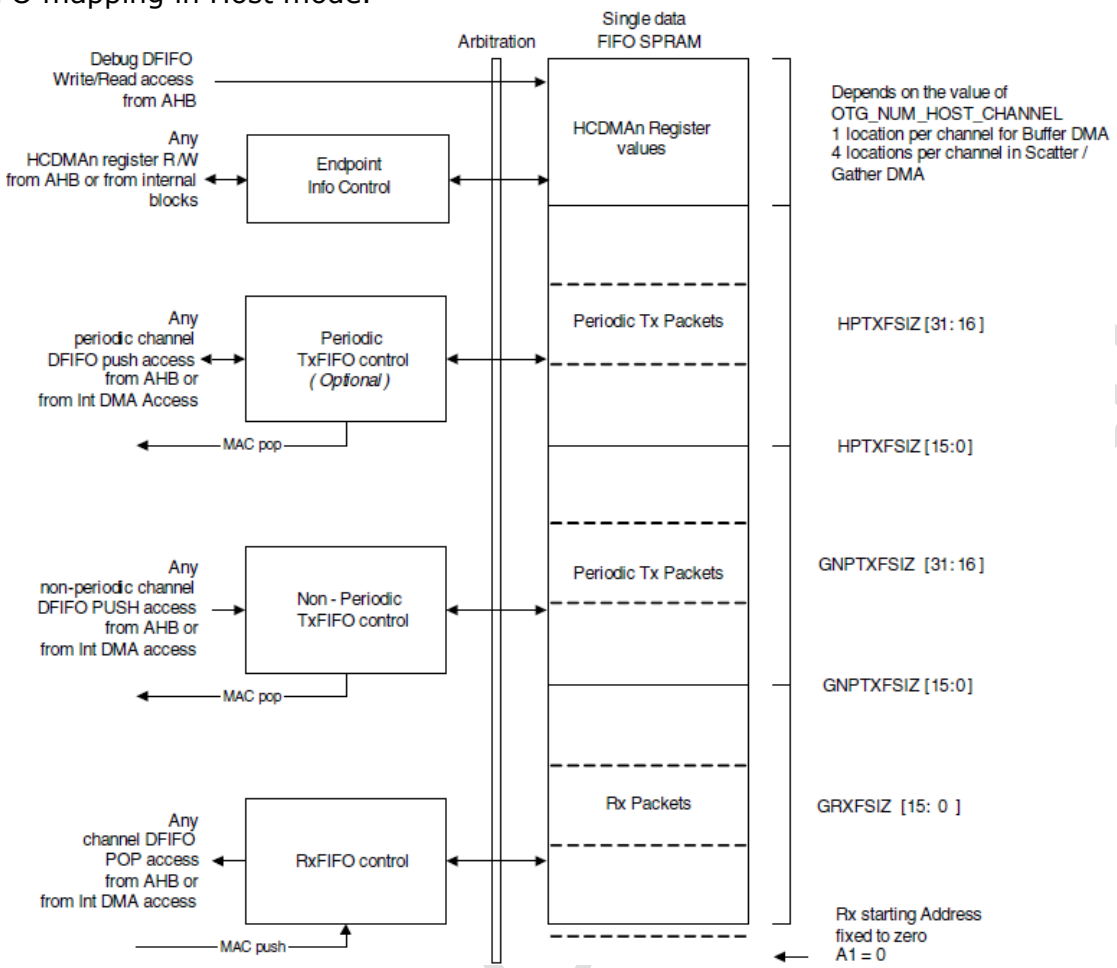


Fig. 13-4 USB OTG 2.0 Controller host mode FIFO address mapping

Note: When the device is operating in Internal DMA mode, the last locations of the SPRAM are used to store the DMAADDR values for each channel.

- FIFO mapping in Device mode.

When the device is operating in non-Descriptor Internal DMA mode, the last locations of the SPRAM are used to store the DMAADDR values for each channel. When the device is operating in Descriptor mode, then the last locations of the SPRAM store the Base Descriptor address, Current Descriptor address, Current Buffer address, and status information for each endpoint direction.

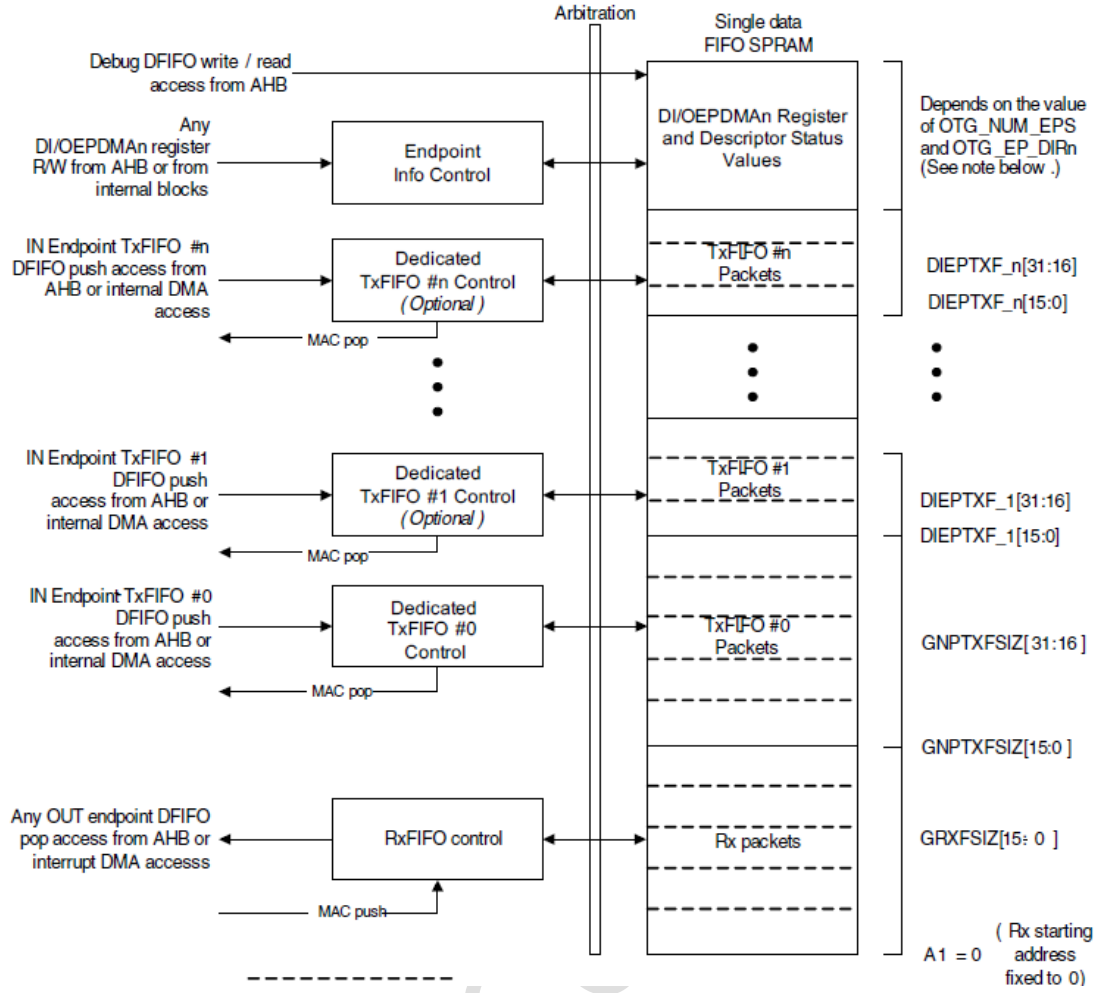


Fig. 13-5 USB OTG 2.0 Controller device mode FIFO address mapping

Note: When the device is operating in non-Scatter Gather Internal DMA mode, the last locations of the SPRAM are used to store the DMAADDR values for each Endpoint (1 location per endpoint).When the device is operating in Scatter Gather mode, then the last locations of the SPRAM store the Base Descriptor address, Current Descriptor address, Current Buffer address, and status information for each endpoint direction (4 locations per Endpoint). If an Endpoint is bidirectional, then 4 locations will be used for IN, and another 4 for OUT).

13.4 USB OTG2.0 PHY

USB PHY supports dual OTG ports' functions and is fully compliant with USB2.0 specification, and support High-speed (480Mbps), full-speed (12Mbps), low-speed (1.5Mbps) transfer. It provides a complete on-chip transceiver physical solution with ESD protection. A minimal number of external components are needed, which include a 45 ohm resistor for resistance calibration purpose. Its feature contains:

- provide dual UTMI ports
- OTG0 Support UART Bypass Function
- Fully compliant with USB specifications Rev 2.0, 1.1 HOST/Device and OTG V1.2.
- Supports 480Mbps (HS), 12Mbps (FS) & 1.5Mbps (LS) serial data transmission
- Supports low latency hub mode with 40 bit time-round trip delay
- 8 bit or 16 bit UTMI interface compliant with UTMI+ specification level 3 Rev 1.
- Loop back BIST mode supported
- Built-in I/O and ESD structure
- On-die self-calibrated HS/FS/LS termination
- 12MHz crystal oscillator with integrated phase-locked loop (PLL) oscillator
- Dual 3.3V / 1.2V supply

13.4.1 Block Diagram

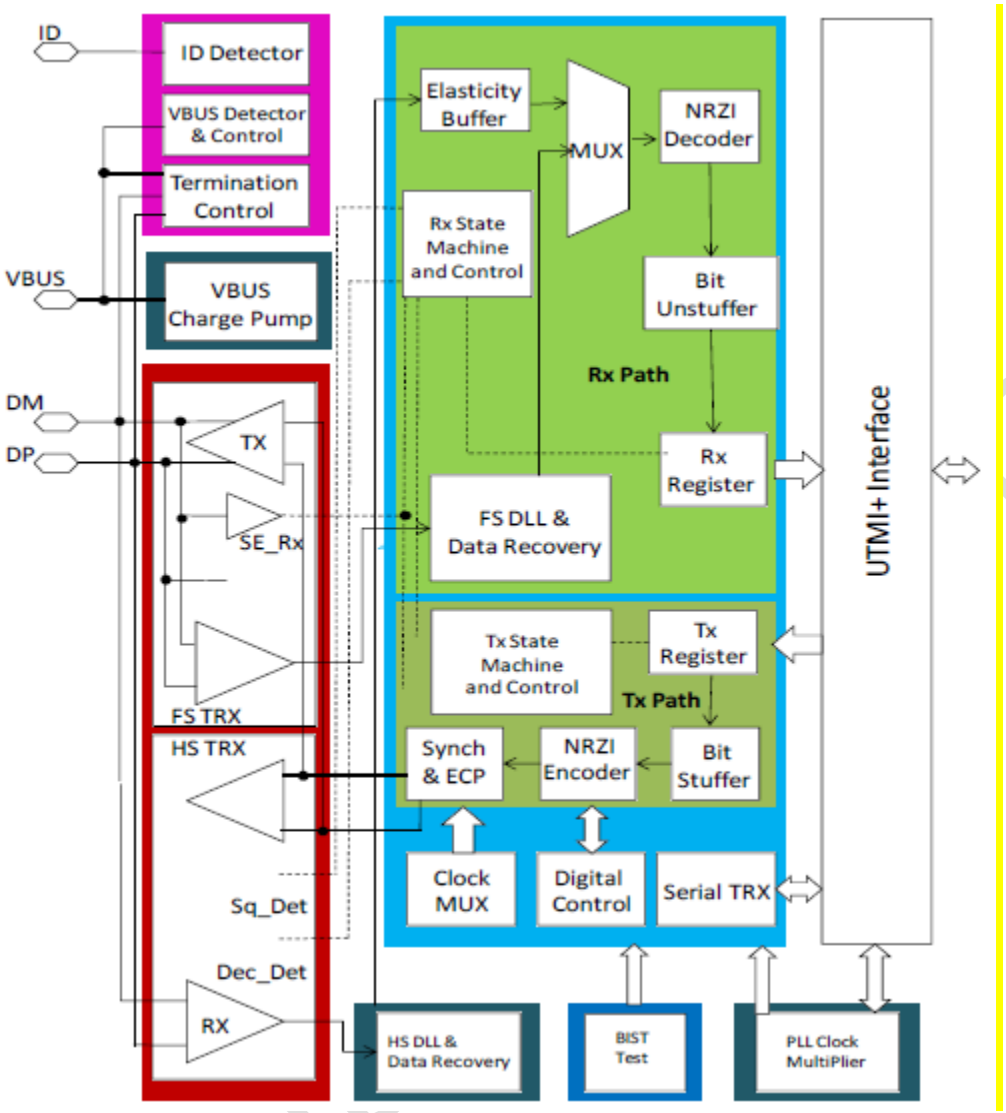


Fig. 13-6 USB PHY Architecture

HS AFE

The HS AFE contains the low-level analog circuitry, and also the HS differential data transmitter and receiver, toper form HS transmission envelope detection and host disconnection detection. It works in HS mode only.

HS Transmit driver

The HS transmit driver is active only when transmit is asserted. In HS transceiver enabled mode and the transmit state machine has data to send, the XCVR selects input Data from transmit data path which will be driven onto the DP/DM signal lines when enabled.

HS Differential Receiver

When enabled, received HS data will be multiplexed through the receive data path to the receive shift and hold registers. It is active only in HS mode.

Transmission envelope detector (Squelch detector)

When the amplitude of the differential signal at a receiver's inputs falls below the squelch threshold, the envelope detector will indicate the invalid data. It must indicate squelch when the signal drops below 100mV differential amplitude, and also, it must indicate that the line is not in the squelch state when the signal exceeds 150mV differential amplitude.

Disconnection envelope detector

In host mode, this envelope detector is active to detect the high speed disconnect state on the line Disconnection must be indicated when the amplitude of the differential signal at the downstream facing driver's connector is more than 625 mV, and it must not be indicated

when the signal amplitude is less than525 mV.

FS/LS AFE

In FS or LS mode, the FS/LS AFE is active to send and receive the FS or LS data on the USB bus. Also it supports the reset, suspend and resume detection through the data line single ended receivers.

FS/LS Transmitter

The FS/LS transmitter is active only when transmit is asserted. In FS or LS transceiver enabled mode and the transmit state machine has data to send, the XCVR selects input. Data from transmit data path will be driven onto the DP/DM signal lines when enabled.

FS/LS Differential Receiver

When enabled, received FS or LS data will be multiplexed through the receive data path to the receive shift and hold registers. It is active only in FS or LS modes.

Single ended receivers

The single ended receivers are used for low-speed and full-speed signaling detection.

Digital Core TX Path

The digital core TX path has some blocks responsible for SYNC and EOP generation, data encoding, bit stuffing and data serialization. And meanwhile, also a TX state machine is involved to manage the communication with the controller.

TX Shift/Hold Register

The TX shift/Hold register module consists of an 8-bit primary shift register for parallel/serial conversion and 8-bit hold register used to buffer the next data to serialize. This module is responsible for reading parallel data from the parallel application bus interface upon command and serializing for transmission over USB.

Bit stuffer

To ensure adequate signal transitions, when sending a packet on USB, a bit stuffer is employed by the transmitter. A '0' has to be inserted after every six consecutive ones in the data stream before the data in NRZI encoded, to force a transition in the NRZI data stream.

NRZI Encoder

The High speed, Full speed or low speed serial transmitted data are encoded by The NRZI encoder. As state transition, a '0' is encoded, and as no state transition, a '1' is encoded.

Transmit state machine

The communication between the controller and the PHY in TX path is controlled by the transmit state machine, which synchronizes the Data with the Sync and the EOP, and also supports the LS, FS and HS Modes.

Digital Core RX Path

The digital core RX path includes blocks responsible for SYNC and EOP detection and stripping, data decoding, bit un-stuffing and data de-serialization. Also a RX state machine is involved to manage the communication with the controller. FS/LS data and clock is recovered in this section.

Elasticity buffer

To compensate for differences between transmitting and receiving clocks, the Elasticity Buffer is used to synchronize the HS extracted data with the PLL internal clock.

Mux

The Mux block allows the data from the HS or FS/LS receivers to be routed to the shared receive logic. The state of the Mux is determined by the XCVR Select input.

NRZI Decoder

The NRZI is responsible for decoding the High speed or Full speed received NRZI encoded data. A change in level is decoded as '0' and no change in level is decoded as '1'.

Bit Un-stuffer

The Bit Un-stuffer not only recognizes the stuffed bits from the data stream, but also discards them. Also it detects bit stuff error, which is interpreted as HS EOP.

RX Shift/Hold Register

This module de-serializes received data and transmits 8-bit parallel data to the application bus interface. It consists of an 8-bit primary shit register for serial to parallel conversion

and an 8-bit hold register for buffering the last de-serialized data byte.

Receiver state machine

The receiver state machine controls the communication between the controller and the PHY in the RX path, strips the SYNC and the EOP from the Data and supports the LS, FS and HS Modes.

PLL Clock Multiplier

This module is composed of the off-chip crystal and the on-chip clock multiplier. It generates the appropriate internal clocks for the UTM and the CLK output signal. All data transfer signals are synchronized with the CLK signal.

External Crystal

The external crystal is composed of a precise resonance frequency crystal and a crystal oscillator. It is optional to have this crystal oscillator integrated on-chip or have it off-chip. This crystal/crystal oscillator provides a very precise clock of 12 MHz with deviation of ±100 ppm. The oscillator is not a part of the PHY, but external.

Clock Multiplier

The UTM interface is described as an un-directional/bi-directional 16-bit parallel interface and the CLK signal is a 30 MHz signal. All data transfer signals should be synchronized with the CLK signal. CLK usable signal is internally implemented which blocks any transitions of CLK until it is usable. Meanwhile, the clock multiplier provides another three clocks in addition to the CLK signal. That is a 480 MHz and 7.5 MHz clock signals.

Clock MUX

The Clock Multiplexer supplies both the transmitter and receiver paths with the adequate bit clock depending on the XcvrSelect signal and to ensure smooth clock switching. It also includes clock gating and power-down features.

Control Logic Block

This block is responsible for controlling, enabling and disabling the different blocks in the system.

OTG Circuitry （optional）

With the OTG circuitry, the system has the capability to dynamically switch between host and peripheral, enable dual role device behavior and point-to-point communication. The OTG circuitry functions as VBUS generation and detection. Both ID detection and terminations control are implemented in it.

ID Detector （optional）

To provide the ID signal that is used to indicate the state of the ID pin on the USB mini receptacle. This pin makes it able to determine which kind of plug is connected and to confirm if the device default state is A device or B device.

VBUS Detector and termination control

The VBUS detector is a set of comparators, functions to monitor and sense the voltage on USB bus power line. For VBUS signaling and discharging, VBUS pull up and pull-down resistors are also implemented.

Automatic Test Functions

Loop-back test to address all IP components.

In loop-back test mode, all transmitted data packets are received back in an internal loop to check IP functional integrity. There are some digital components that cannot be tested with the scan technique due to the high-speed nature of the digital part. To be regarded as a good idea, Loop-back allows testing full design paths at speed. It should complement the testing suite for digital core to achieve the highest coverage possible. According to the UTMI specification Section 5.18, version 1.05 Page 34, the 8 bits un-directional data bus can be implemented as 8 bits bi-directional one. This implementation will hinder the loop-back test functionality.

13.5 UART BYPASS FUNCITON

When in UART bypass mode,UART2 is connect to USB interface; Otherwise, UART2 use normal UART interface。

Signal	CONNECT	I/O	Description
BYPASSDMDATA0	UART2_SOUT	I	Data for DM0 Transmitter Digital Bypass
BYPASSDMEN0	GRF_UOC1_CON4[11]	I	DM0 Transmitter Digital Bypass Enable
BYPASSSEL0	GRF_UOC1_CON4[13]	I	Transmitter Digital Bypass mode Enable
FSVPLUS0	UART2_SIN	O	Single-Ended D- Indicator The controller signal indicates the state of the DP during normal operation or UART data reception
OTGDISABLE0	GRF_UOC1_CON4[10]	I	1'b1: OTG0 disable; 1'b0: OTG0 normal mode
COMMONONN	GRF_UOC1_CON4[15]	I	Common Block Power-Down Control This signal controls the power-down signals in PLL blocks when the USBPHY is in Suspend Mode. 1: PLL blocks are powered down. 0: PLL blocks remain powered This signal is a strapping option that must be set prior to a power-on reset and remain static during normal operation.

Note: USB OTG2.0 PHY support UART Bypass Function.

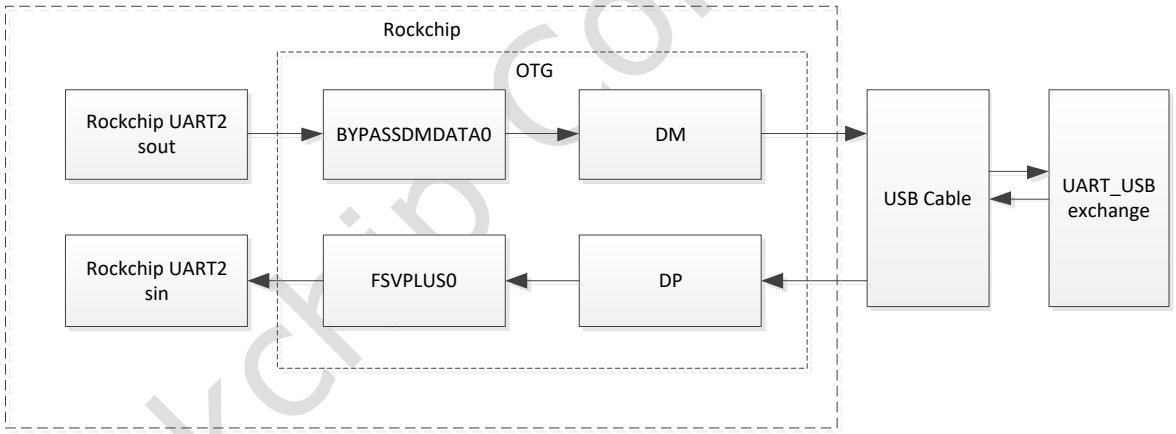


Fig. 13-7 UART Application

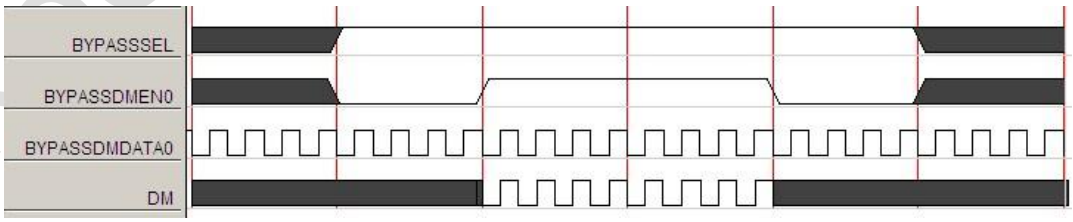


Fig. 13-8 UART Timing Sequence

To use UART and auto resume functions:
1. Disable the OTG block by setting OTGDISABLE0 to 1'b1.
2. Disable the pull-up resistance on the DP line by setting OPMODE0[1:0] to 2'b01.
3. To ensure that the XO, Bias, and PLL blocks are powered down in Suspend mode, set COMMONONN to 1'b1.
4. Place the USB PHY in Suspend mode by setting SUSPENDM0 to 1'b0.
5. Set BYPASSSEL0 to 1'b1.
6. To transmit data, controls BYPASSDMEN0, and BYPASSDMDATA0.

To receive data, monitor FSVPLUS0.

To return to normal operating mode:
1. Ensure that there is no activity on the USB.
2. Set BYPASSSEL0 to 1'b0.
3. Set SUSPENDM0 to 1’b1.Resume the USB PHY.
4. Set COMMONONN to 1’b0.
5. set OTGDISABLE0 to 1'b0.

13.6 Register Description

13.6.1 Register Summary

Name	Offset	Size	Reset Value	Description
USBOTG_GOTGCTL	0x0000	W	0x00000000	Control and Status Register
USBOTG_GOTGINT	0x0004	W	0x00000000	Interrupt Register
USBOTG_GAHBCFG	0x0008	W	0x00000000	AHB Configuration Register
USBOTG_GUSBCFG	0x000c	W	0x00001400	USB Configuration Register
USBOTG_GRSTCTL	0x0010	W	0x80000000	Reset Register
USBOTG_GINTSTS	0x0014	W	0x00000000	Interrupt Register
USBOTG_GINTMSK	0x0018	W	0x00000000	Interrupt Mask Register
USBOTG_GRXSTSR	0x001c	W	0x00000000	Receive Status Debug Read Register
USBOTG_GRXSTSP	0x0020	W	0x00000000	Receive Status Read and Pop Register
USBOTG_GRXFSIZ	0x0024	W	0x00000000	Receive FIFO Size Register
USBOTG_GNPTXFSIZ	0x0028	W	0x00000000	Non-Periodic Transmit FIFO Size Register
USBOTG_GNPTXSTS	0x002c	W	0x00000000	Non-Periodic Transmit FIFO/Queue Status Register
USBOTG_GI2CCTL	0x0030	W	0x11000000	I2C Address Register
USBOTG_GPVNDCTL	0x0034	W	0x00000000	PHY Vendor Control Register
USBOTG_GGPIO	0x0038	W	0x00000000	General Purpose Input/ Output Register
USBOTG_GUID	0x003c	W	0x00000000	User ID Register
USBOTG_GSNPSID	0x0040	W	0x00004f54	Core ID Register
USBOTG_GHWCFG1	0x0044	W	0x00000000	User HW Config1 Register
USBOTG_GHWCFG2	0x0048	W	0x00000000	User HW Config2 Register
USBOTG_GHWCFG3	0x004c	W	0x00000000	User HW Config3 Register
USBOTG_GHWCFG4	0x0050	W	0x00000000	User HW Config4 Register
USBOTG_GLPMCFG	0x0054	W	0x00000000	Core LPM Configuration Register
USBOTG_GPWRDN	0x0058	W	0x00000000	Global Power Down Register
USBOTG_GDFIFOCFG	0x005c	W	0x00000000	Global DFIFO Software Configure Register
USBOTG_GADPCTL	0x0060	W	0x00000000	ADP Timer, Control and Status Register
USBOTG_HPTXFSIZ	0x0100	W	0x00000000	Host Periodic Transmit FIFO Size Register

Name	Offset	Size	Reset Value	Description
USBOTG_DIEPTXFn	0x0104	W	0x00000000	Device Periodic Transmit FIFO-n Size Register
USBOTG_HCFG	0x0400	W	0x00000000	Host Configuration Register
USBOTG_HFIR	0x0404	W	0x00000000	Host Frame Interval Register
USBOTG_HFNUM	0x0408	W	0x0000ffff	Host Frame Number/Frame Time Remaining Register
USBOTG_HPTXSTS	0x0410	W	0x00000000	Host Periodic Transmit FIFO/Queue Status Register
USBOTG_HAINT	0x0414	W	0x00000000	Host All Channels Interrupt Register
USBOTG_HAINTMSK	0x0418	W	0x00000000	Host All Channels Interrupt Mask Register
USBOTG_HPRT	0x0440	W	0x00000000	Host Port Control and Status Register
USBOTG_HCCHARn	0x0500	W	0x00000000	Host Channel-n Characteristics Register
USBOTG_HCSPLTn	0x0504	W	0x00000000	Host Channel-n Split Control Register
USBOTG_HCINTn	0x0508	W	0x00000000	Host Channel-n Interrupt Register
USBOTG_HCINTMSKn	0x050c	W	0x00000000	Host Channel-n Interrupt Mask Register
USBOTG_HCTSIZn	0x0510	W	0x00000000	Host Channel-n Transfer Size Register
USBOTG_HCDMAn	0x0514	W	0x00000000	Host Channel-n DMA Address Register
USBOTG_HCDMABn	0x051c	W	0x00000000	Host Channel-n DMA Buffer Address Register
USBOTG_DCFG	0x0800	W	0x08200000	Device Configuration Register
USBOTG_DCTL	0x0804	W	0x00002000	Device Control Register
USBOTG_DSTS	0x0808	W	0x00000000	Device Status Register
USBOTG_DIEPMSK	0x0810	W	0x00000000	Device IN Endpoint common interrupt mask register
USBOTG_DOEPMSK	0x0814	W	0x00000000	Device OUT Endpoint common interrupt mask register
USBOTG_DAINT	0x0818	W	0x00000000	Device All Endpoints interrupt register
USBOTG_DAINTMSK	0x081c	W	0x00000000	Device All Endpoint interrupt mask register
USBOTG_DTKNQR1	0x0820	W	0x00000000	Device IN token sequence learning queue read register1
USBOTG_DTKNQR2	0x0824	W	0x00000000	Device IN token sequence learning queue read register2
USBOTG_DVBUSDIS	0x0828	W	0x00000b8f	Device VBUS discharge time register

Name	Offset	Size	Reset Value	Description
USBOTG_DVBUSPULSE	0x082c	W	0x00000000	Device VBUS Pulsing Timer Register
USBOTG_DTHRCTL	0x0830	W	0x08100020	Device Threshold Control Register
USBOTG_DIEPEMPMSK	0x0834	W	0x00000000	Device IN endpoint FIFO empty interrupt mask register
USBOTG_DEACHINT	0x0838	W	0x00000000	Device each endpoint interrupt register
USBOTG_DEACHINTMSK	0x083c	W	0x00000000	Device each endpoint interrupt register mask
USBOTG_DIEPEACHMSKn	0x0840	W	0x00000000	Device each IN endpoint -n interrupt Register
USBOTG_DOEPEACHMSKn	0x0880	W	0x00000000	Device each out endpoint-n interrupt register
USBOTG_DIEPCTL0	0x0900	W	0x00008000	Device control IN endpoint 0 control register
USBOTG_DIEPINTn	0x0908	W	0x00000000	Device Endpoint-n Interrupt Register
USBOTG_DIEPTSIZn	0x0910	W	0x00000000	Device endpoint n transfer size register
USBOTG_DIEPDMAn	0x0914	W	0x00000000	Device endpoint-n DMA address register
USBOTG_DTXFSTSn	0x0918	W	0x00000000	Device IN endpoint transmit FIFO status register
USBOTG_DIEPDMABn	0x091c	W	0x00000000	Device endpoint-n DMA buffer address register
USBOTG_DIEPCTLn	0x0920	W	0x00000000	Device endpoint-n control register
USBOTG_DOEPCTL0	0x0b00	W	0x00000000	Device control OUT endpoint 0 control register
USBOTG_DOEPINTn	0x0b08	W	0x00000000	Device endpoint-n control register
USBOTG_DOEPTSIZn	0x0b10	W	0x00000000	Device endpoint n transfer size register
USBOTG_DOEPDMAn	0x0b14	W	0x00000000	Device Endpoint-n DMA Address Register
USBOTG_DOEPDMABn	0x0b1c	W	0x00000000	Device endpoint-n DMA buffer address register
USBOTG_DOEPCTLn	0x0b20	W	0x00000000	Device endpoint-n control register
USBOTG_PCGCR	0x0b24	W	0x200b8000	Power and clock gating control register
USBOTG_EPBUF0	0x1000	W	0x00000000	Device endpoint 0 / host out channel 0 address
USBOTG_EPBUF1	0x2000	W	0x00000000	Device endpoint 1 / host out channel 1 address
USBOTG_EPBUF2	0x3000	W	0x00000000	Device endpoint 2 / host out channel 2 address

Name	Offset	Size	Reset Value	Description
USBOTG_EPBUF3	0x4000	W	0x00000000	Device endpoint 3 / host out channel 3 address
USBOTG_EPBUF4	0x5000	W	0x00000000	Device endpoint 4 / host out channel 4 address
USBOTG_EPBUF5	0x6000	W	0x00000000	Device endpoint 5 / host out channel 5 address
USBOTG_EPBUF6	0x7000	W	0x00000000	Device endpoint 6 / host out channel 6 address
USBOTG_EPBUF7	0x8000	W	0x00000000	Device endpoint 7 / host out channel 7 address

*Notes:Size:**B**- Byte (8 bits) access, **HW**- Half WORD (16 bits) access, **W**-WORD (32 bits) access*

13.6.2 Detail Register Description

USBOTG_GOTGCTL

Address: Operational Base + offset (0x0000)
Control and Status Register

Bit	Attr	Reset Value	Description
31:28	RO	0x0	reserved
27	RW	0x0	ChirpEn Chirp on enable This bit is a reserved bit.
26:22	RO	0x00	MultValidBc Multi Valued ID pin Battery Charger ACA inputs in the following order: Bit 26 - rid_float. Bit 25 - rid_gnd Bit 24 - rid_a Bit 23 - rid_b Bit 22 - rid_c These bits are reserved and will read 5'h0.
21	RO	0x0	reserved
20	RW	0x0	OTGVer OTG version Indicates the OTG revision. 0: OTG Version 1.3. In this version the core supports Data line pulsing and VBUS pulsing for SRP. 1: OTG Version 2.0. In this version the core supports only Data line pulsing for SRP.

Bit	Attr	Reset Value	Description
19	RO	0x0	BSesVld B-session valid Indicates the Device mode transceiver status. 0: B-session is not valid. 1: B-session is valid. In OTG mode, you can use this bit to determine if the device is connected or disconnected. Note: If you do not enabled OTG features (such as SRP and HNP), the read reset value will be 1.The VBUS assigns the values internally for non-SRP or non-HNP configurations.
18	RO	0x0	ASesVld A-session valid Indicates the Host mode transceiver status. 0: A-session is not valid 1: A-session is valid Note: If you do not enabled OTG features (such as SRP and HNP), the read reset value will be 1.The VBUS assigns the values internally for non-SRP or non-HNP configurations.
17	RO	0x0	DbnTime Long/short debounce time Indicates the debounce time of a detected connection. 0: Long debounce time, used for physical connections (100 ms + 2.5 us) 1: Short debounce time, used for soft connections (2.5 us)
16	RO	0x0	ConIDSts Connector ID Status Indicates the connector ID status on a connect event. 0: The core is in A-Device mode 1: The core is in B-Device mode
15:12	RO	0x0	reserved
11	RW	0x0	DevHNPEn Device HNP Enable The application sets this bit when it successfully receives a SetFeature. SetHNPEnable command from the connected USB host. 0: HNP is not enabled in the application 1: HNP is enabled in the application
10	RW	0x0	HstSetHNPEn Host set HNP enable The application sets this bit when it has successfully enabled HNP (using the SetFeature.SetHNPEnable command) on the connected device. 0: Host Set HNP is not enabled 1: Host Set HNP is enabled

Bit	Attr	Reset Value	Description
9	RW	0x0	HNPReq HNP request The application sets this bit to initiate an HNP request to the connected USB host. The application can clear this bit by writing a 0 when the Host Negotiation Success Status Change bit in the OTG Interrupt register (GOTGINT.HstNegSucStsChng) is set. The core clears this bit when the HstNegSucStsChng bit is cleared. 0: No HNP request 1: HNP request
8	RO	0x0	HstNegScs Host Negotiation Success The core sets this bit when host negotiation is successful. The core clears this bit when the HNP Request (HNPReq) bit in this register is set. 0: Host negotiation failure 1: Host negotiation success
7:2	RO	0x0	reserved
1	RW	0x0	SesReq Session Request The application sets this bit to initiate a session request on the USB. The application can clear this bit by writing a 0 when the Host Negotiation Success Status Change bit in the OTG Interrupt register (GOTGINT.HstNegSucStsChng) is set. The core clears this bit when the HstNegSucStsChng bit is cleared. If you use the USB 1.1 Full-Speed Serial Transceiver interface to initiate the session request, the application must wait until the VBUS discharges to 0.2 V, after the B-Session Valid bit in this register (GOTGCTL.BSesVld) is cleared. This discharge time varies between different PHYs and can be obtained from the PHY vendor. 0: No session request 1: Session request
0	RO	0x0	SesReqScs Session Request Success The core sets this bit when a session request initiation is successful. 0: Session request failure 1: Session request success

USBOTG_GOTGINT

Address: Operational Base + offset (0x0004)

Interrupt Register

Bit	Attr	Reset Value	Description
31:21	RO	0x0	reserved

Bit	Attr	Reset Value	Description
20	W1C	0x0	MultiValueChg Multi-Valued input changed This bit is reserved.
19	W1C	0x0	DbnceDone Debounce Done The core sets this bit when the debounce is completed after the device connected. The application can start driving USB reset after seeing this interrupt. This bit is only valid when the HNP Capable or SRP Capable bit is set in the Core USB Configuration register (GUSBCFG.HNPCap or GUSBCFG.SRPCap, respectively).
18	W1C	0x0	ADevTOUTChg A-Device Timeout Change The core sets this bit to indicate that the A-device has timed out while waiting for the B-device to connect.
17	W1C	0x0	HstNegDet Host Negotiation Detected The core sets this bit when it detects a host negotiation request on the USB
16:10	RO	0x0	reserved
9	W1C	0x0	HstNegSucStsChng Host Negotiation Success Status Change The core sets this bit on the success or failure of a USB host negotiation request. The application must read the Host Negotiation Success bit of the OTG Control and Status register (GOTGCTL.HstNegScs) to check for success or failure
8	W1C	0x0	SesReqSucStsChng Session Request Success Status Change The core sets this bit on the success or failure of a session request.The application must read the Session Request Success bit in the OTG Control and Status register (GOTGCTL.SesReqScs) to check for success or failure.
7:3	RO	0x0	reserved
2	W1C	0x0	SesEndDet Session End Detected The core sets this bit when the utmisrp_bvalid signal is deasserted
1:0	RO	0x0	reserved

USBOTG_GAHBCFG

Address: Operational Base + offset (0x0008)

AHB Configuration Register

Bit	Attr	Reset Value	Description
31:23	RO	0x0	reserved

Bit	Attr	Reset Value	Description
22	RW	0x0	NotiAllDmaWrit Notify All Dma Write Transactions This bit is programmed to enable the System DMA Done functionality for all the DMA write Transactions corresponding to the Channel/Endpoint. This bit is valid only when GAHBCFG.RemMemSupp is set to 1. GAHBCFG.NotiAllDmaWrit = 1. HSOTG core asserts int_dma_req for all the DMA write transactions on the AHB interface along with int_dma_done, chep_last_transact and chep_number signal informations. The core waits for sys_dma_done signal for all the DMA write transactions in order to complete the transfer of a particular Channel/Endpoint. GAHBCFG.NotiAllDmaWrit = 0. HSOTG core asserts int_dma_req signal only for the last transaction of DMA write transfer corresponding to a particular Channel/Endpoint. Similarly, the core waits for sys_dma_done signal only for that transaction of DMA write to complete the transfer of a particular Channel/Endpoint.
21	RW	0x0	RemMemSupp Remote Memory Support This bit is programmed to enable the functionality to wait for the system DMA Done Signal for the DMA Write Transfers. GAHBCFG.RemMemSupp=1. The int_dma_req output signal is asserted when HSOTG DMA starts write transfer to the external memory. When the core is done with the Transfers it asserts int_dma_done signal to flag the completion of DMA writes from HSOTG. The core then waits for sys_dma_done signal from the system to proceed further and complete the Data Transfer corresponding to a particular Channel/Endpoint. GAHBCFG.RemMemSupp=0. The int_dma_req and int_dma_done signals are not asserted and the core proceeds with the assertion of the XferComp interrupt as soon as the DMA write transfer is done at the HSOTG Core Boundary and it does not wait for the sys_dma_done signal to complete the DATA transfers.
20:9	RO	0x0	reserved
8	RW	0x0	PTxFEmpLvl Periodic TxFIFO Empty Level Indicates when the Periodic TxFIFO Empty Interrupt bit in the Core Interrupt register (GINTSTS.PTxFEmp) is triggered. This bit is used only in Slave mode. 0: GINTSTS.PTxFEmp interrupt indicates that the Periodic TxFIFO is half empty 1: GINTSTS.PTxFEmp interrupt indicates that the Periodic TxFIFO is completely empty

Bit	Attr	Reset Value	Description
7	RW	0x0	NPTxFEmpLvl Non-Periodic TxFIFO Empty Level This bit is used only in Slave mode. In host mode and with Shared FIFO with device mode, this bit indicates when the Non-Periodic TxFIFO Empty Interrupt bit in the Core Interrupt register GINTSTS.NPTxFEmp) is triggered. With dedicated FIFO in device mode, this bit indicates when IN endpoint Transmit FIFO empty interrupt (DIEPINTn.TxFEmp) is triggered. Host mode and with Shared FIFO with device mode: 1'b0: GINTSTS.NPTxFEmp interrupt indicates that the Non-Periodic TxFIFO is half empty 1'b1: GINTSTS.NPTxFEmp interrupt indicates that the Non-Periodic TxFIFO is completely empty Dedicated FIFO in device mode: 1'b0: DIEPINTn.TxFEmp interrupt indicates that the IN Endpoint TxFIFO is half empty 1'b1: DIEPINTn.TxFEmp interrupt indicates that the IN Endpoint TxFIFO is completely empty
6	RO	0x0	reserved
5	RW	0x0	DMAEn DMA Enable 0: Core operates in Slave mode 1: Core operates in a DMA mode This bit is always 0 when Slave-Only mode has been selected.
4:1	RW	0x0	HBstLen Burst Length/Type This field is used in both External and Internal DMA modes. In External DMA mode, these bits appear on dma_burst[3:0] ports, External DMA Mode defines the DMA burst length in terms of 32-bit words: 4'b0000: 1 word 4'b0001: 4 words 4'b0010: 8 words 4'b0011: 16 words 4'b0100: 32 words 4'b0101: 64 words 4'b0110: 128 words 4'b0111: 256 words Others: Reserved Internal DMA Mode AHB Master burst type: 4'b0000: Single 4'b0001: INCR 4'b0011: INCR4 4'b0101: INCR8 4'b0111: INCR16 Others: Reserved

Bit	Attr	Reset Value	Description
0	RW	0x0	GlblIntrMsk Global Interrupt Mask The application uses this bit to mask or unmask the interrupt line assertion by itself. Irrespective of this bit's setting, the interrupt status registers are updated by the core. 1'b0: Mask the interrupt assertion to the application. 1'b1: Unmask the interrupt assertion to the application.

USBOTG_GUSBCFG
Address: Operational Base + offset (0x000c)
USB Configuration Register

Bit	Attr	Reset Value	Description
31	RW	0x0	CorruptTxpacket Corrupt Tx packet This bit is for debug purposes only. Never set this bit to 1.
30	RW	0x0	ForceDevMode Force Device Mode Writing a 1 to this bit forces the core to device mode irrespective of utmiotg_iddig input pin. 1'b0: Normal Mode 1'b1: Force Device Mode After setting the force bit, the application must wait at least 25 ms before the change to take effect. When the simulation is in scale down mode, waiting for 500 us is sufficient. This bit is valid only when OTG_MODE = 0, 1 or 2. In all other cases, this bit reads 0.
29	RW	0x0	ForceHstMode Force Host Mode Writing a 1 to this bit forces the core to host mode irrespective of utmiotg_iddig input pin. 1'b0: Normal Mode 1'b1: Force Host Mode After setting the force bit, the application must wait at least 25 ms before the change to take effect. When the simulation is in scale down mode, waiting for 500 us is sufficient. This bit is valid only when OTG_MODE =0, 1 or 2. In allother cases, this bit reads 0.
28	RW	0x0	TxEndDelay Tx End Delay Writing a 1 to this bit enables the TxEndDelay timers in the core. 1'b0: Normal mode 1'b1: Introduce Tx end delay timers

Bit	Attr	Reset Value	Description
27	RW	0x0	IC_USBTrafCtl IC_USB TrafficPullRemove Control When this bit is set, pullup/pulldown resistors are detached from the USB during traffic signaling.This bit is valid only when configuration parameter OTG_ENABLE_IC_USB = 1 and register field GUSBCFG.IC_USBCap is set to 1.
26	RW	0x0	IC_USBCap IC_USB-Capable The application uses this bit to control the IC_USB capabilities. 1'b0: IC_USB PHY Interface is not selected. 1'b1: IC_USB PHY Interface is selected. This bit is writable only if OTG_ENABLE_IC_USB=1 and OTG_FSPHY_INTERFACE!=0.The reset value depends on the configuration parameter OTG_SELECT_IC_USB when OTG_ENABLE_IC_USB = 1. In all other cases, this bit is set to 1'b0 and the bit is read only.
25	RW	0x0	ULPIIfDis ULPI Interface Protect Disable Controls circuitry built into the PHY for protecting the ULPI interface when the link tri-states STP and data. Any pull-ups or pull-downs employed by this feature can be disabled. Please refer to the ULPI Specification for more detail. 1'b0: Enables the interface protect circuit 1'b1: Disables the interface protect circuit
24	RW	0x0	IndPassThrough Indicator Pass Through Controls whether the Complement Output is qualified with the Internal VBUS Valid comparator before being used in the VBUS State in the RX CMD. Please refer to the ULPI Specification for more detail. 1'b0: Complement Output signal is qualified with the Internal VBUSValid comparator. 1'b1: Complement Output signal is not qualified with the Internal VBUSValid comparator.
23	RW	0x0	IndComple Indicator Complement Controls the PHY to invert the ExternalVBUSIndicator input signal,generating the Complement Output. Please refer to the ULPI Specification for more detail 1'b0: PHY does not invert ExternalVBUSIndicator signal 1'b1: PHY does invert ExternalVBUSIndicator signal

Bit	Attr	Reset Value	Description
22	RW	0x0	TermSelDLPulse TermSel DLine Pulsing Selection This bit selects utmi_termselect to drive data line pulse during SRP. 1'b0: Data line pulsing using utmi_txvalid (default). 1'b1: Data line pulsing using utmi_termsel.
21	RW	0x0	ULPIExtVBUSIndicator ULPI External VBUS Indicator This bit indicates to the ULPI PHY to use an external VBUS over-current indicator. 1'b0: PHY uses internal VBUS valid comparator. 1'b1: PHY uses external VBUS valid comparator. (Valid only when RTL parameter OTG_HSPHY_INTERFACE = 2 or 3)
20	RW	0x0	ULPIExtVBUSDrv ULPI External VBUS Drive This bit selects between internal or external supply to drive 5V on VBUS,in ULPI PHY. 1'b0: PHY drives VBUS using internal charge pump (default). 1'b1: PHY drives VBUS using external supply. (Valid only when RTL parameter OTG_HSPHY_INTERFACE = 2 or 3)
19	RW	0x0	ULPIClkSusM ULPI Clock SuspendM This bit sets the ClockSuspendM bit in the Interface Control register on the ULPI PHY. This bit applies only in serial or carkit modes. 1'b0: PHY powers down internal clock during suspend. 1'b1: PHY does not power down internal clock. (Valid only when RTL parameter OTG_HSPHY_INTERFACE = 2 or 3)
18	RW	0x0	ULPIAutoRes ULPI Auto Resume This bit sets the AutoResume bit in the Interface Control register on the ULPI PHY. 1'b0: PHY does not use AutoResume feature. 1'b1: PHY uses AutoResume feature. (Valid only when RTL parameter OTG_HSPHY_INTERFACE = 2 or 3)

Bit	Attr	Reset Value	Description
17	RW	0x0	ULPIFsLs ULPI FS/LS Select The application uses this bit to select the FS/LS serial interface for the ULPI PHY. This bit is valid only when the FS serial transceiver is selected on the ULPI PHY. 1'b0: ULPI interface 1'b1: ULPI FS/LS serial interface (Valid only when RTL parameters OTG_HSPHY_INTERFACE = 2 or 3 and OTG_FSPHY_INTERFACE = 1, 2, or 3)
16	RW	0x0	OtgI2CSel UTMIFS or I2C Interface Select The application uses this bit to select the I2C interface. 1'b0: UTMI USB 1.1 Full-Speed interface for OTG signals 1'b1: I2C interface for OTG signals This bit is writable only if I2C and UTMIFS were specified for Enable I2C Interface (parameter OTG_I2C_INTERFACE = 2). Otherwise, reads return 0.
15	RW	0x0	PhyLPwrClkSel PHY Low-Power Clock Select Selects either 480-MHz or 48-MHz (low-power) PHY mode. In FS and LS modes, the PHY can usually operate on a 48-MHz clock to save power. 1'b0: 480-MHz Internal PLL clock 1'b1: 48-MHz External Clock In 480 MHz mode, the UTMI interface operates at either 60 or 30-MHz, depending upon whether 8- or 16-bit data width is selected. In 48-MHz mode, the UTMI interface operates at 48 MHz in FS and LS modes. This bit drives the utmi_fsls_low_power core output signal, and is valid only for UTMI+ PHYs.
14	RO	0x0	reserved
13:10	RW	0x5	USBTrdTim USB Turnaround Time Sets the turnaround time in PHY clocks. Specifies the response time for a MAC request to the Packet FIFO Controller (PFC) to fetch data from the DFIF(SPRAM). This must be programmed to 4'h5: When the MAC interface is 16-bit UTMI+. 4'h9: When the MAC interface is 8-bit UTMI+. Note: The values above are calculated for the minimum AHB frequency of 30MHz. USB turnaround time is critical for certification where long cables and 5-Hubs are used, so if you need the AHB to run at less than 30 MHz, and if USB turnaround time is not critical, these bits can be programmed to a larger value.

Bit	Attr	Reset Value	Description
9	RW	0x0	HNPCap HNP-Capable The application uses this bit to control the DWC_otg core's HNP capabilities. 0: HNP capability is not enabled. 1: HNP capability is enabled. This bit is writable only if an HNP mode was specified for Mode of Operation (parameter OTG_MODE). Otherwise, reads return 0.
8	RW	0x0	SRPCap SRP-Capable The application uses this bit to control the DWC_otg core SRP capabilities. If the core operates as a non-SRP-capable B-device, it cannot request the connected A-device (host) to activate VBUS and start a session. 0: SRP capability is not enabled. 1: SRP capability is enabled. This bit reads return 0.
7	RW	0x0	DDRSel ULPI DDR Select The application uses this bit to select a Single Data Rate (SDR) or Double Data Rate (DDR) or ULPI interface. 0: Single Data Rate ULPI Interface, with 8-bit-wide data bus 1: Double Data Rate ULPI Interface, with 4-bit-wide data bus
6	RW	0x0	PHYSel USB 2.0 High-Speed PHY or USB 1.1 Full-Speed Serial Transceiver The application uses this bit to select either a high-speed UTMI+ or ULPI PHY, or a full-speed transceiver. 0: USB 2.0 high-speed UTMI+ or ULPI PHY 1: USB 1.1 full-speed serial transceiver If a USB 1.1 Full-Speed Serial Transceiver interface was not selected (parameter OTG_FSPHY_INTERFACE = 0), this bit is always 0, with Write Only access. If a high-speed PHY interface was not selected (parameter OTG_HSPHY_INTERFACE = 0), this bit is always 1, with Write Only access.

Bit	Attr	Reset Value	Description
5	RW	0x0	FSIntf Full-Speed Serial Interface Select The application uses this bit to select either a unidirectional or bidirectional USB 1.1 full-speed serial transceiver interface. 0: 6-pin unidirectional full-speed serial interface 1: 3-pin bidirectional full-speed serial interface If a USB 1.1 Full-Speed Serial Transceiver interface was not selected (parameter OTG_FSPHY_INTERFACE = 0), this bit is always 0, with Write Only access. If a USB 1.1 FS interface was selected (parameter OTG_FSPHY_INTERFACE! = 0), then the application can set this bit to select between the 3- and 6-pin interfaces, and access is Read and Write.
4	RW	0x0	ULPI_UTMI_Sel ULPI or UTMI+ Select The application uses this bit to select either a UTMI+ interface or ULPI Interface. 0: UTMI+ Interface 1: ULPI Interface
3	RW	0x0	PHYIf PHY Interface The application uses this bit to configure the core to support a UTMI+ PHY with an 8- or 16-bit interface. When a ULPI PHY is chosen, this must be set to 8-bit mode. 0: 8 bits 1: 16 bits This bit is writable only if UTMI+ and ULPI were selected (parameter OTG_HSPHY_DWIDTH = 3). Otherwise, this bit returns the value for the power-on interface selected during configuration.

Bit	Attr	Reset Value	Description
2:0	RW	0x0	TOutCal HS/FS Timeout Calibration The number of PHY clocks that the application programs in this field is added to the high-speed/full-speed inter-packet timeout duration in the core to account for any additional delays introduced by the PHY. This can be required, because the delay introduced by the PHY in generating the line state condition can vary from one PHY to another. The USB standard timeout value for high-speed operation is 736 to 816 (inclusive) bit times. The USB standard timeout value for full-speed operation is 16 to 18 (inclusive) bit times. The application must program this field based on the speed of enumeration. The number of bit times added per PHY clock are: High-speed operation: One 30-MHz PHY clock = 16 bit times One 60-MHz PHY clock = 8 bit times Full-speed operation: One 30-MHz PHY clock = 0.4 bit times One 60-MHz PHY clock = 0.2 bit times One 48-MHz PHY clock = 0.25 bit times

USBOTG_GRSTCTL

Address: Operational Base + offset (0x0010)

Reset Register

Bit	Attr	Reset Value	Description
31	RO	0x1	AHBIdle AHB Master Idle Indicates that the AHB Master State Machine is in the IDLE condition.
30	RO	0x0	DMAReq DMA Request Signal Indicates that the DMA request is in progress. Used for debug.
29:11	RO	0x0	reserved

Bit	Attr	Reset Value	Description
10:6	RW	0x00	TxFNum TxFIFO Number This is the FIFO number that must be flushed using the TxFIFO Flush bit. This field must not be changed until the core clears the TxFIFO Flush bit. 5'h0: Non-periodic TxFIFO flush in Host mode; Non-periodic TxFIFO flush in device mode when in shared FIFO operation. Tx FIFO 0 flush in device mode when in dedicated FIFO mode. 5'h1: Periodic TxFIFO flush in Host mode: Periodic TxFIFO 1 flush in Device mode when in shared FIFO operation; TXFIFO 1 flush in device mode when in dedicated FIFO mode. 5'h2: Periodic TxFIFO 2 flush in Device mode when in shared FIFO operation: TXFIFO 2 flush in device mode when in dedicated FIFO mode. ... 5'hF: Periodic TxFIFO 15 flush in Device mode when in shared FIFO operation: TXFIFO 15 flush in device mode when in dedicated FIFO mode. 5'h10: Flush all the transmit FIFOs in device or host mode.
5	R/W SC	0x0	TxFFlsh TxFIFO Flush This bit selectively flushes a single or all transmit FIFOs, but cannot do so if the core is in the midst of a transaction. The application must write this bit only after checking that the core is neither writing to the TxFIFO nor reading from the TxFIFO. Verify using these registers: Read NAK Effective Interrupt ensures the core is not reading from the FIFO. Write GRSTCTL.AHBIdle ensures the core is not writing anything to the FIFO. Flushing is normally recommended when FIFOs are re-configured or when switching between Shared FIFO and Dedicated Transmit FIFO operation. FIFO flushing is also recommended during device endpoint disable. The application must wait until the core clears this bit before performing any operations. This bit takes eight clocks to clear, using the slower clock of phy_clk or hclk.
4	R/W SC	0x0	RxFFlsh RxFIFO Flush The application can flush the entire RxFIFO using this bit, but must first ensure that the core is not in the middle of a transaction. The application must only write to this bit after checking that the core is neither reading from the RxFIFO nor writing to the RxFIFO. The application must wait until the bit is cleared before performing any other operations. This bit requires 8 clocks (slowest of PHY or AHB clock) to clear.

Bit	Attr	Reset Value	Description
3	R/WSC	0x0	INTknQFlsh IN Token Sequence Learning Queue Flush This bit is valid only if OTG_EN_DED_TX_FIFO = 0. The application writes this bit to flush the IN Token Sequence Learning Queue.
2	W1C	0x0	FrmCntrRst Host Frame Counter Reset The application writes this bit to reset the (micro)frame number counter inside the core. When the (micro)frame counter is reset, the subsequent SOF sent out by the core has a (micro)frame number of 0.
1	R/WSC	0x0	Reset A write to this bit issues a soft reset to the DWC_otg_power_dn module of the core.

Bit	Attr	Reset Value	Description
0	R/W SC	0x0	CSftRst Core Soft Reset Resets the hclk and phy_clock domains as follows: Clears the interrupts and all the CSR registers except the following register bits: CGCCTL.RstPdwnModule PCGCCTL.GateHclk PCGCCTL.PwrClmp PCGCCTL.StopPPhyLPwrClkSelclk GUSBCFG.PhyLPwrClkSel GUSBCFG.DDRSel GUSBCFG.PHYSel GUSBCFG.FSIntf GUSBCFG.ULPI_UTMI_Sel GUSBCFG.PHYIf HCFG.FSLSPclkSel DCFG.DevSpd GGPIO GPWRDN GADPCTL All module state machines (except the AHB Slave Unit) are reset to the IDLE state, and all the transmit FIFOs and the receive FIFO are flushed. Any transactions on the AHB Master are terminated as soon as possible, after gracefully completing the last data phase of an AHB transfer. Any transactions on the USB are terminated immediately. When Hibernation or ADP feature is enabled, the PMU module is not reset by the Core Soft Reset. The application can write to this bit any time it wants to reset the core. This is a self-clearing bit and the core clears this bit after all the necessary logic is reset in the core, which can take several clocks, depending on the current state of the core. Once this bit is cleared software must wait at least 3 PHY clocks before doing any access to the PHY domain (synchronization delay). Software must also must check that bit 31 of this register is 1 (AHB Master is IDLE) before starting any operation. Typically software reset is used during software development and also when you dynamically change the PHY selection bits in the USB configuration registers listed above. When you change the PHY, the corresponding clock for the PHY is selected and used in the PHY domain. Once a new clock is selected, the PHY domain has to be reset for proper operation.

USBOTG_GINTSTS
Address: Operational Base + offset (0x0014)
Interrupt Register

Bit	Attr	Reset Value	Description
31	W1C	0x0	WkUpInt Resume/Remote Wakeup Detected Interrupt Wakeup Interrupt during Suspend(L2) or LPM(L1) state. During Suspend(L2): Device Mode:This interrupt is asserted only when Host Initiated Resume is detected on USB. Host Mode:This interrupt is asserted only when Device Initiated Remote Wakeup is detected on USB. During LPM(L1): Device Mode:This interrupt is asserted for either Host Initiated Resume or Device Initiated Remote Wakeup on USB. Host Mode:This interrupt is asserted for either Host Initiated Resume or Device Initiated Remote Wakeup on USB.
30	W1C	0x0	SessReqInt Session Request/New Session Detected Interrupt In Host mode, this interrupt is asserted when a session request is detected from the device. In Host mode, this interrupt is asserted when a session request is detected from the device. In Device mode, this interrupt is asserted when the utmisrp_bvalid signal goes high.
29	W1C	0x0	DisconnInt Disconnect Detected Interrupt This interrupt is asserted when a device disconnect is detected.
28	W1C	0x0	ConIDStsChng Connector ID Status Change This interrupt is asserted when there is a change in connector ID status.
27	W1C	0x0	LPM_Int LPM Transaction Received Interrupt Device Mode: This interrupt is asserted when the device receives an LPM transaction and responds with a non-ERRORed response. Host Mode: This interrupt is asserted when the device responds to an LPM transaction with a non-ERRORed response or when the host core has completed LPM transactions for the programmed number of times (GLPMCFG.RetryCnt). This field is valid only if the Core LPM Configuration register's LPMCapable (LPMCap) field is set to 1.
26	RO	0x0	PTxFEmp Periodic TxFIFO Empty This interrupt is asserted when the Periodic Transmit FIFO is either half or completely empty and there is space for at least one entry to be written in the Periodic Request Queue. The half or completely empty status is determined by the Periodic TxFIFO Empty Level bit in the Core AHB Configuration register (GAHBCFG.PTxFEmpLvl).

Bit	Attr	Reset Value	Description
25	RO	0x0	HChInt Host Channels Interrupt The core sets this bit to indicate that an interrupt is pending on one of the channels of the core (in Host mode). The application must read the Host All Channels Interrupt (HAINT) register to determine the exact number of the channel on which the interrupt occurred, and then read the corresponding Host Channel-n Interrupt (HCINTn) register to determine the exact cause of the interrupt. The application must clear the appropriate status bit in the HCINTn register to clear this bit.
24	RO	0x0	PrtInt Host Port Interrupt The core sets this bit to indicate a change in port status of one of the DWC_otg core ports in Host mode. The application must read the Host Port Control and Status (HPRT) register to determine the exact event that caused this interrupt. The application must clear the appropriate status bit in the Host Port Control and Status register to clear this bit.
23	RW	0x0	ResetDet Reset Detected Interrupt The core asserts this interrupt in Device mode when it detects a reset on the USB in Partial Power-Down mode when the device is in Suspend. This interrupt is not asserted in Host mode.
22	W1C	0x0	FetSusp Data Fetch Suspended This interrupt is valid only in DMA mode. This interrupt indicates that the core has stopped fetching data for IN endpoints due to the unavailability of TxFIFO space or Request Queue space. This interrupt is used by the application for an endpoint mismatch algorithm. For example, after detecting an endpoint mismatch, the application: Sets a global non-periodic IN NAK handshake, Disables In endpoints, Flushes the FIFO, Determines the token sequence from the IN Token Sequence Learning Queue, Re-enables the endpoints, Clear the global non-periodic IN NAK handshake. If the global non-periodic IN NAK is cleared, the core has not yet fetched data for the IN endpoint, and the IN token is received: the core generates an "IN token received when FIFO empty" interrupt. The OTG then sends the host a NAK response. To avoid this scenario, the application can check the GINTSTS.FetSusp interrupt, which ensures that the FIFO is full before clearing a global NAK handshake. Alternatively, the application can mask the IN token received when FIFO empty interrupt when clearing a global IN NAK handshake.

Bit	Attr	Reset Value	Description
21	W1C	0x0	incomplP Incomplete Periodic Transfer In Host mode, the core sets this interrupt bit when there are incomplete periodic transactions still pending which are scheduled for the current microframe. Incomplete Isochronous OUT Transfer (incompISOOUT) The Device mode, the core sets this interrupt to indicate that there is at least one isochronous OUT endpoint on which the transfer is not completed in the current microframe. This interrupt is asserted along with the End of Periodic Frame Interrupt (EOPF) bit in this register.
20	W1C	0x0	incompISOIN Incomplete Isochronous IN Transfer The core sets this interrupt to indicate that there is at least one isochronous IN endpoint on which the transfer is not completed in the current microframe. This interrupt is asserted along with the End of Periodic Frame Interrupt (EOPF) bit in this register. Note: This interrupt is not asserted in Scatter/Gather DMA mode.
19	RO	0x0	OEPInt OUT Endpoints Interrupt The core sets this bit to indicate that an interrupt is pending on one of the OUT endpoints of the core (in Device mode). The application must read the Device All Endpoints Interrupt (DAINT) register to determine the exact number of the OUT endpoint on which the interrupt occurred, and then read the corresponding Device OUT Endpoint-n Interrupt (DOEPINTn) register to determine the exact cause of the interrupt. The application must clear the appropriate status bit in the corresponding DOEPINTn register to clear this bit.
18	RO	0x0	IEPInt IN Endpoints Interrupt The core sets this bit to indicate that an interrupt is pending on one of the IN endpoints of the core (in Device mode). The application must read the Device All Endpoints Interrupt (DAINT) register to determine the exact number of the IN endpoint on which the interrupt occurred, and then read the corresponding Device IN Endpoint-n Interrupt (DIEPINTn) register to determine the exact cause of the interrupt. The application must clear the appropriate status bit in the corresponding DIEPINTn register to clear this bit.
17	W1C	0x0	EPMis Endpoint Mismatch Interrupt Note: This interrupt is valid only in shared FIFO operation. Indicates that an IN token has been received for a non-periodic endpoint, but the data for another endpoint is present in the top of the Non-periodic Transmit FIFO and the IN endpoint mismatch count programmed by the application has expired.

Bit	Attr	Reset Value	Description
16	W1C	0x0	RstrDoneInt Restore Done Interrupt The core sets this bit to indicate that the restore command after Hibernation was completed by the core. The core continues from Suspended state into the mode dictated by PCGCCTL.RestoreMode field.This bit is valid only when Hibernation feature is enabled.
15	W1C	0x0	EOPF End of Periodic Frame Interrupt Indicates that the period specified in the Periodic Frame Interval field of the Device Configuration register (DCFG.PerFrInt) has been reached in the current microframe.
14	W1C	0x0	ISOOutDrop Isochronous OUT Packet Dropped Interrupt The core sets this bit when it fails to write an isochronous OUT packet into the RxFIFO because the RxFIFO does not have enough space to accommodate a maximum packet size packet for the isochronous OUT endpoint.
13	W1C	0x0	EnumDone Enumeration Done The core sets this bit to indicate that speed enumeration is complete. The application must read the Device Status (DSTS) register to obtain the enumerated speed.
12	W1C	0x0	USBRst USB Reset The core sets this bit to indicate that a reset is detected on the USB.
11	W1C	0x0	USBSusp USB Suspend The core sets this bit to indicate that a suspended was detected on the USB. The core enters the Suspended state when there is no activity on the utmi_linestate signal for an extended period of time.
10	W1C	0x0	ErlySusp Early Suspend The core sets this bit to indicate that an Idle state has been detected on the USB for 3ms.
9	W1C	0x0	I2CINT I2C Interrupt The core sets this interrupt when I2C access is completed on the I2C interface. This field is used only if the I2C interface was enabled. Otherwise, reads return 0.

Bit	Attr	Reset Value	Description
8	W1C	0x0	ULPICKINT ULPI Carkit Interrupt This field is used only if the Carkit interface was enabled. Otherwise, reads return 0. The core sets this interrupt when a ULPI Carkit interrupt is received. The core's PHY sets ULPI Carkit interrupt in UART or Audio mode. I2C Carkit Interrupt (I2CCKINT) This field is used only if the I2C interface was enabled. Otherwise, reads return 0.The core sets this interrupt when a Carkit interrupt is received. The core's PHY sets the I2C Carkit interrupt in Audio mode.
7	RO	0x0	GOUTNakEff Global OUT NAK Effective Indicates that the Set Global OUT NAK bit in the Device Control register DCTL.SGOUTNak), set by the application, has taken effect in the core. This bit can be cleared by writing the Clear Global OUT NAK bit in the Device Control register (DCTL.CGOUTNak).
6	RO	0x0	GINNakEff Global IN Non-Periodic NAK Effective Indicates that the Set Global Non-periodic IN NAK bit in the Device Control register (DCTL.SGNPInNak), set by the application, have taken effect in the core. That is, the core has sampled the Global IN NAK bit set by the application. This bit can be cleared by clearing the Clear Global Nonperiodic IN NAK bit in the Device Control register (DCTL.CGNPInNak). This interrupt does not necessarily mean that a NAK handshake is sent out on the USB. The STALL bit takes precedence over the NAK bit.
5	RO	0x0	NPTxFEmp Non-Periodic TxFIFO Empty This interrupt is valid only when OTG_EN_DED_TX_FIFO = 0. This interrupt is asserted when the Non-periodic TxFIFO is either half or completely empty, and there is space for at least one entry to be written to the Non-periodic Transmit Request Queue. The half or completely empty status is determined by the Non-periodic TxFIFO Empty Level bit in the Core AHB Configuration register (GAHBCFG.NPTxFEmpLvl).
4	RO	0x0	RxFLvl RxFIFO Non-Empty Indicates that there is at least one packet pending to be read from the RxFIFO.

Bit	Attr	Reset Value	Description
3	W1C	0x0	Sof Start of (micro)Frame In Host mode, the core sets this bit to indicate that an SOF (FS), micro-SOF(HS), or Keep-Alive (LS) is transmitted on the USB. The application must write a 1 to this bit to clear the interrupt. In Device mode, in the core sets this bit to indicate that an SOF token has been received on the USB. The application can read the Device Status register to get the current (micro)frame number. This interrupt is seen only when the core is operating at either HS or FS.
2	RO	0x0	OTGInt OTG Interrupt The core sets this bit to indicate an OTG protocol event. The application must read the OTG Interrupt Status (GOTGINT) register to determine the exact event that caused this interrupt. The application must clear the appropriate status bit in the GOTGINT register to clear this bit.
1	W1C	0x0	ModeMis Mode Mismatch Interrupt The core sets this bit when the application is trying to access: A Host mode register, when the core is operating in Device mode; A Device mode register, when the core is operating in Host mode. The register access is completed on the AHB with an OKAY response, but is ignored by the core internally and does not affect the operation of the core.
0	RO	0x0	CurMod Current Mode of Operation Indicates the current mode. 1'b0: Device mode 1'b1: Host mode

USBOTG_GINTMSK

Address: Operational Base + offset (0x0018)

Interrupt Mask Register

Bit	Attr	Reset Value	Description
31	RW	0x0	WkUpIntMsk Resume/Remote Wakeup Detected Interrupt Mask
30	RW	0x0	SessReqIntMsk Session Request/New Session Detected Interrupt Mask
29	RW	0x0	DisconnIntMsk Disconnect Detected Interrupt Mask
28	RW	0x0	ConIDStsChngMsk Connector ID Status Change Mask
27	RW	0x0	LPM_IntMsk LPM Transaction Received Interrupt Mask

Bit	Attr	Reset Value	Description
26	RW	0x0	PTxFEmpMsk Periodic TxFIFO Empty Mask
25	RW	0x0	HChIntMsk Host Channels Interrupt Mask
24	RW	0x0	PrtIntMsk Host Port Interrupt Mask
23	RW	0x0	ResetDetMsk Reset Detected Interrupt Mask
22	RW	0x0	FetSuspMsk Data Fetch Suspended Mask
21	RW	0x0	incomplPMsk_incompISOOUTMsk Incomplete Periodic Transfer Mask(Host only) Incomplete Isochronous OUT Transfer Mask(Device only)
20	RW	0x0	incompISOINMsk Incomplete Isochronous IN Transfer Mask
19	RW	0x0	OEPIntMsk OUT Endpoints Interrupt Mask
18	RW	0x0	IEPIntMsk IN Endpoints Interrupt Mask
17	RW	0x0	EPMisMsk Endpoint Mismatch Interrupt Mask
16	RW	0x0	RstrDoneIntMsk Restore Done Interrupt Mask This field is valid only when Hibernation feature is enabled.
15	RW	0x0	EOPFMsk End of Periodic Frame Interrupt Mask
14	RW	0x0	ISOOutDropMsk Isochronous OUT Packet Dropped Interrupt Mask
13	RW	0x0	EnumDoneMsk Enumeration Done Mask
12	RW	0x0	USBRstMsk USB Reset Mask
11	RW	0x0	USBSuspMsk USB Suspend Mask
10	RW	0x0	ErlySuspMsk Early Suspend Mask
9	RW	0x0	I2CIntMsk I2C Interrupt Mask
8	RW	0x0	ULPICKINTMsk_I2CCKINTMsk ULPI Carkit Interrupt Mask (ULPICKINTMsk) I2C Carkit Interrupt Mask (I2CCKINTMsk)
7	RW	0x0	GOUTNakEffMsk Global OUT NAK Effective Mask
6	RW	0x0	GINNakEffMsk Global Non-periodic IN NAK Effective Mask

Bit	Attr	Reset Value	Description
5	RW	0x0	NPTxFEmpMsk Non-periodic TxFIFO Empty Mask
4	RW	0x0	RxFLvlMsk Receive FIFO Non-Empty Mask
3	RW	0x0	SofMsk Start of (micro)Frame Mask
2	RW	0x0	OTGIntMsk OTG Interrupt Mask
1	RW	0x0	ModeMisMsk Mode Mismatch Interrupt Mask
0	RO	0x0	reserved

USBOTG_GRXSTSR

Address: Operational Base + offset (0x001c)

Receive Status Debug Read Register

Bit	Attr	Reset Value	Description
31:25	RO	0x0	reserved
24:21	RO	0x0	FN Frame Number (Device Only)This is the least significant 4 bits of the (micro)frame number in which the packet is received on the USB. This field is supported only when isochronous OUT endpoints are supported.
20:17	RO	0x0	PktSts Packet Status Indicates the status of the received packet(Host Only) 4'b0010: IN data packet received 4'b0011: IN transfer completed (triggers an interrupt) 4'b0101: Data toggle error (triggers an interrupt) 4'b0111: Channel halted (triggers an interrupt) Others: Reserved Indicates the status of the received packet(Device only) 4'b0001: Global OUT NAK (triggers an interrupt) 4'b0010: OUT data packet received 4'b0011: OUT transfer completed (triggers an interrupt) 4'b0100: SETUP transaction completed (triggers an interrupt) 4'b0110: SETUP data packet received Others: Reserved
16:15	RO	0x0	DPID Data PID Indicates the Data PID of the received packet 2'b00: DATA0 2'b10: DATA1 2'b01: DATA2 2'b11: MDATA

Bit	Attr	Reset Value	Description
14:4	RW	0x000	BCnt Byte Count Indicates the byte count of the received data packet.
3:0	RO	0x0	ChNum_EPNum Channel Number(Host) Endpoint Number(Device) (Host Only) Indicates the channel number to which the current received packet belongs. (Device Only) Indicates the endpoint number to which the current received packet belongs.

USBOTG_GRXSTSP

Address: Operational Base + offset (0x0020)

Receive Status Read and Pop Register

Bit	Attr	Reset Value	Description
31:25	RO	0x0	reserved
24:21	RO	0x0	FN Frame Number (Device Only) This is the least significant 4 bits of the (micro)frame number in which the packet is received on the USB. This field is supported only when isochronous OUT endpoints are supported.
20:17	RO	0x0	PktSts Packet Status Indicates the status of the received packet(Host Only) 4'b0010: IN data packet received 4'b0011: IN transfer completed (triggers an interrupt) 4'b0101: Data toggle error (triggers an interrupt) 4'b0111: Channel halted (triggers an interrupt) Others: Reserved Indicates the status of the received packet(Device only) 4'b0001: Global OUT NAK (triggers an interrupt) 4'b0010: OUT data packet received 4'b0011: OUT transfer completed (triggers an interrupt) 4'b0100: SETUP transaction completed (triggers an interrupt) 4'b0110: SETUP data packet received Others: Reserved
16:15	RO	0x0	DPID Data PID Indicates the Data PID of the received OUT data packet 2'b00: DATA0 2'b10: DATA1 2'b01: DATA2 2'b11: MDATA
14:4	RO	0x000	BCnt Byte Count Indicates the byte count of the received data packet.

Bit	Attr	Reset Value	Description
3:0	RO	0x0	ChNum_EPNum Channel Number(Host) Endpoint Number(Device) (Host Only) Indicates the channel number to which the current received packet belongs. (Device Only) Indicates the endpoint number to which the current received packet belongs.

USBOTG_GRXFSIZ
Address: Operational Base + offset (0x0024)
Receive FIFO Size Register

Bit	Attr	Reset Value	Description
31:16	RO	0x0	reserved
15:0	RW	0x0000	RxFDep RxFIFO Depth This value is in terms of 32-bit words. Minimum value is 16. Maximum value is 32,768. The power-on reset value of this register is specified as the Largest Rx Data FIFO Depth. If Enable Dynamic FIFO Sizing was deselected, these flops are optimized, and reads return the power-on value. If Enable Dynamic FIFO Sizing was selected, you can write a new value in this field. You can write a new value in this field. Programmed values must not exceed the power-on value.

USBOTG_GNPTXFSIZ
Address: Operational Base + offset (0x0028)
Non-Periodic Transmit FIFO Size Register

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	NPTxFDep Non-periodic TxFIFO For host mode, this field is always valid. For Device mode, this field is valid only when OTG_EN_DED_TX_FIFO==0. This value is in terms of 32-bit words. Minimum value is 16. Maximum value is 32,768 This field is determined by Enable Dynamic FIFO Sizing. OTG_DFIFO_DYNAMIC = 0: These flops are optimized, and reads return the power-on value. OTG_DFIFO_DYNAMIC = 1: The application can write a new value in this field. Programmed values must not exceed the power-on value. The power-on reset value of this field is specified by OTG_EN_DED_TX_FIFO: OTG_EN_DED_TX_FIFO = 0:The reset value is the Largest Non-periodic Tx Data FIFO Depth parameter, OTG_TX_NPERIO_DFIFO_DEPTH. OTG_EN_DED_TX_FIFO = 1:The reset value is parameter OTG_TX_HNPERIO_DFIFO_DEPTH.

Bit	Attr	Reset Value	Description
15:0	RW	0x0000	NPTxFStAddr Non-periodic Transmit RAM For host mode, this field is always valid. This field contains the memory start address for Non-periodic Transmit FIFO RAM. This field is determined by Enable Dynamic FIFO Sizing(OTG_DFIFO_DYNAMIC): OTG_DFIFO_DYNAMIC = 0: These flops are optimized, and reads return the power-on value. OTG_DFIFO_DYNAMIC = 1: The application can write a new value in this field. Programmed values must not exceed the power-on value. The power-on reset value of this field is specified by Largest Rx Data FIFO Depth (parameter OTG_RX_DFIFO_DEPTH).

USBOTG_GNPTXSTS

Address: Operational Base + offset (0x002c)

Non-Periodic Transmit FIFO/Queue Status Register

Bit	Attr	Reset Value	Description
31	RO	0x0	reserved
30:24	RO	0x00	NPTxQTop Top of the Non-periodic Transmit Request Queue Entry in the Non-periodic Tx Request Queue that is currently being processed by the MAC. Bits [30:27]: Channel/endpoint number Bits [26:25]: 2'b00: IN/OUT token 2'b01: Zero-length transmit packet (device IN/host OUT) 2'b10: PING/CSPLIT token 2'b11: Channel halt command Bit [24]: Terminate (last entry for selected channel/endpoint)
23:16	RO	0x00	NPTxQSpcAvail Non-periodic Transmit Request Queue Space Available Indicates the amount of free space available in the Non-periodic Transmit Request Queue. This queue holds both IN and OUT requests in Host mode. Device mode has only IN requests. 8'h0: Non-periodic Transmit Request Queue is full 8'h1: 1 location available 8'h2: 2 locations available n: n locations available (0 <=n <= 8) Others: Reserved

Bit	Attr	Reset Value	Description
15:0	RO	0x0000	NPTxFSpcAvail Non-periodic TxFIFO Space Avail Indicates the amount of free space available in the Non-periodic TxFIFO. Values are in terms of 32-bit words. 16'h0: Non-periodic TxFIFO is full 16'h1: 1 word available 16'h2: 2 words available 16'hn: n words available (where 0 <=n <=32,768) 16'h8000: 32,768 words available Others: Reserved

USBOTG_GI2CCTL
Address: Operational Base + offset (0x0030)
I2C Address Register

Bit	Attr	Reset Value	Description
31	R/WSC	0x0	BsyDne I2C Busy/Done The application sets this bit to 1'b1 to start a request on the I2C interface. When the transfer is complete, the core de-asserts this bit to 1'b0. As long as the bit is set, indicating that the I2C interface is busy, the application cannot start another request on the interface.
30	RW	0x0	RW Read/Write Indicator Indicates whether a read or write register transfer must be performed on the interface. Read/write bursting is not supported for registers. 1'b1: Read 1'b0: Write
29	RO	0x0	reserved
28	RW	0x1	I2CDatSe0 I2C DatSe0 USB Mode Selects the FS interface USB mode. 1'b1: VP_VM USB mode 1'b0: DAT_SE0 USB mode
27:26	RW	0x0	I2CDevAdr I2C Device Address Selects the address of the I2C Slave on the USB 1.1 full-speed serial transceiver that the core uses for OTG signaling. 2'b00: 7'h2C 2'b01: 7'h2D 2'b10: 7'h2E 2'b11: 7'h2F

Bit	Attr	Reset Value	Description
25	RW	0x0	I2CSuspCtl I2C Suspend Control Selects how Suspend is connected to a full-speed transceiver in I2C mode. 1'b0: Use the dedicated utmi_suspend_n pin 1'b1: Use an I2C write to program the Suspend bit in the PHY register
24	RO	0x1	Ack I2C ACK Indicates whether an ACK response was received from the I2C Slave. This bit is valid when BsyDne is cleared by the core, after application has initiated an I2C access. 1'b0: NAK 1'b1: ACK
23	RW	0x0	I2CEn I2C Enable Enables the I2C Master to initiate I2C transactions on the I2C interface
22:16	RW	0x00	Addr I2C Address This is the 7-bit I2C device address used by software to access any external I2C Slave, including the I2C Slave on a USB 1.1 OTG full-speed serial transceiver. Software can change this address to access different I2C Slaves.
15:8	RW	0x00	RegAddr I2C Register Addr This field programs the address of the register to be read from or written to.
7:0	RW	0x00	RWData I2C Read/Write Data After a register read operation, this field holds the read data for the application. During a write operation, the application can use this register to program the write data to be written to a register. During writes, this field holds the write data.

USBOTG_GPVNDCTL

Address: Operational Base + offset (0x0034)

PHY Vendor Control Register

Bit	Attr	Reset Value	Description
31	R/WSC	0x0	DisUlpiDrvr Disable ULPI Drivers This field reads return 0.
30:28	RO	0x0	reserved

Bit	Attr	Reset Value	Description
27	R/W SC	0x0	VStsDone VStatus Done The core sets this bit when the vendor control access is done. This bit is cleared by the core when the application sets the New Register Request bit (bit 25).
26	RO	0x0	VStsBsy VStatus Busy The core sets this bit when the vendor control access is in progress and clears this bit when done.
25	R/W SC	0x0	NewRegReq New Register Request The application sets this bit for a new vendor control access.
24:23	RO	0x0	reserved
22	RW	0x0	RegWr Register Write Set this bit for register writes, and clear it for register reads.
21:16	RW	0x00	RegAddr Register Address The 6-bit PHY register address for immediate PHY Register Set access. Set to 6'h2F for Extended PHY Register Set access.
15:8	RW	0x00	VCtrl UTMI+ Vendor Control Register Address The 4-bit register address a vendor defined 4-bit parallel output bus. Bits 11:8 of this field are placed on utmi_vcontrol[3:0]. ULPI Extended Register Address (ExtRegAddr) The 6-bit PHY extended register address.
7:0	RW	0x00	RegData Register Data Contains the write data for register write. Read data for register read, valid when VStatus Done is set.

USBOTG_GGPIO

Address: Operational Base + offset (0x0038)

General Purpost Input/Output Register

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	GPO General Purpose Output This field is driven as an output from the core, gp_o[15:0]. The application can program this field to determine the corresponding value on the gp_o[15:0] output.
15:0	RO	0x0000	GPI General Purpose Input This field's read value reflects the gp_i[15:0] core input value.

USBOTG_GUID

Address: Operational Base + offset (0x003c)
User ID Register

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	UserID Application-programmable ID field.

USBOTG_GSNPSID
Address: Operational Base + offset (0x0040)
Core ID Register

Bit	Attr	Reset Value	Description
31:0	RO	0x00004f54	CoreID Release number of the core being used

USBOTG_GHWCFG1
Address: Operational Base + offset (0x0044)
User HW Config1 Register

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	epdir Endpoint Direction This 32-bit field uses two bits per endpoint to determine the endpoint direction. Endpoint Bits [31:30]: Endpoint 15 direction Bits [29:28]: Endpoint 14 direction ... Bits [3:2]: Endpoint 1 direction Bits[1:0]: Endpoint 0 direction (always BIDIR) Direction 2'b00: BIDIR (IN and OUT) endpoint 2'b01: IN endpoint 2'b10: OUT endpoint 2'b11: Reserved

USBOTG_GHWCFG2
Address: Operational Base + offset (0x0048)
User HW Config2 Register

Bit	Attr	Reset Value	Description
31	RO	0x0	OTG_ENABLE_IC_USB IC_USB mode specified for mode of operation (parameter OTG_ENABLE_IC_USB). To choose IC_USB_MODE, both OTG_FSPHY_INTERFACE and OTG_ENABLE_IC_USB must be 1.
30:26	RO	0x00	TknQDepth Device Mode IN Token Sequence Learning Queue Depth Range: 0-30

Bit	Attr	Reset Value	Description
25:24	RO	0x0	PTxQDepth Host Mode Periodic Request Queue Depth 2'b00: 2 2'b01: 4 2'b10: 8 Others: Reserved
23:22	RO	0x0	NPTxQDepth Non-periodic Request Queue Depth 2'b00: 2 2'b01: 4 2'b10: 8 Others: Reserved
21	RO	0x0	reserved
20	RO	0x0	MultiProcIntrpt Multi Processor Interrupt Enabled 1'b0: No 1'b1: Yes
19	RO	0x0	DynFifoSizing Dynamic FIFO Sizing Enabled 1'b0: No 1'b1: Yes
18	RO	0x0	PerioSupport Periodic OUT Channels Supported in Host Mode 1'b0: No 1'b1: Yes
17:14	RO	0x0	NumHstChnl Number of Host Channels Indicates the number of host channels supported by the core in Host mode. The range of this field is 0-15: 0 specifies 1 channel, 15 specifies 16 channels.
13:10	RO	0x0	NumDevEps Number of Device Endpoints Indicates the number of device endpoints supported by the core in Device mode in addition to control endpoint 0. The range of this field is 1-15.
9:8	RO	0x0	FSPhyType Full-Speed PHY Interface Type 2'b00: Full-speed interface not supported 2'b01: Dedicated full-speed interface 2'b10: FS pins shared with UTMI+ pins 2'b11: FS pins shared with ULPI pins

Bit	Attr	Reset Value	Description
7:6	RO	0x0	HSPhyType High-Speed PHY Interface Type 2'b00: High-Speed interface not supported 2'b01: UTMI+ 2'b10: ULPI 2'b11: UTMI+ and ULPI
5	RO	0x0	SingPnt Point-to-Point 1'b0: Multi-point application (hub and split support) 1'b1: Single-point application (no hub and no split support)
4:3	RO	0x0	OtgArch Architecture 2'b00: Slave-Only 2'b01: External DMA 2'b10: Internal DMA Others: Reserved
2:0	RO	0x0	OtgMode Mode of Operation 3'b000: HNP- and SRP-Capable OTG (Host and Device) 3'b001: SRP-Capable OTG (Host and Device) 3'b010: Non-HNP and Non-SRP Capable OTG (Host and Device) 3'b011: SRP-Capable Device 3'b100: Non-OTG Device 3'b101: SRP-Capable Host 3'b110: Non-OTG Host Others: Reserved

USBOTG_GHWCFG3
Address: Operational Base + offset (0x004c)
User HW Config3 Register

Bit	Attr	Reset Value	Description
31:16	RO	0x0000	DfifoDepth DFIFO Depth This value is in terms of 32-bit words. Minimum value is 32 Maximum value is 32,768
15	RO	0x0	OTG_ENABLE_LPM LPM mode specified for Mode of Operation (parameter OTG_ENABLE_LPM).
14	RO	0x0	OTG_BC_SUPPORT This bit indicates the HS OTG controller support for Battery Charger. 0 : No Battery Charger Support 1 : Battery Charger support present.

Bit	Attr	Reset Value	Description
13	RO	0x0	OTG_ENABLE_HSIC HSIC mode specified for Mode of Operation (parameter OTG_ENABLE_HSIC). Value Range: 0-1 1: HSIC-capable with shared UTMI PHY interface 0: Non-HSIC-capable
12	RO	0x0	OTG_ADP_SUPPORT This bit indicates whether ADP logic is present within or external to the HS OTG controller 0: No ADP logic present with HSOTG controller 1: ADP logic is present along with HSOTG controller.
11	RO	0x0	RstType Reset Style for Clocked always Blocks in RTL 1'b0: Asynchronous reset is used in the core 1'b1: Synchronous reset is used in the core
10	RO	0x0	OptFeature Optional Features Removed Indicates whether the User ID register, GPIO interface ports, and SOF toggle and counter ports were removed for gate count optimization by enabling Remove Optional Features? 1'b0: No 1'b1: Yes
9	RO	0x0	VndctlSupt Vendor Control Interface Support 1'b0: Vendor Control Interface is not available on the core. 1'b1: Vendor Control Interface is available.
8	RO	0x0	I2CIntSel I2C Selection 1'b0: I2C Interface is not available on the core. 1'b1: I2C Interface is available on the core.
7	RO	0x0	OtgEn OTG Function Enabled The application uses this bit to indicate the DWC_otg core's OTG capabilities. 1'b0: Not OTG capable 1'b1: OTG Capable

Bit	Attr	Reset Value	Description
6:4	RO	0x0	PktSizeWidth Width of Packet Size Counters 3'b000: 4 bits 3'b001: 5 bits 3'b010: 6 bits 3'b011: 7 bits 3'b100: 8 bits 3'b101: 9 bits 3'b110: 10 bits Others: Reserved
3:0	RO	0x0	XferSizeWidth Width of Transfer Size Counters 4'b0000: 11 bits 4'b0001: 12 bits ... 4'b1000: 19 bits Others: Reserved

USBOTG_GHWCFG4
Address: Operational Base + offset (0x0050)
User HW Config4 Register

Bit	Attr	Reset Value	Description
31	RO	0x0	SGDMA Scatter/Gather DMA 1'b1: Dynamic configuration
30	RO	0x0	SGDMACon Scatter/Gather DMA configuration 1'b0: Non-Scatter/Gather DMA configuration 1'b1: Scatter/Gather DMA configuration
29:26	RO	0x0	INEps Number of Device Mode IN Endpoints Including Control Endpoint Range 0 -15 0:1 IN Endpoint 1:2 IN Endpoints .... 15:16 IN Endpoints
25	RW	0x0	DedFifoMode Enable Dedicated Transmit FIFO for device IN Endpoints 1'b0: Dedicated Transmit FIFO Operation not enabled. 1'b1: Dedicated Transmit FIFO Operation enabled.
24	RW	0x0	SessEndFltr session_end Filter Enabled 1'b0: No filter 1'b1: Filter

Bit	Attr	Reset Value	Description
23	RW	0x0	BValidFltr "b_valid" Filter Enabled 1'b0: No filter 1'b1: Filter
22	RO	0x0	AValidFltr "a_valid" Filter Enabled 1'b0: No filter 1'b1: Filter
21	RO	0x0	VBUSValidFltr "VBUS_valid" Filter Enabled 1'b0: No filter 1'b1: Filter
20	RO	0x0	IddgFltr "iddig" Filter Enable 1'b0: No filter 1'b1: Filter
19:16	RO	0x0	NumCtlEps Number of Device Mode Control Endpoints in Addition to Endpoin Range: 0-15
15:14	RO	0x0	PhyDataWidth UTMI+ PHY/ULPI-to-Internal UTMI+ Wrapper Data Width When a ULPI PHY is used, an internal wrapper converts ULPI to UTMI+. 2'b00: 8 bits 2'b01: 16 bits 2'b10: 8/16 bits, software selectable Others: Reserved
13:7	RO	0x0	reserved
6	RO	0x0	EnHiber Enable Hibernation 1'b0: Hibernation feature not enabled 1'b1: Hibernation feature enabled
5	RO	0x0	AhbFreq Minimum AHB Frequency Less Than 60 MHz 1'b0: No 1'b1: Yes
4	RO	0x0	EnParPwrDown Enable Partial Power Down 1'b0: Partial Power Down Not Enabled 1'b1: Partial Power Down Enabled
3:0	RO	0x0	NumDevPerioEps Number of Device Mode Periodic IN Endpoints Range: 0-15

USBOTG_GLPMCFG

Address: Operational Base + offset (0x0054)
Core LPM Configuration Register

Bit	Attr	Reset Value	Description
31	RW	0x0	InvSelHsic HSIC-Invert Select HSIC The application uses this bit to control the DWC_otg core HSIC enable/disable. This bit overrides and functionally inverts the if_sel_hsic input port signal. If the core operates as non-HSIC-capable, it can only connect to non-HSIC-capable PHYs. If the core operates as HSIC-capable, it can only connect to HSICcapable PHYs. If the if_sel_hsic input signal is1: 1'b1: HSIC capability is not enabled. 1'b0: HSIC capability is enabled, If InvSelHsic = 1'b0: HSIC capability is enabled. If the if_sel_hsic input signal is 0: 1'b1: HSIC capability is enabled, 1'b0: HSIC capability is not enabled. This bit is writable only if an HSIC mode was specified for Mode of Operation (parameter OTG_ENABLE_HSIC). This bit is valid only if OTG_ENABLE_HSIC is enabled.
30	RW	0x0	HSICCon HSIC-Connect The application must use this bit to initiate the HSIC Attach sequence. Host Mode: Once this bit is set, the host core configures to drive the HSIC Idle state (STROBE = 1 & DATA = 0) on the bus. It then waits for the device to initiate the Connect sequence. Device Mode: Once this bit is set, the device core waits for the HSIC Idle line state on the bus. Upon receiving the Idle line state, it initiates the HSIC Connect sequence. This bit is valid only if OTG_ENABLE_HSIC is 1, if_sel_hsic = 1 and InvSelHSIC is 0. Otherwise, it is read-only.
29:28	RO	0x0	reserved
27:25	RO	0x0	LPM_RetryCnt_Sts LPM Retry Count Status Number of LPM host retries remaining to be transmitted for the current LPM sequence.
24	RW	0x0	SndLPM Send LPM Transaction Host Mode: When the application software sets this bit, an LPM transaction containing two tokens, EXT and LPM, is sent. The hardware clears this bit once a valid response (STALL, NYET, or ACK) is received from the device or the core has finished transmitting the programmed number of LPM retries. Note: This bit must only be set when the host is connected to a local port.

Bit	Attr	Reset Value	Description
23:21	R/WSC	0x0	LPM_Retry_Cnt LPM Retry Count When the device gives an ERROR response, this is the number of additional LPM retries that the host performs until a valid device response (STALL, NYET, or ACK) is received.
20:17	RW	0x0	LPM_Chnl_Indx LPM Channel Index The channel number on which the LPM transaction must be applied while sending an LPM transaction to the local device. Based on the LPM channel index, the core automatically inserts the device address and endpoint number programmed in the corresponding channel into the LPM transaction.
16	RO	0x0	L1ResumeOK Sleep State Resume OK Indicates that the application or host can start a resume from the Sleep state. This bit is valid in the LPM Sleep (L1) state. It is set in Sleep mode after a delay of 50 us (TL1Residency). The bit is reset when SlpSts is 0 1'b1: The application/core can start resume from the Sleep state 1'b0: The application/core cannot start resume from the Sleep state

Bit	Attr	Reset Value	Description
15	RO	0x0	SlpSts Port Sleep Status Device Mode: This bit is set as long as a Sleep condition is present on the USB bus. The core enters the Sleep state when an ACK response is sent to an LPM transaction and the timer TL1TokenRetry has expired. To stop the PHY clock, the application must set the Port Clock Stop bit, which asserts the PHY Suspend input pin. The application must rely on SlpSts and not ACK in CoreL1Res to confirm transition into sleep. The core comes out of sleep: When there is any activity on the USB line_state, When the application writes to the Remote Wakeup Signaling bit in the Device Control register (DCTL.RmtWkUpSig) or when the application resets or soft-disconnects the device. Host Mode: The host transitions to the Sleep (L1) state as a side-effect of a successful LPM transaction by the core to the local port with an ACK response from the device. The read value of this bit reflects the port's current sleep status. The core clears this bit after: The core detects a remote L1 Wakeup signal, The application sets the Port Reset bit or the Port L1Resume bit in the HPRT register or The application sets the L1Resume/ Remote Wakeup Detected Interrupt bit or Disconnect Detected Interrupt bit in the Core Interrupt register (GINTSTS.L1WkUpInt or GINTSTS.DisconnInt, respectively). Values: 1'b0: Core not in L1 1'b1: Core in L1
14:13	RO	0x0	CoreL1Res LPM Response Device Mode: The core's response to the received LPM transaction is reflected in these two bits. Host Mode: The handshake response received from the local device for LPM transaction. 11: ACK 10: NYET 01: STALL 00: ERROR (No handshake response)

Bit	Attr	Reset Value	Description
12:8	RW	0x00	HIRD_Thres HIRD Threshold Device Mode: The core asserts L1SuspendM to put the PHY into Deep Low-Power mode in L1 when the HIRD value is greater than or equal to the value defined in this field (GLPMCFG.HIRD_Thres[3:0]), and HIRD_Thres[4] is set to 1'b1. Host Mode: The core asserts L1SuspendM to put the PHY into Deep Low-Power mode in L1 when HIRD_Thres[4] is set to 1'b1. HIRD_Thres[3:0] specifies the time for which resume signaling is to be reflected by the host TL1HubDrvResume2) on the USB when it detects device-initiated resume. HIRD_Thres must not be programmed with a value greater than 4'b1100 in Host mode, because this exceeds maximum TL1HubDrvResume2. Sl. No HIRD_Thres[3:0] Host mode resume time(us) 1 4'b0000 60 2 4'b0001 135 3 4'b0010 210 4 4'b0011 285 5 4'b0100 360 6 4'b0101 435 7 4'b0110 510 8 4'b0111 585 9 4'b1000 660 10 4'b1001 735 11 4'b1010 810 12 4'b1011 885 13 4'b1100 960 14 4'b1101 invalid 15 4'b1110 invalid 16 4'b1111 invalid

Bit	Attr	Reset Value	Description
7	RW	0x0	EnblSlpM Enable utmi_sleep_n For ULPI interface: The application uses this bit to write to the function control [7] in the L1 state, to enable the PHY to go into Low Power mode. For the host, this bit is valid only in Local Device mode. 1'b0: Writes to the ULPI Function Control Bit[7] are disabled. 1'b1: The core is enabled to write to the ULPI Function Control Bit[7], which enables the PHY to enter Low-Power mode. Note: When a ULPI interface is configured, enabling this bit results in a write to Bit 7 of the ULPI Function Control register. The ULPI PHY supports writing to this bit, and in the L1 state asserts SleepM when utmi_l1_suspend_n cannot be asserted. When a ULPI interface is configured, this bit must always be set if you are using the ULPI PHY. Note: For ULPI interface, do not clear this bit during the resume. For all other interfaces: The application uses this bit to control utmi_sleep_n assertion to the PHY in the L1 state. For the host, this bit is valid only in Local Device mode. 1'b0: utmi_sleep_n assertion from the core is not transferred to the external PHY. 1'b1: utmi_sleep_n assertion from the core is transferred to the external PHY when utmi_l1_suspend_n cannot be asserted.
6	RW	0x0	bRemoteWake RemoteWakeEnable Host Mode: The remote wakeup value to be sent in the LPM transaction's wIndex field. Device Mode: This field is updated with the received bRemoteWake LPM token's bmAttribute when an ACK/NYET/STALL response is sent to an LPM transaction.

Bit	Attr	Reset Value	Description
5:2	RW	0x0	HIRD Host-Initiated Resume Duration Host Mode: The value of HIRD to be sent in an LPM transaction. This value is also used to initiate resume for a durationL1HubDrvResume1 for host initiated resume. Device Mode: This field is updated with the Received LPM Token HIRD bmAttribute when an ACK/NYET/STALL response is sent to an LPM transaction Sl. No HIRD[3:0] THIRD (us) 1 4'b0000 50 2 4'b0001 125 3 4'b0010 200 4 4'b0011 275 5 4'b0100 350 6 4'b0101 425 7 4'b0110 500 8 4'b0111 575 9 4'b1000 650 10 4'b1001 725 11 4'b1010 800 12 4'b1011 875 13 4'b1100 950 14 4'b1101 1025 15 4'b1110 1100 16 4'b1111 1175
1	RW	0x0	AppL1Res LPM response programmed by application Handshake response to LPM token pre-programmed by device application software. The response depends on GLPMCFG.LPMCap. If GLPMCFG.LPMCap is 1'b0, the core always responds with a NYET. If GLPMCFG.LPMCap is 1'b1, the core responds as follows: 1: ACK. Even though an ACK is pre-programmed, the core responds with an ACK only on a successful LPM transaction. The LPM transaction is successful if: There are no PID/CRC5 errors in both the EXT token and the LPM token (else ERROR); A valid bLinkState = 0001B (L1) is received in the LPM transaction (else STALL); No data is pending in the Transmit queue (else NYET) 0: NYET. The pre-programmed software bit is overridden for response to LPM token when:(1)The received bLinkState is not L1 (STALL response); (2)An error is detected in either of the LPM token packets due to corruption (ERROR response).

Bit	Attr	Reset Value	Description
0	RW	0x0	LPMCap LPM-Capable The application uses this bit to control the DWC_otg core LPM capabilities. If the core operates as a non-LPM-capable host, it cannot request the connected device/hub to activate LPM mode. If the core operates as a non-LPM-capable device, it cannot respond to any LPM transactions. 1'b0: LPM capability is not enabled. 1'b1: LPM capability is enabled. This bit is writable only if an LPM mode was specified for Mode of Operation (parameter OTG_ENABLE_LPM). Otherwise, reads return 0.

USBOTG_GPWRDN

Address: Operational Base + offset (0x0058)

Global Power Down Register

Bit	Attr	Reset Value	Description
31:29	RO	0x0	reserved
28:24	RO	0x00	MultValIdBC Multi Valued ID pin Battery Charger ACA inputs in the following order: Bit 26 - rid_float. Bit 25 - rid_gnd Bit 24 - rid_a Bit 23 - rid_b Bit 22 - rid_c These bits are present only if OTG_BC_SUPPORT = 1. Otherwise, these bits are reserved and will read 5'h0.
23	W1C	0x0	ADPInt ADP Interupt This bit is set whenever there is a ADP event.
22	RO	0x0	BSessVld B Session Valid This field reflects the B session valid status signal from the PHY. 1'b0: B-Valid is 0. 1'b1: B-Valid is 1. This bit is valid only when GPWRDN.PMUActv is 1.

Bit	Attr	Reset Value	Description
21	RO	0x0	IDDIG This bit indicates the status of the signal IDDIG. The application must read this bit after receiving GPWRDN.StsChngInt and decode based on the previous value stored by the application. Indicates the current mode. 1'b1: Device mode 1'b0: Host mode This bit is valid only when GPWRDN.PMUActv is 1.
20:19	RO	0x0	LineState This field indicates the current linestate on USB as seen by the PMU module. 2'b00: DM = 0, DP = 0. 2'b01: DM = 0, DP = 1. 2'b10: DM = 1, DP = 0. 2'b11: Not-defined. This bit is valid only when GPWRDN.PMUActv is 1.
18	RW	0x0	StsChngIntMsk Mask For StsChng Interrupt
17	W1C	0x0	StsChngInt This field indicates a status change in either the IDDIG or BSessVld signal. 1'b0: No Status change 1'b1: status change detected After receiving this interrupt the application should read the GPWRDN register and interpret the change in IDDIG or BSesVld with respect to the previous value stored by the application.
16	RW	0x0	SRPDetectMsk Mask For SRPDetect Interrupt
15	W1C	0x0	SRPDetect This field indicates that SRP has been detected by the PMU.This field generates an interrupt. After detecting SRP during hibernation the application should not restore the core. The application should get into the initialization process. 1'b0: SRP not detected 1'b1: SRP detected
14	RW	0x0	ConnDetMsk Mask for ConnectDet interrupt This bit is valid only when OTG_EN_PWROPT = 2.
13	W1C	0x0	ConnectDet This field indicates that a new connect has been detected 1'b0: Connect not detected 1'b1: Connect detected This bit is valid only when OTG_EN_PWROPT = 2.

Bit	Attr	Reset Value	Description
12	RW	0x0	DisconnectDetectMsk Mask For DisconnectDetect Interrupt This bit is valid only when OTG_EN_PWROPT = 2.
11	W1C	0x0	DisconnectDetect This field indicates that Disconnect has been detected by the PMU. This field generates an interrupt. After detecting disconnect during hibernation the application must not restore the core, but instead start the initialization process. 1'b0: Disconnect not detected 1'b1: Disconnect detected This bit is valid only when OTG_EN_PWROPT = 2.
10	RW	0x0	ResetDetMsk Mask For ResetDetected interrupt.This bit is valid only when OTG_EN_PWROPT = 2.
9	W1C	0x0	ResetDetected This field indicates that Reset has been detected by the PMU module. This field generates an interrupt. 1'b0: Reset Not Detected 1'b1: Reset Detected This bit is valid only when OTG_EN_PWROPT = 2.
8	RW	0x0	LineStageChangeMsk Mask For LineStateChange interrupt This bit is valid only when OTG_EN_PWROPT = 2.
7	W1C	0x0	LnStsChng Line State Change This interrupt is asserted when there is a Linestate Change detected by the PMU. The application should read GPWRDN.Linestate to determine the current linestate on USB. 1'b0: No LineState change on USB 1'b1: LineState change on USB This bit is valid only when GPWRDN.PMUActv is 1.This bit is valid only when OTG_EN_PWROPT = 2.
6	RW	0x0	DisableVBUS The application should program this bit if HPRT0.PrtPwr was programmed to 0 before entering Hibernation. This is to indicate PMU whether session was ended before entering Hibernation. 1'b0: HPRT0.PrtPwr was not programmed to 0. 1'b1: HPRT0.PrtPwr was programmed to 0.

Bit	Attr	Reset Value	Description
5	RW	0x0	PwrDnSwtch Power Down Switch This bit indicates to the DWC_otg core VDD switch is in ON/OFF state 1'b0: DWC_otg is in ON state 1'b1: DWC_otg is in OFF state *Note: This bit must not be written to during normal mode of operation.*
4	RW	0x0	PwrDnRst_n Power Down ResetN The application must program this bit to reset the DWC OTG core during the Hibernation exit process or during ADP when powering up the core (in case the DWC OTG core was powered off during ADP process). 1'b1: DWC_otg is in normal operation 1'b0: reset DWC_otg *Note: This bit must not be written to during normal mode of operation.*
3	RW	0x0	PwrDnClmp Power Down Clamp The application must program this bit to enable or disable the clamps to all the outputs of the DWC OTG core module to prevent the corruption of other active logic. 1'b0: Disable PMU power clamp 1'b1: Enable PMU power clamp
2	RW	0x0	Restore The application should program this bit to enable or disable restore mode from the PMU module. 1'b0: DWC_otg in normal mode of operation 1'b1: DWC_otg in restore mode *Note: This bit must not be written to during normal mode of operation. This bit is valid only when OTG_EN_PWROPT = 2*.
1	RW	0x0	PMUActv PMU Active This is bit is to enable or disable the PMU logic. 1'b0: Disable PMU module 1'b1: Enable PMU module *Note: This bit must not be written to during normal mode of operation.*

Bit	Attr	Reset Value	Description
0	RW	0x0	PMUIntSel PMU Interrupt Select When the hibernation functionality is selected using the configuration option OTG_EN_PWR_OPT = 2, a write to this bit with 1'b1 enables the PMU to generate interrupts to the application. During this state all interrupts from the core module are blocked to the application. Note: This bit must be set to 1'b1 before the core is put into hibernation 1'b0: Internal DWC_otg_core interrupt is selected 1'b1: the external DWC_otg_pmu interrupt is selected Note: This bit must not be written to during normal mode of operation.

USBOTG_GDFIFOCFG
Address: Operational Base + offset (0x005c)
Global DFIFO Software Config Register

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	EPInfoBaseAddr This field provides the start address of the EP info controller.
15:0	RW	0x0000	GDFIFOCfg This field is for dynamic programming of the DFIFO Size. This value takes effect only when the application programs a non-zero value to this register.The core does not have any corrective logic if the FIFO sizes are programmed incorrectly.

USBOTG_GADPCTL
Address: Operational Base + offset (0x0060)
ADP Timer,Control and Status Register

Bit	Attr	Reset Value	Description
31:29	RO	0x0	reserved
28:27	R/W SC	0x0	AR Access Request 2'b00 Read/Write Valid (updated by the core) 2'b01 Read 2'b10 Write 2'b11 Reserved
26	RW	0x0	AdpTmoutMsk ADP Timeout Interrupt Mask When this bit is set, it unmasks the interrupt because of AdpTmoutInt. This bit is valid only if OTG_Ver = 1'b1(GOTGCTL[20]).

Bit	Attr	Reset Value	Description
25	RW	0x0	AdpSnsIntMsk ADP Sense Interrupt Mask When this bit is set, it unmasks the interrupt due to AdpSnsInt. This bit is valid only if OTG_Ver = 1'b1(GOTGCTL[20]).
24	RW	0x0	AdpPrbIntMsk ADP Probe Interrupt Mask When this bit is set, it unmasks the interrupt due to AdpPrbInt. This bit is valid only if OTG_Ver = 1'b1(GOTGCTL[20]).
23	W1C	0x0	AdpTmoutInt ADP Timeout Interrupt This bit is relevant only for an ADP probe. When this bit is set, it means that the ramp time has completed (GADPCTL.RTIM has reached its terminal value of 0x7FF). This is a debug feature that allows software to read the ramp time after each cycle. This bit is valid only if OTG_Ver = 1'b1.
22	W1C	0x0	AdpSnsInt ADP Sense Interrupt When this bit is set, it means that the VBUS voltage is greater than VadpSns value or VadpSns is reached. This bit is valid only if OTG_Ver = 1'b1 (GOTGCTL[20]).
21	W1C	0x0	AdpPrbInt ADP Probe Interrupt When this bit is set, it means that the VBUS voltage is greater than VadpPrb or VadpPrb is reached. This bit is valid only if OTG_Ver = 1'b1 (GOTGCTL[20]).
20	RW	0x0	ADPEn ADP Enable When set, the core performs either ADP probing or sensing based on EnaPrb or EnaSns. This bit is valid only if OTG_Ver = 1'b1 (GOTGCTL[20]).
19	R/W SC	0x0	ADPRes ADP Reset When set, ADP controller is reset. This bit is auto-cleared after the reset procedure is complete in ADP controller. This bit is valid only if OTG_Ver = 1'b1 (GOTGCTL[20]).
18	RW	0x0	EnaSns Enable Sense When programmed to 1'b1, the core performs a sense operation. This bit is valid only if OTG_Ver = 1'b1 (GOTGCTL[20]).
17	RW	0x0	EnaPrb Enable Probe When programmed to 1'b1, the core performs a probe operation. This bit is valid only if OTG_Ver = 1'b1 (GOTGCTL[20]).

Bit	Attr	Reset Value	Description
16:6	RO	0x000	RTIM RAMP TIME These bits capture the latest time it took for VBUS to ramp from VADP_SINK to VADP_PRB. The bits are defined in units of 32 kHz clock cycles as follows: 0x000 - 1 cycles 0x001 - 2 cycles 0x002 - 3 cycles and so on till 0x7FF - 2048 cycles A time of 1024 cycles at 32 kHz corresponds to a time of 32 msec. (Note for scaledown ramp_timeout = prb_delta == 2'b00 => 200 cycles prb_delta == 2'b01 => 100 cycles prb_delta == 2'b01 => 50 cycles prb_delta == 2'b01 => 25 cycles.)
5:4	RW	0x0	PrbPer Probe Period These bits sets the TadpPrd as follows: 2'b00 - 0.625 to 0.925 sec (typical 0.775 sec) 2'b01 - 1.25 to 1.85 sec (typical 1.55 sec) 2'b10 - 1.9 to 2.6 sec (typical 2.275 sec) 2'b11 - Reserved (PRB_PER is also scaledown prb_per== 2'b00 => 400 ADP clocks prb_per== 2'b01 => 600 ADP clocks prb_per== 2'b10 => 800 ADP clocks prb_per==2'b11 => 1000 ADP clocks)
3:2	RW	0x0	PrbDelta Probe Delta These bits set the resolution for RTIM value. The bits are defined in units of 32 kHz clock cycles as follows: 2'b00 - 1 cycles 2'b01 - 2 cycles 2'b10 - 3 cycles 2'b11 - 4 cycles For example if this value is chosen to 2'b01, it means that RTIM increments forevery three 32Khz clock cycles.

Bit	Attr	Reset Value	Description
1:0	RW	0x0	PrbDschg Probe Discharge These bits set the times for TadpDschg. These bits are defined as follows: 2'b00 4 msec (Scaledown 2 32Khz clock cycles) 2'b01 8 msec (Scaledown 4 32Khz clock cycles) 2'b10 16 msec (Scaledown 8 32Khz clock cycles) 2'b11 32 msec (Scaledown 16 32Khz clock cycles)

USBOTG_HPTXFSIZ

Address: Operational Base + offset (0x0100)

Host Periodic Transmit FIFO Size Register

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	PTxFSize Host Periodic TxFIFO Depth This value is in terms of 32-bit words. Minimum value is 16 Maximum value is 32,768 The power-on reset value of this register is specified as the Largest Host Mode Periodic Tx Data FIFO Depth (parameter OTG_TX_HPERIO_DFIFO_DEPTH). If Enable Dynamic FIFO Sizing was deselected (parameter OTG_DFIFO_DYNAMIC = 0), these flops are optimized, and reads return the power-on value. If Enable Dynamic FIFO Sizing was selected (parameter OTG_DFIFO_DYNAMIC = 1), you can write a new value in this field. Programmed values must not exceed the power-on value set.
15:0	RW	0x0000	PTxFStAddr Host Periodic TxFIFO Start Address The power-on reset value of this register is the sum of the Largest Rx Data FIFO Depth and Largest Non-periodic Tx Data FIFO Depth specified. These parameters are: In shared FIFO operation: OTG_RX_DFIFO_DEPTH + OTG_TX_NPERIO_DFIFO_DEPTH. In dedicated FIFO mode: OTG_RX_DFIFO_DEPTH + OTG_TX_HNPERIO_DFIFO_DEPTH. If Enable Dynamic FIFO Sizing was deselected (parameter OTG_DFIFO_DYNAMIC = 0), these flops are optimized, and reads return the power-on value. If Enable Dynamic FIFO Sizing was selected(parameter OTG_DFIFO_DYNAMIC = 1), you can write a new value in this field.Programmed values must not exceed the power-on value.

USBOTG_DIEPTXFn

Address: Operational Base + offset (0x0104)

Device Periodic Transmit FIFO-n Size Register

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	INEP1TxFDep IN Endpoint TxFIFO Depth This value is in terms of 32-bit words. Minimum value is 16 Maximum value is 32,768 The power-on reset value of this register is specified as the Largest IN Endpoint FIFO number Depth (parameter OTG_TX_DINEP_DFIFO_DEPTH_n)(0 < n <= 15). If Enable Dynamic FIFO Sizing was deselected (parameter OTG_DFIFO_DYNAMIC = 0), these flops are optimized, and reads return the power-on value. If Enable Dynamic FIFO Sizing was selected (parameter OTG_DFIFO_DYNAMIC = 1), you can write a new value in this field. Programmed values must not exceed the power-on value.
15:0	RW	0x0000	INEP1TxFStAddr IN Endpoint FIFO1 Transmit RAM Start Address This field contains the memory start address for IN endpoint Transmit FIFOn (0 < n <= 15). The power-on reset value of this register is specified as the Largest Rx Data FIFO Depth (parameter OTG_RX_DFIFO_DEPTH). OTG_RX_DFIFO_DEPTH + SUM 0 to n-1 (OTG_DINEP_TXFIFO_DEPTH_n) For example start address of IN endpoint FIFO 1 is OTG_RX_DFIFO_DEPTH + OTG_DINEP_TXFIFO_DEPTH_0. The start address of IN endpoint FIFO 2 is OTG_RX_DFIFO_DEPTH + OTG_DINEP_TXFIFO_DEPTH_0 + OTG_DINEP_TXFIFO_DEPTH_1. If Enable Dynamic FIFO Sizing? was deselected (parameter OTG_DFIFO_DYNAMIC = 0), these flops are optimized, and reads return the power-on value. If Enable Dynamic FIFO Sizing was selected (parameter OTG_DFIFO_DYNAMIC = 1), and you have programmed a new value for RxFIFO depth, you can write that value in this field. Programmed values must not exceed the power-on value set.

USBOTG_HCFG

Address: Operational Base + offset (0x0400)

Host Configuration Register

Bit	Attr	Reset Value	Description
31:27	RO	0x0	reserved

Bit	Attr	Reset Value	Description
26	RW	0x0	PerSchedEna Enable Periodic Scheduling Applicable in Scatter/Gather DMA mode only. Enables periodic scheduling within the core. Initially, the bit is reset. The core will not process any periodic channels. As soon as this bit is set, the core will get ready to start scheduling periodic channels and sets HCFG.PerSchedStat. The setting of HCFG.PerSchedStat indicates the core has enabled periodic scheduling. Once HCFG.PerSchedEna is set, the application is not supposed to again reset the bit unless HCFG.PerSchedStat is set. As soon as this bit is reset, the core will get ready to stop scheduling periodic channels and resets HCFG.PerSchedStat. In non Scatter/Gather DMA mode, this bit is reserved.
25:24	RW	0x0	FrListEn Frame List Entries The value in the register specifies the number of entries in the Frame list. This field is valid only in Scatter/Gather DMA mode.
23	RW	0x0	DescDMA Enable Scatter/gather DMA in Host mode When the Scatter/Gather DMA option selected during configuration of the RTL, the application can set this bit during initialization to enable the Scatter/Gather DMA operation. NOTE: This bit must be modified only once after a reset. The following combinations are available for programming: GAHBCFG.DMAEn=0, HCFG.DescDMA=0 => Slave mode GAHBCFG.DMAEn=0, HCFG.DescDMA=1 => Invalid GAHBCFG.DMAEn=1, HCFG.DescDMA=0 => Buffered DMA mode GAHBCFG.DMAEn=1, HCFG.DescDMA=1 => Scatter/Gather DMA mode In non Scatter/Gather DMA mode, this bit is reserved.
22:16	RO	0x0	reserved
15:8	RW	0x00	ResValid Resume Validation Period This field is effective only when HCFG.Ena32KHzS is set. It controls the resume period when the core resumes from suspend. The core counts the ResValid number of clock cycles to detect a valid resume when this is set.
7	RW	0x0	Ena32KHzS Enable 32-KHz Suspend Mode This bit can only be set if the USB 1.1 Full-Speed Serial Transceiver Interface has been selected. If USB 1.1 Full-Speed Serial Transceiver Interface has not been selected, this bit must be zero. When the USB 1.1 Full-Speed Serial Transceiver Interface is chosen and this bit is set, the core expects the 48-MHz PHY clock to be switched to 32 KHz during a suspend.

Bit	Attr	Reset Value	Description
6:3	RO	0x0	reserved
2	RW	0x0	FSLSSupp FS- and LS-Only Support The application uses this bit to control the core enumeration speed. Using this bit, the application can make the core enumerate as a FS host, even if the connected device supports HS traffic. Do not make changes to this field after initial programming. 1'b0: HS/FS/LS, based on the maximum speed supported by the connected device 1'b1: FS/LS-only, even if the connected device can support HS
1:0	RW	0x0	FSLSPclkSel FS/LS PHY Clock Select 2'b00: PHY clock is running at 30/60 MHz 2'b01: PHY clock is running at 48 MHz Others: Reserved

USBOTG_HFIR
Address: Operational Base + offset (0x0404)
Host Frame Interval Register

Bit	Attr	Reset Value	Description
31:16	RO	0x0	reserved
15:0	RW	0x0000	FrInt Frame Interval The value that the application programs to this field specifies the interval between two consecutive SOFs (FS) or micro-SOFs (HS) or Keep-Alive tokens (HS). This field contains the number of PHY clocks that constitute the required frame interval. The default value set in this field for a FS operation when the PHY clock frequency is 60MHz. The application can write a value to this register only after the Port Enable bit of the Host Port Control and Status register (HPRT.PrtEnaPort) has been set. If no value is programmed, the core calculates the value based on the PHY clock specified in the FS/LS PHY Clock Select field of the Host Configuration register (HCFG.FSLSPclkSel). Do not change the value of this field after the initial configuration. 125 us * (PHY clock frequency for HS) 1 ms * (PHY clock frequency for FS/LS)

USBOTG_HFNUM
Address: Operational Base + offset (0x0408)
Host Frame Number/Frame Time Remaining Register

Bit	Attr	Reset Value	Description
31:16	RO	0x0000	FrRem Frame Time Remaining Indicates the amount of time remaining in the current micro frame (HS) or frame (FS/LS), in terms of PHY clocks. This field decrement on each PHY clock. When it reaches zero, this field is reloaded with the value in the Frame Interval register and a new SOF is transmitted on the USB.
15:0	RO	0xffff	FrNum Frame Number This field increment when a new SOF is transmitted on the USB, and is reset to 0 when it reaches 16'h3FFF. This field is writable only if Remove Optional Features was not selected (OTG_RM_OTG_FEATURES = 0). Otherwise, reads return the frame number value.

USBOTG_HPTXSTS
Address: Operational Base + offset (0x0410)
Host Periodic Transmit FIFO/Queue Status Register

Bit	Attr	Reset Value	Description
31:24	RO	0x00	PTxQTop Top of the Periodic Transmit Request Queue This indicates the entry in the Periodic Tx Request Queue that is currently being processes by the MAC. This register is used for debugging. Bit [31]: Odd/Even (micro)frame 1'b0: send in even (micro)frame 1'b1: send in odd (micro)frame Bits [30:27]: Channel/endpoint number Bits [26:25]: Type 2'b00: IN/OUT 2'b01: Zero-length packet 2'b10: CSPLIT 2'b11: Disable channel command Bit [24]: Terminate (last entry for the selected channel/endpoint)
23:16	RO	0x00	PTxQSpcAvail Periodic Transmit Request Queue Space Available Indicates the number of free locations available to be written in the Periodic Transmit Request Queue. This queue holds both IN and OUT requests. 8'h0: Periodic Transmit Request Queue is full 8'h1: 1 location available 8'h2: 2 locations available n: n locations available (0 <=n <= 16) Others: Reserved

Bit	Attr	Reset Value	Description
15:0	RW	0x0000	PTxFSpcAvail Periodic Transmit Data FIFO Space Available Indicates the number of free locations available to be written to in the Periodic TxFIFO. Values are in terms of 32-bit words 16'h0: Periodic TxFIFO is full 16'h1: 1 word available 16'h2: 2 words available 16'hn: n words available (where 0 . n . 32,768) 16'h8000: 32,768 words available Others: Reserved

USBOTG_HAINT
Address: Operational Base + offset (0x0414)
Host All Channels Interrupt Reigster

Bit	Attr	Reset Value	Description
31:16	RO	0x0	reserved
15:0	RO	0x0000	HAINT Channel Interrupts One bit per channel: Bit 0 for Channel 0, bit 15 for Channel 15

USBOTG_HAINTMSK
Address: Operational Base + offset (0x0418)
Host All Channels Interrupt Mask Register

Bit	Attr	Reset Value	Description
31:16	RO	0x0	reserved
15:0	RW	0x0000	HAINTMsk Channel Interrupt Mask One bit per channel: Bit 0 for channel 0, bit 15 for channel 15

USBOTG_HPRT
Address: Operational Base + offset (0x0440)
Host Port Control and Status Register

Bit	Attr	Reset Value	Description
31:19	RO	0x0	reserved
18:17	RO	0x0	PrtSpd Port Speed Indicates the speed of the device attached to this port. 2'b00: High speed 2'b01: Full speed 2'b10: Low speed 2'b11: Reserved

Bit	Attr	Reset Value	Description
16:13	RW	0x0	PrtTstCtl Port Test Control The application writes a nonzero value to this field to put the port into a Test mode,and the corresponding pattern is signaled on the port. 4'b0000: Test mode disabled 4'b0001: Test_J mode 4'b0010: Test_K mode 4'b0011: Test_SE0_NAK mode 4'b0100: Test_Packet mode 4'b0101: Test_Force_Enable Others: Reserved
12	R/W SC	0x0	PrtPwr Port Power The application uses this field to control power to this port (write 1'b1 to set to 1'b1and write 1'b0 to set to 1'b0), and the core can clear this bit on an over current condition. 1'b0: Power off 1'b1: Power on
11:10	RO	0x0	PrtLnSts Port Line Status Indicates the current logic level USB data lines Bit [10]: Logic level of D+ Bit [11]: Logic level of D
9	RO	0x0	reserved
8	RW	0x0	PrtRst Port Reset When the application sets this bit, a reset sequence is started on this port. The application must time the reset period and clear this bit after the reset sequence is complete. 1'b0: Port not in reset 1'b1: Port in reset To start a reset on the port, the application must leave this bit set for at least the minimum duration mentioned below, as specified in the USB 2.0 specification, Section 7.1.7.5. The application can leave it set for another 10 ms in addition to the required minimum duration, before clearing the bit, even though there is no maximum limit set by the USB standard. High speed: 50 ms Full speed/Low speed: 10 ms

Bit	Attr	Reset Value	Description
7	R/W SC	0x0	PrtSusp Port Suspend The application sets this bit to put this port in Suspend mode. The core only stops sending SOFs when this is set. To stop the PHY clock, the application must set the Port Clock Stop bit, which asserts suspend input pin of the PHY. The read value of this bit reflects the current suspend status of the port. This bit is cleared by the core after a remote wakeup signal is detected or the application sets the Port Reset bit or Port Resume bit in this register or the Resume/Remote Wakeup Detected Interrupt bit or Disconnect Detected Interrupt bit in the Core Interrupt register (GINTSTS.WkUpInt or GINTSTS.DisconnInt, respectively). 1'b0: Port not in Suspend mode 1'b1: Port in Suspend mode
6	R/W SC	0x0	PrtRes Port Resume The application sets this bit to drive resume signaling on the port. The core continues to drive the resume signal until the application clears this bit. If the core detects a USB remote wakeup sequence, as indicated by the Port Resume/Remote Wakeup Detected Interrupt bit of the Core Interrupt register (GINTSTS.WkUpInt), the core starts driving resume signaling without application intervention and clears this bit when it detects a disconnect condition. The read value of this bit indicates whether the core is currently driving resume signaling. 1'b0: No resume driven 1'b1: Resume driven When LPM is enabled and the core is in the L1 (Sleep) state, setting this bit results in the following behavior: The core continues to drive the resume signal until a pre-determined time specified in the GLPMCFG.HIRD_Thres[3:0] field. If the core detects a USB remote wakeup sequence, as indicated by the Port L1 Resume/Remote L1 Wakeup Detected Interrupt bit of the Core Interrupt register (GINTSTS.L1WkUpInt), the core starts driving resume signaling without application intervention and clears this bit at the end of the resume. The read value of this bit indicates whether the core is currently driving resume signaling. 1'b0: No resume driven 1'b1: Resume driven

Bit	Attr	Reset Value	Description
5	W1C	0x0	PrtOvrCurrChng Port Overcurrent Change The core sets this bit when the status of the Port Overcurrent Active bit (bit 4) in this register changes.
4	RO	0x0	PrtOvrCurrAct Port Overcurrent Active Indicates the overcurrent condition of the port. 1'b0: No overcurrent condition 1'b1: Overcurrent condition
3	W1C	0x0	PrtEnChng Port Enable/Disable Change The core sets this bit when the status of the Port Enable bit [2] of this register changes.
2	W1C	0x0	PrtEna Port Enable A port is enabled only by the core after a reset sequence, and is disabled by an over-current condition, a disconnect condition, or by the application clearing this bit. The application cannot set this bit by a register write. It can only clear it to disable the port. This bit does not trigger any interrupt to the application. 1'b0: Port disabled 1'b1: Port enabled
1	W1C	0x0	PrtConnDet Port Connect Detected The core sets this bit when a device connection is detected to trigger an interrupt to the application using the Host Port Interrupt bit of the Core Interrupt register (GINTSTS.PrtInt). The application must write a 1 to this bit to clear the interrupt.
0	RO	0x0	PrtConnSts Port Connect Status 0: No device is attached to the port. 1: A device is attached to the port.

USBOTG_HCCHARn
Address: Operational Base + offset (0x0500)
Host Channel-n Characteristics Register

Bit	Attr	Reset Value	Description
31	R/WSC	0x0	ChEna Channel Enable When Scatter/Gather mode is enabled 1'b0: Indicates that the descriptor structure is not yet ready. 1'b1: Indicates that the descriptor structure and data buffer with data is setup and this channel can access the descriptor. When Scatter/Gather mode is disabled, This field is set by the application and cleared by the OTG host. 1'b0: Channel disabled 1'b1: Channel enabled
30	R/WSC	0x0	ChDis Channel Disable The application sets this bit to stop transmitting/receiving data on a channel, even before the transfer for that channel is complete. The application must wait for the Channel Disabled interrupt before treating the channel as disabled.
29	RW	0x0	OddFrm Odd Frame This field is set (reset) by the application to indicate that the OTG host must perform a transfer in an odd (micro)frame. This field is applicable for only periodic (isochronous and interrupt) transactions. 1'b0: Even (micro)frame 1'b1: Odd (micro)frame This field is not applicable for Scatter/Gather DMA mode and need not be programmed by the application and is ignored by the core.
28:22	RW	0x00	DevAddr Device Address This field selects the specific device serving as the data source or sink.

Bit	Attr	Reset Value	Description
21:20	RW	0x0	MC_EC Multi Count (MC) / Error Count (EC) When the Split Enable bit of the Host Channel-n Split Control register (HCSPLTn.SpltEna) is reset (1'b0), this field indicates to the host the number of transactions that must be executed per microframe for this periodic endpoint. For non periodic transfers, this field is used only in DMA mode, and specifies the number packets to be fetched for this channel before the internal DMA engine changes arbitration. 2'b00: Reserved This field yields undefined results. 2'b01: 1 transaction 2'b10: 2 transactions to be issued for this endpoint per microframe 2'b11: 3 transactions to be issued for this endpoint per microframe When HCSPLTn.SpltEna is set (1'b1), this field indicates the number of immediate retries to be performed for a periodic split transactions on transaction errors. This field must be set to at least 2'b01.
19:18	RW	0x0	EPType Endpoint Type Indicates the transfer type selected. 2'b00: Control 2'b01: Isochronous 2'b10: Bulk 2'b11: Interrupt
17	RW	0x0	LSpdDev Low-Speed Device This field is set by the application to indicate that this channel is communicating to a low-speed device.
16	RO	0x0	reserved
15	RW	0x0	EPDir Endpoint Direction Indicates whether the transaction is IN or OUT. 1'b0: OUT 1'b1: IN
14:11	RW	0x0	EPNum Endpoint Number Indicates the endpoint number on the device serving as the data source or sink.
10:0	RW	0x000	MPS Maximum Packet Size Indicates the maximum packet size of the associated endpoint.

USBOTG_HCSPLTn

Address: Operational Base + offset (0x0504)
Host Channel-n Split Control Register

Bit	Attr	Reset Value	Description
31	RW	0x0	SpltEna Split Enable The application sets this field to indicate that this channel is enabled to perform split transactions.
30:17	RO	0x0	reserved
16	RW	0x0	CompSplt Do Complete Split The application sets this field to request the OTG host to perform a complete split transaction.
15:14	RW	0x0	XactPos Transaction Position This field is used to determine whether to send all, first, middle, or last payloads with each OUT transaction. 2'b11: All. This is the entire data payload is of this transaction (which is less than or equal to 188 bytes). 2'b10: Begin. This is the first data payload of this transaction (which is larger than 188 bytes). 2'b00: Mid. This is the middle payload of this transaction (which is larger than 188bytes). 2'b01: End. This is the last payload of this transaction (which is larger than 188 bytes).
13:7	RW	0x00	HubAddr Hub Address This field holds the device address of the transaction translator's hub.
6:1	RO	0x0	reserved
0	RW	0x0	PrtAddr Port Address This field is the port number of the recipient transaction translator.

USBOTG_HCINTn
Address: Operational Base + offset (0x0508)
Host Channel-n Interrupt Register

Bit	Attr	Reset Value	Description
31:14	RO	0x0	reserved
13	W1C	0x0	DESC_LST_ROLLIntr Descriptor rollover interrupt This bit is valid only when Scatter/Gather DMA mode is enabled. The core sets this bit when the corresponding channel's descriptor list rolls over. For non Scatter/Gather DMA mode, this bit is reserved.

Bit	Attr	Reset Value	Description
12	W1C	0x0	XCS_XACT_ERR Excessive Transaction Error This bit is valid only when Scatter/Gather DMA mode is enabled. The core sets this bit when 3 consecutive transaction errors occurred on the USB bus. XCS_XACT_ERR will not be generated for Isochronous channels. For non Scatter/Gather DMA mode, this bit is reserved.
11	W1C	0x0	BNAIntr BNA (Buffer Not Available) Interrupt This bit is valid only when Scatter/Gather DMA mode is enabled. The core generates this interrupt when the descriptor accessed is not ready for the Core to process. BNA will not be generated for Isochronous channels. For non Scatter/Gather DMA mode, this bit is reserved.
10	W1C	0x0	DataTglErr Data Toggle Error In Scatter/Gather DMA mode, the interrupt due to this bit is masked in the core.
9	W1C	0x0	FrmOvrun Frame Overrun In Scatter/Gather DMA mode, the interrupt due to this bit is masked in the core
8	W1C	0x0	BblErr Babble Error In Scatter/Gather DMA mode, the interrupt due to this bit is masked in the core.
7	W1C	0x0	XactErr Transaction Error Indicates one of the following errors occurred on the USB: CRC check failure, Timeout, Bit stuff error, False EOP. In Scatter/Gather DMA mode, the interrupt due to this bit is masked in the core.
6	WO	0x0	NYET NYET Response Received Interrupt In Scatter/Gather DMA mode, the interrupt due to this bit is masked in the core.
5	W1C	0x0	ACK ACK Response Received/Transmitted Interrupt In Scatter/Gather DMA mode, the interrupt due to this bit is masked in the core.
4	W1C	0x0	NAK NAK Response Received Interrupt In Scatter/Gather DMA mode, the interrupt due to this bit is masked in the core.

Bit	Attr	Reset Value	Description
3	W1C	0x0	STALL STALL Response Received Interrupt In Scatter/Gather DMA mode, the interrupt due to this bit is masked in the core.
2	W1C	0x0	AHBErr AHB Error This is generated only in DMA mode when there is an AHB error during AHB read/write.The application can read the corresponding channel's DMA address register to get the error address.
1	W1C	0x0	ChHltd Channel Halted In non Scatter/Gather DMA mode, it indicates the transfer completed abnormally either because of any USB transaction error or in response to disable request by the application or because of a completed transfer. In Scatter/Gather DMA mode, this indicates that transfer completed due to any of the following: EOL being set in descriptor, AHB error, Excessive transaction errors, In response to disable request by the application, Babble, Stall, Buffer Not Available (BNA)
0	W1C	0x0	XferCompl Transfer Completed For Scatter/Gather DMA mode, it indicates that current descriptor processing got completed with IOC bit set in its descriptor. In non Scatter/Gather DMA mode, it indicates that Transfer completed normally without any errors.

USBOTG_HCINTMSKn

Address: Operational Base + offset (0x050c)

Host Channel-n Interrupt Mask Register

Bit	Attr	Reset Value	Description
31:14	RO	0x0	reserved
13	RW	0x0	DESC_LST_ROLLIntrMsk Descriptor rollover interrupt Mask register This bit is valid only when Scatter/Gather DMA mode is enabled. In non Scatter/Gather DMA mode, this bit is reserved.
12	RO	0x0	reserved
11	RW	0x0	BNAIntrMsk BNA (Buffer Not Available) Interrupt mask register This bit is valid only when Scatter/Gather DMA mode is enabled. In non Scatter/Gather DMA mode, this bit is reserved.
10	RW	0x0	DataTglErrMsk Data Toggle Error Mask This bit is not applicable in Scatter/Gather DMA mode.

Bit	Attr	Reset Value	Description
9	RW	0x0	FrmOvrunMsk Frame Overrun Mask This bit is not applicable in Scatter/Gather DMA mode.
8	RW	0x0	BblErrMsk Babble Error Mask This bit is not applicable in Scatter/Gather DMA mode.
7	RW	0x0	XactErrMsk Transaction Error Mask This bit is not applicable in Scatter/Gather DMA mode
6	RW	0x0	NyetMsk NYET Response Received Interrupt Mask This bit is not applicable in Scatter/Gather DMA mode.
5	RW	0x0	AckMsk ACK Response Received/Transmitted Interrupt Mask This bit is not applicable in Scatter/Gather DMA mode.
4	RW	0x0	NakMsk NAK Response Received Interrupt Mask This bit is not applicable in Scatter/Gather DMA mode.
3	RW	0x0	StallMsk STALL Response Received Interrupt Mask This bit is not applicable in Scatter/Gather DMA mode.
2	RW	0x0	AHBErrMsk AHB Error Mask Note: This bit is only accessible when OTG_ARCHITECTURE = 2
1	RW	0x0	ChHltdMsk Channel Halted Mask
0	RW	0x0	XferComplMsk Transfer Completed Mask This bit is valid only when Scatter/Gather DMA mode is enabled. In non Scatter/Gather DMA mode, this bit is reserved.

USBOTG_HCTSIZn

Address: Operational Base + offset (0x0510)

Host Channel-n Transfer Size Register

Bit	Attr	Reset Value	Description
31	RW	0x0	DoPng Do Ping This bit is used only for OUT transfers. Setting this field to 1 directs the host to do PING protocol. Note: Do not set this bit for IN transfers. If this bit is set for IN transfers it disables the channel.

Bit	Attr	Reset Value	Description
30:29	RW	0x0	Pid PID The application programs this field with the type of PID to use for the initial transaction. The host maintains this field for the rest of the transfer. 2'b00: DATA0 2'b01: DATA2 2'b10: DATA1 2'b11: MDATA (non-control)/SETUP (control)
28:19	RW	0x000	PktCnt Packet Count This field is programmed by the application with the expected number of packets to be transmitted (OUT) or received (IN).The host decrements this count on every successful transmission or reception of an OUT/IN packet. Once this count reaches zero, the application is interrupted to indicate normal completion. The width of this counter is specified as Width of Packet Counters (parameter OTG_PACKET_COUNT_WIDTH).
18:0	RW	0x00000	XferSize Transfer Size For an OUT, this field is the number of data bytes the host sends during the transfer. For an IN, this field is the buffer size that the application has Reserved for the transfer. The application is expected to program this field as an integer multiple of the maximum packet size for IN transactions (periodic and non-periodic). The width of this counter is specified as Width of Transfer Size Counters (parameter OTG_TRANS_COUNT_WIDTH).

USBOTG_HCDMAn
Address: Operational Base + offset (0x0514)
Host Channel-n DMA Address Register

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	DMAAddr DMA Address This field holds the start address in the external memory from which the data for the endpoint must be fetched or to which it must be stored. This register is incremented on every AHB transaction.

USBOTG_HCDMABn
Address: Operational Base + offset (0x051c)
Host Channel-n DMA Buffer Address Register

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	HCDMABn Holds the current buffer address This register is updated as and when the data transfer for the corresponding end point is in progress. This register is present only in Scatter/Gather DMA mode. Otherwise this field is reserved.

USBOTG_DCFG
Address: Operational Base + offset (0x0800)
Device Cconfiguration Register

Bit	Attr	Reset Value	Description
31:26	RW	0x02	ResValid Resume Validation Period This field controls the period when the core resumes from asuspended. When this bit is set, the core counts for the ResValid number of clock cycles to detect a valid resume. This field is effective only when DCFG.Ena32KHzSusp is set.
25:24	RW	0x0	PerSchIntvl Periodic Scheduling Interval PerSchIntvl must be programmed only for Scatter/Gather DMA mode. Description: This field specifies the amount of time the Internal DMA engine must allocate for fetching periodic IN endpoint data. Based on the number of periodic endpoints, this value must be specified as 25,50 or 75% of (micro)frame. When any periodic endpoints are active, the internal DMA engine allocates the specified amount of time in fetching periodic IN endpoint data. When no periodic endpoints are active, then the internal DMA engine services nonperiodic endpoints, ignoring this field. After the specified time within a (micro)frame, the DMA switches to fetching for nonperiodic endpoints. 2'b00: 25% of (micro)frame. 2'b01: 50% of (micro)frame. 2'b10: 75% of (micro)frame. 2'b11: Reserved.

Bit	Attr	Reset Value	Description
23	RW	0x0	DescDMA Enable Scatter/Gather DMA in Device mode When the Scatter/Gather DMA option selected during configuration of the RTL, the application can set this bit during initialization to enable the Scatter/Gather DMA operation. NOTE: This bit must be modified only once after a reset. The following combinations are available for programming: GAHBCFG.DMAEn=0,DCFG.DescDMA=0 => Slave mode GAHBCFG.DMAEn=0,DCFG.DescDMA=1 => Invalid GAHBCFG.DMAEn=1,DCFG.DescDMA=0 => Buffered DMA mode GAHBCFG.DMAEn=1,DCFG.DescDMA=1 => Scatter/Gather DMA mode
22:18	RW	0x08	EPMisCnt IN Endpoint Mismatch Count This field is valid only in shared FIFO operation. The application programs this filed with a count that determines when the core generates an Endpoint Mismatch interrupt (GINTSTS.EPMis). The core loads this value into an internal counter and decrements it. The counter is reloaded whenever there is a match or when the counter expires. The width of this counter depends on the depth of the Token Queue.
17:13	RO	0x0	reserved
12:11	RW	0x0	PerFrInt Periodic Frame Interval Indicates the time within a (micro)frame at which the application must be notified using the End Of Periodic Frame Interrupt. This can be used to determine if all the isochronous traffic for that (micro) frame is complete. 2'b00: 80% of the (micro)frame interval 2'b01: 85% 2'b10: 90% 2'b11: 95%
10:4	RW	0x00	DevAddr Device Address The application must program this field after every SetAddress control command.
3	RW	0x0	Ena32KHzS Enable 32-KHz Suspend Mode When the USB 1.1 Full-Speed Serial Transceiver Interface is chosen and this bit is set, the core expects the 48-MHz PHY clock to be switched to 32 KHz during a suspend. This bit can only be set if USB 1.1 Full-Speed Serial Transceiver Interface has been selected. If USB 1.1 Full-Speed Serial Transceiver Interface has not been selected, this bit must be zero.

Bit	Attr	Reset Value	Description
2	RW	0x0	NZStsOUTHShk Non-Zero-Length Status OUT Handshake The application can use this field to select the handshake the core sends on receiving a nonzero-length data packet during the OUT transaction of a control transfer's Status stage. 1'b1: Send a STALL handshake on a nonzero-length status OUT transaction and do not send the received OUT packet to the application. 1'b0: Send the received OUT packet to the application (zero-length or non-zero-length) and send a handshake based on the NAK and STALL bits for the endpoint in the Device Endpoint Control register.
1:0	RW	0x0	DevSpd Device Speed Indicates the speed at which the application requires the core to enumerate, or the maximum speed the application can support. However, the actual bus speed is determined only after the chirp sequence is completed, and is based on the speed of the USB host to which the core is connected. 2'b00: High speed (USB 2.0 PHY clock is 30 MHz or 60 MHz) 2'b01: Full speed (USB 2.0 PHY clock is 30 MHz or 60 MHz) 2'b10: Reserved 2'b11: Full speed (USB 1.1 transceiver clock is 48 MHz)

USBOTG_DCTL

Address: Operational Base + offset (0x0804)

Device Control Register

Bit	Attr	Reset Value	Description
31:17	RO	0x0	reserved
16	RW	0x0	NakOnBble Set NAK automatically on babble The core sets NAK automatically for the endpoint on which babble is received.

Bit	Attr	Reset Value	Description
15	RW	0x0	IgnrFrmNum Ignore frame number for isochronous endpoints in case of Scatte Do NOT program IgnrFrmNum bit to 1'b1 when the core is operating in Threshold mode. Note: When Scatter/Gather DMA mode is enabled this feature is not applicable to high-speed, high-bandwidth transfers. When this bit is enabled, there must be only one packet per descriptor. 0: The core transmits the packets only in the frame number in which they are intended to be transmitted. 1: The core ignores the frame number, sending packets immediately as the packets are ready. Scatter/Gather: In Scatter/Gather DMA mode, when this bit is enabled, the packets are not flushed when an ISOC IN token is received for an elapsed frame. When Scatter/Gather DMA mode is disabled, this field is used by the application to enable periodic transfer interrupt. The application can program periodic endpoint transfers for multiple micro)frames. 0: Periodic transfer interrupt feature is disabled; the application must program transfers for periodic endpoints every (micro)frame 1: Periodic transfer interrupt feature is enabled; the application can program transfers for multiple (micro)frames for periodic endpoints. In non-Scatter/Gather DMA mode, the application receives transfer complete interrupt after transfers for multiple (micro)frames are completed.
14:13	RW	0x1	GMC Global Multi Count GMC must be programmed only once after initialization. Applicable only for Scatter/Gather DMA mode. This indicates the number of packets to be serviced for that end point before moving to the next end point. It is only for nonperiodic endpoints. 2'b00: Invalid. 2'b01: 1 packet. 2'b10: 2 packets. 2'b11: 3 packets. When Scatter/Gather DMA mode is disabled, this field is reserved, and reads 2'b00.
12	RO	0x0	reserved
11	RW	0x0	PWROnPrgDone Power-On Programming Done The application uses this bit to indicate that register programming is completed after a wake-up from Power Down mode.

Bit	Attr	Reset Value	Description
10	WO	0x0	CGOUTNak Clear Global OUT NAK A write to this field clears the Global OUT NAK.
9	WO	0x0	SGOUTNak Set Global OUT NAK A write to this field sets the Global OUT NAK. The application uses this bit to send a NAK handshake on all OUT endpoints. The application must set this bit only after making sure that the Global OUT NAK Effective bit in the Core Interrupt Register (GINTSTS.GOUTNakEff) is cleared.
8	WO	0x0	CGNPInNak Clear Global Non-periodic IN NAK A write to this field clears the Global Non-periodic IN NAK.
7	WO	0x0	SGNPInNak Set Global Non-periodic IN NAK A write to this field sets the Global Non-periodic IN NAK.The application uses this bit to send a NAK handshake on all non-periodic IN endpoints. The core can also set this bit when a timeout condition is detected on a non-periodic endpoint in shared FIFO operation. The application must set this bit only after making sure that the Global IN NAK Effective bit in the Core Interrupt Register (GINTSTS.GINNakEff) is cleared.
6:4	RW	0x0	TstCtl Test Control 3'b000: Test mode disabled 3'b001: Test_J mode 3'b010: Test_K mode 3'b011: Test_SE0_NAK mode 3'b100: Test_Packet mode 3'b101: Test_Force_Enable Others: Reserved
3	RO	0x0	GOUTNakSts Global OUT NAK Status 1'b0: A handshake is sent based on the FIFO Status and the NAK and STALL bit settings. 1'b1: No data is written to the RxFIFO, irrespective of space availability. Sends a NAK handshake on all packets, except on SETUP transactions. All isochronous OUT packets are dropped
2	RO	0x0	GNPINNakSts Global Non-periodic IN NAK Status 1'b0: A handshake is sent out based on the data availability in the transmit FIFO. 1'b1: A NAK handshake is sent out on all non-periodic IN endpoints, irrespective of the data availability in the transmit FIFO.

Bit	Attr	Reset Value	Description
1	RW	0x0	SftDiscon Soft Disconnect The application uses this bit to signal the DWC_otg core to do a soft disconnect. As long as this bit is set, the host does not see that the device is connected, and the device does not receive signals on the USB. The core stays in the disconnected state until the application clears this bit. 1'b0: Normal operation. When this bit is cleared after a soft disconnect, the core drives the phy_opmode_o signal on the UTMI+ to 2'b00, which generates a device connect event to the USB host. When the device is reconnected, the USB host restarts device enumeration. 1'b1: The core drives the phy_opmode_o signal on the UTMI+ to 2'b01, which generates a device disconnect event to the USB host.
0	RW	0x0	RmtWkUpSig Remote Wakeup Signaling When the application sets this bit, the core initiates remote signaling to wake the USB host. The application must set this bit to instruct the core to exit the Suspend state. As specified in the USB 2.0 specification, the application must clear this bit 1蝻ms after setting it.If LPM is enabled and the core is in the L1 (Sleep) state, when the application sets this bit, the core initiates L1 remote signaling to wake up the USB host. The application must set this bit to instruct the core to exit the Sleep state. As specified in the LPM specification, the hardware automatically clears this bit 50 us (TL1DevDrvResume) after being set by the application. The application must not set this bit when GLPMCFG bRemoteWake from the previous LPM transaction is zero.

USBOTG_DSTS

Address: Operational Base + offset (0x0808)

Device Status Register

Bit	Attr	Reset Value	Description
31:22	RO	0x0	reserved
21:8	RW	0x0000	SOFFN Frame or Microframe Number of the Received SOF When the core is operating at high speed, this field contains a microframe number. When the core is operating at full or low speed, this field contains a frame number.
7:4	RO	0x0	reserved

Bit	Attr	Reset Value	Description
3	RW	0x0	ErrticErr Erratic Error The core sets this bit to report any erratic errors (phy_rxvalid_i/phy_rxvldh_i or phy_rxactive_i is asserted for at least 2 ms, due to PHY error) seen on the UTMI+. Due to erratic errors, the DWC_otg core goes into Suspended state and an interrupt is generated to the application with Early Suspend bit of the Core Interrupt register (GINTSTS.ErlySusp). If the early suspend is asserted due to an erratic error, the application can only perform a soft disconnect recover.
2:1	RW	0x0	EnumSpd Enumerated Speed Indicates the speed at which the DWC_otg core has come up after speed detection through a chirp sequence. 2'b00: High speed (PHY clock is running at 30 or 60 MHz) 2'b01: Full speed (PHY clock is running at 30 or 60 MHz) 2'b10: Low speed (PHY clock is running at 48 MHz, internal phy_clk at 6 MHz) 2'b11: Full speed (PHY clock is running at 48 MHz) Low speed is not supported for devices using a UTMI+ PHY.
0	RW	0x0	SuspSts Suspend Status In Device mode, this bit is set as long as a Suspend condition is detected on the USB. The core enters the Suspended state when there is no activity on the utmi_linestate signal for an extended period of time. The core comes out of the suspend: When there is any activity on the utmi_linestate signal, When the application writes to the Remote Wakeup Signaling bit in the Device Control register (DCTL.RmtWkUpSig).

USBOTG_DIEPMSK

Address: Operational Base + offset (0x0810)

Device IN Endpoint common interrupt mask register

Bit	Attr	Reset Value	Description
31:14	RO	0x0	reserved
13	RW	0x0	NAKMsk NAK interrupt Mask
12:10	RO	0x0	reserved
9	RW	0x0	BNAInIntrMsk BNA Interrupt Mask
8	RW	0x0	TxfifoUndrnMsk Fifo Underrun Mask
7	RO	0x0	reserved
6	RW	0x0	INEPNakEffMsk IN Endpoint NAK Effective Mask

Bit	Attr	Reset Value	Description
5	RW	0x0	INTknEPMisMsk IN Token received with EP Mismatch Mask
4	RW	0x0	INTknTXFEmpMsk IN Token Received When TxFIFO Empty Mask
3	RW	0x0	TimeOUTMsk Timeout Condition Mask
2	RW	0x0	AHBErrMsk AHB Error Mask
1	RW	0x0	EPDisbldMsk Endpoint Disabled Interrupt Mask
0	RW	0x0	XferComplMsk Transfer Completed Interrupt Mask

USBOTG_DOEPMSK
Address: Operational Base + offset (0x0814)
Device OUT Endpoint common interrupt mask register

Bit	Attr	Reset Value	Description
31:15	RO	0x0	reserved
14	RW	0x0	NYETMsk NYET Interrupt Mask
13	RW	0x0	NAKMsk NAK Interrupt Mask
12	RW	0x0	BbleErrMsk Babble Interrupt Mask
11:10	RO	0x0	reserved
9	RW	0x0	BnaOutIntrMsk BNA interrupt Mask
8	RW	0x0	OutPktErrMsk OUT Packet Error Mask
7	RO	0x0	reserved
6	RW	0x0	Back2BackSETup Back-to-Back SETUP Packets Received Mask Applies to control OUT endpoints only.
5	RO	0x0	reserved
4	RW	0x0	OUTTknEPdisMsk OUT Token Received when Endpoint Disabled Mask Applies to control OUT endpoints only.
3	RW	0x0	SetUPMsk SETUP Phase Done Mask Applies to control endpoints only.
2	RW	0x0	AHBErrMsk AHB Error
1	RW	0x0	EPDisbldMsk Endpoint Disabled Interrupt Mask

Bit	Attr	Reset Value	Description
0	RW	0x0	XferComplMsk Transfer Completed Interrupt Mask

USBOTG_DAINT
Address: Operational Base + offset (0x0818)
Device All Endpoints interrupt register

Bit	Attr	Reset Value	Description
31:16	RO	0x0000	OutEPInt OUT Endpoint Interrupt Bits One bit per OUT endpoint: Bit 16 for OUT endpoint 0, bit 31 for OUT endpoint 15
15:0	RO	0x0000	InEpInt IN Endpoint Interrupt Bits One bit per IN Endpoint: Bit 0 for IN endpoint 0, bit 15 for endpoint 15

USBOTG_DAINTMSK
Address: Operational Base + offset (0x081c)
Device All Endpoint interrupt mask register

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	OutEpMsk OUT EP Interrupt Mask Bits One per OUT Endpoint: Bit 16 for OUT EP 0, bit 31 for OUT EP 15
15:0	RW	0x0000	InEpMsk IN EP Interrupt Mask Bits One bit per IN Endpoint: Bit 0 for IN EP 0, bit 15 for IN EP 15

USBOTG_DTKNQR1
Address: Operational Base + offset (0x0820)
Device IN token sequence learning queue read register1

Bit	Attr	Reset Value	Description
31:8	RO	0x000000	EPTkn Endpoint Token Four bits per token represent the endpoint number of the token: Bits [31:28]: Endpoint number of Token 5 Bits [27:24]: Endpoint number of Token 4 ..... Bits [15:12]: Endpoint number of Token 1 Bits [11:8]: Endpoint number of Token 0
7	RO	0x0	WrapBit Wrap Bit This bit is set when the write pointer wraps. It is cleared when the learning queue is cleared.
6:5	RO	0x0	reserved

Bit	Attr	Reset Value	Description
4:0	RO	0x00	INTknWPtr IN Token Queue Write Pointer

USBOTG_DTKNQR2
Address: Operational Base + offset (0x0824)
Device IN token sequence learning queue read register2

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	EPTkn Endpoint Token Four bits per token represent the endpoint number of the token: Bits [31:28]: Endpoint number of Token 13 Bits [27:24]: Endpoint number of Token 12 ..... Bits [7:4]: Endpoint number of Token 7 Bits [3:0]: Endpoint number of Token 6

USBOTG_DVBUSDIS
Address: Operational Base + offset (0x0828)
Device VBUS discharge time register

Bit	Attr	Reset Value	Description
31:16	RO	0x0	reserved
15:0	RW	0x0b8f	DVBUSDis Device VBUS Discharge Time Specifies the VBUS discharge time after VBUS pulsing during SRP. This value equals: VBUS discharge time in PHY clocks / 1,024.The value you use depends whether the PHY is operating at 30 MHz (16-bit data width) or 60 MHz (8-bit data width). Depending on your VBUS load, this value can need adjustment.

USBOTG_DVBUSPULSE
Address: Operational Base + offset (0x082c)
Device VBUS Pulsing Timer Register

Bit	Attr	Reset Value	Description
31:12	RO	0x0	reserved
11:0	RW	0x000	DVBUSPulse Device VBUS Pulsing Time Specifies the VBUS pulsing time during SRP. This value equals: VBUS pulsing time in PHY clocks / 1,024.The value you use depends whether the PHY is operating at 30 MHz (16-bit data width) or 60 MHz (8-bit data width).

USBOTG_DTHRCTL
Address: Operational Base + offset (0x0830)
Device Threshold Control Register

Bit	Attr	Reset Value	Description
31:28	RO	0x0	reserved
27	RW	0x1	ArbPrkEn Arbiter Parking Enable This bit controls internal DMA arbiter parking for IN endpoints. When thresholding is enabled and this bit is set to one, then the arbiter parks on the IN endpoint for which there is a token received on the USB. This is done to avoid getting into underrun conditions. By default the parking is enabled.
26	RO	0x0	reserved
25:17	RW	0x008	RxThrLen Receive Threshold Length This field specifies Receive thresholding size in DWORDS. This field also specifies the amount of data received on the USB before the core can start transmitting on the AHB. The threshold length has to be at least eight DWORDS. The recommended value for ThrLen is to be the same as the programmed AHB Burst Length (GAHBCFG.HBstLen).
16	RW	0x0	RxThrEn Receive Threshold Enable When this bit is set, the core enables thresholding in the receive direction.
15:13	RO	0x0	reserved
12:11	RW	0x0	AHBThrRatio AHB Threshold Ratio These bits define the ratio between the AHB threshold and the MAC threshold for the transmit path only. The AHB threshold always remains less than or equal to the USB threshold, because this does not increase overhead. Both the AHB and the MAC threshold must be DWORD-aligned. The application needs to program TxThrLen and the AHBThrRatio to make the AHB Threshold value DWORD aligned. If the AHB threshold value is not DWORD aligned, the core might not behave correctly. When programming the TxThrLen and AHBThrRatio, the application must ensure that the minimum AHB threshold value does not go below 8 DWORDS to meet the USB turnaround time requirements. 2'b00: AHB threshold = MAC threshold 2'b01: AHB threshold = MAC threshold / 2 2'b10: AHB threshold = MAC threshold / 4 2'b11: AHB threshold = MAC threshold / 8

Bit	Attr	Reset Value	Description
10:2	RW	0x008	TxThrLen Transmit Threshold Length This field specifies Transmit thresholding size in DWORDS. This field also forms the MAC threshold and specifies the amount of data, in bytes, to be in the corresponding endpoint transmit FIFO before the core can start a transmit on the USB. When the value of AHBThrRatio is 2'h00, the threshold length must be at least 8 DWORDS. If the AHBThrRatio is nonzero, the application must ensure that the AHB threshold value does not go below the recommended 8 DWORDs. This field controls both isochronous and non-isochronous IN endpoint thresholds. The recommended value for ThrLen is to be the same as the programmed AHB Burst Length (GAHBCFG.HBstLen).
1	RW	0x0	ISOThrEn ISO IN Endpoints Threshold Enable When this bit is set, the core enables thresholding for isochronous IN endpoints.
0	RW	0x0	NonISOThrEn Non-ISO IN Endpoints Threshold Enable When this bit is set, the core enables thresholding for Non Isochronous IN endpoints.

USBOTG_DIEPEMPMSK
Address: Operational Base + offset (0x0834)
Device IN endpoint FIFO empty interrupt mask register

Bit	Attr	Reset Value	Description
31:16	RO	0x0	reserved
15:0	RW	0x0000	InEpTxfEmpMsk IN EP Tx FIFO Empty Interrupt Mask Bits These bits act as mask bits for DIEPINTn. TxFEmp interrupt One bit per IN Endpoint: Bit 0 for IN endpoint 0 ... Bit 15 for endpoint 15

USBOTG_DEACHINT
Address: Operational Base + offset (0x0838)
Device each endpoint interrupt register

Bit	Attr	Reset Value	Description
31:16	RO	0x0000	EchOutEPInt OUT Endpoint Interrupt Bits One bit per OUT endpoint: Bit 16 for OUT endpoint 0 ... Bit 31 for OUT endpoint 15
15:0	RO	0x0000	EchInEpInt IN Endpoint Interrupt Bits One bit per IN Endpoint: Bit 0 for IN endpoint 0 ... Bit 15 for endpoint 15

USBOTG_DEACHINTMSK
Address: Operational Base + offset (0x083c)
Device each endpoint interrupt register mask

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	EchOutEpMsk OUT EP Interrupt Mask Bits One per OUT Endpoint: Bit 16 for IN endpoint 0 ... Bit 31 for endpoint 15
15:0	RW	0x0000	EchInEpMsk IN EP Interrupt Mask Bits One bit per IN Endpoint: Bit 0 for IN endpoint 0 ... Bit 15 for endpoint 15

USBOTG_DIEPEACHMSKn
Address: Operational Base + offset (0x0840)
Device each IN endpoint -n interrupt Register

Bit	Attr	Reset Value	Description
31:14	RO	0x0	reserved
13	RW	0x0	NAKMsk NAK interrupt Mask
12:10	RO	0x0	reserved
9	RW	0x0	BNAInIntrMsk BNA interrupt Mask
8	RW	0x0	TxfifoUndrnMsk Fifo Under run Mask
7	RO	0x0	reserved
6	RW	0x0	INEPNakEffMsk IN Endpoint NAK Effective Mask

Bit	Attr	Reset Value	Description
5	RW	0x0	INTknEPMisMsk IN Token received with EP Mismatch Mask
4	RW	0x0	INTknTXFEmpMsk IN Token Received When TxFIFO Empty Mask
3	RW	0x0	TimeOUTMsk Timeout Condition Mask(Non-isochronous endpoints)
2	RW	0x0	AHBErrMsk AHB Error Mask
1	RW	0x0	EPDisbldMsk Endpoint Disabled Interrupt Mask
0	RW	0x0	XferComplMsk Transfer Completed Interrupt Mask

USBOTG_DOEPEACHMSKn

Address: Operational Base + offset (0x0880)

Device each out endpoint-n interrupt register

Bit	Attr	Reset Value	Description
31:15	RO	0x0	reserved
14	RW	0x0	NYETMsk NYET interrupt Mask
13	RW	0x0	NAKMsk NAK interrupt Mask
12	RW	0x0	BbleErrMsk Babble interrupt Mask
11:10	RO	0x0	reserved
9	RW	0x0	BnaOutIntrMsk BNA interrupt Mask
8	RW	0x0	OutPktErrMsk OUT Packet Error Mask
7	RO	0x0	reserved
6	RW	0x0	Back2BackSETup Back-to-Back SETUP Packets Received Mask Applies to control OUT endpoints only.
5	RO	0x0	reserved
4	RW	0x0	OUTTknEPdisMsk OUT Token Received when Endpoint Disabled Mask Applies to control OUT endpoints only.
3	RW	0x0	SetUPMsk SETUP Phase Done Mask Applies to control endpoints only.
2	RW	0x0	AHBErrMsk AHB Error
1	RW	0x0	EPDisbldMsk Endpoint Disabled Interrupt Mask

Bit	Attr	Reset Value	Description
0	RW	0x0	XferComplMsk Transfer Completed Interrupt Mask

USBOTG_DIEPCTL0
Address: Operational Base + offset (0x0900)
Device control IN endpoint 0 control register

Bit	Attr	Reset Value	Description
31	R/WSC	0x0	EPEna Endpoint Enable When Scatter/Gather DMA mode is enabled, for IN endpoints this bit indicates that the descriptor structure and data buffer with data ready to transmit is setup. When Scatter/Gather DMA mode is disabled-such as in buffer-pointer based DMA mode-this bit indicates that data is ready to be transmitted on the endpoint. The core clears this bit before setting the following interrupts on this endpoint: Endpoint Disabled; Transfer Completed.
30	R/WSC	0x0	EPDis Endpoint Disable The application sets this bit to stop transmitting data on an endpoint, even before the transfer for that endpoint is complete. The application must wait for the Endpoint Disabled interrupt before treating the endpoint as disabled. The core clears this bit before setting the Endpoint Disabled Interrupt. The application must set this bit only if Endpoint Enable is already set for this endpoint.
29:28	RO	0x0	reserved
27	WO	0x0	SNAK Set NAK A write to this bit sets the NAK bit for the endpoint. Using this bit, the application can control the transmission of NAK handshakes on an endpoint. The core can also set this bit for an endpoint after a SETUP packet is received on that endpoint.
26	WO	0x0	CNAK Clear NAK A write to this bit clears the NAK bit for the endpoint.
25:23	RO	0x0	reserved
22	RW	0x0	TxFNum TxFIFO Number For Shared FIFO operation, this value is always set to 0, indicating that control IN endpoint 0 data is always written in the Non-Periodic Transmit FIFO. For Dedicated FIFO operation, this value is set to the FIFO number that is assigned to IN Endpoint 0.

Bit	Attr	Reset Value	Description
21	R/WSC	0x0	Stall STALL Handshake The application can only set this bit, and the core clears it, when a SETUP token is received for this endpoint. If a NAK bit, Global Non-periodic IN NAK, or Global OUT NAK is set along with this bit, the STALL bit takes priority.
20	RO	0x0	reserved
19:18	RO	0x0	EPType Endpoint Type Hardcoded to 00 for control
17	RO	0x0	NAKSts NAK Status Indicates the following: 1'b0: The core is transmitting non-NAK handshakes based on the FIFO status 1'b1: The core is transmitting NAK handshakes on this endpoint. When this bit is set, either by the application or core, the core stops transmitting data, even if there is data available in the TxFIFO. Irrespective of this bit's setting, the core always responds to SETUP data packets with an ACK handshake.
16	RO	0x0	reserved
15	RO	0x1	USBActEP USB Active Endpoint This bit is always set to 1, indicating that control endpoint 0 is always active in all configurations and interfaces.
14:11	RW	0x0	NextEp Next Endpoint Applies to non-periodic IN endpoints only. Indicates the endpoint number to be fetched after the data for the current endpoint is fetched. The core can access this field, even when the Endpoint Enable (EPEna) bit is not set. This field is not valid in Slave mode. Note: This field is valid only for Shared FIFO operations.
10:2	RO	0x0	reserved
1:0	RW	0x0	MPS Maximum Packet Size Applies to IN and OUT endpoints. The application must program this field with the maximum packet size for the current logical endpoint. 2'b00: 64 bytes 2'b01: 32 bytes 2'b10: 16 bytes 2'b11: 8 bytes

USBOTG_DIEPINTn
Address: Operational Base + offset (0x0908)

Device Endpoint-n Interrupt Register

Bit	Attr	Reset Value	Description
31:15	RO	0x0	reserved
14	W1C	0x0	NYETIntrpt NYET interrupt The core generates this interrupt when a NYET response is transmitted for a non isochronous OUT endpoint.
13	W1C	0x0	NAKIntrpt NAK interrupt The core generates this interrupt when a NAK is transmitted or received by the device. In case of isochronous IN endpoints the interrupt gets generated when a zero length packet is transmitted due to un-availability of data in the TXFifo.
12	W1C	0x0	BbleErrIntrpt BbleErr (Babble Error) interrupt The core generates this interrupt when babble is received for the endpoint.
11	W1C	0x0	PktDrpSts Packet Dropped Status This bit indicates to the application that an ISOC OUT packet has been dropped. This bit does not have an associated mask bit and does not generate an interrupt. Dependency: This bit is valid in non Scatter/Gather DMA mode when periodic transfer interrupt feature is selected.
10	RO	0x0	reserved
9	W1C	0x0	BNAIntr BNA (Buffer Not Available) Interrupt The core generates this interrupt when the descriptor accessed is not ready for the Core to process, such as Host busy or DMA done Dependency: This bit is valid only when Scatter/Gather DMA mode is enabled.
8	W1C	0x0	TxfifoUndrn FIFO Underrun Applies to IN endpoints only. The core generates this interrupt when it detects a transmit FIFO underrun condition for this endpoint. Dependency: This interrupt is valid only when both of the following conditions are true: Parameter OTG_EN_DED_TX_FIFO==1; Thresholding is enabled; OUT Packet Error(OutPktErr). Applies to OUT endpoints only. This interrupt is asserted when the core detects an overflow or a CRC error for an OUT packet. Dependency: This interrupt is valid only when both of the following conditions are true: Parameter OTG_EN_DED_TX_FIFO==1; Thresholding is enabled.

Bit	Attr	Reset Value	Description
7	W1C	0x0	TxFEmp Transmit FIFO Empty This bit is valid only for IN Endpoints.This interrupt is asserted when the TxFIFO for this endpoint is either half or completely empty. The half or completely empty status is determined by the TxFIFO Empty Level bit in the Core AHB Configuration register (GAHBCFG.NPTxFEmpLvl)).
6	W1C	0x0	INEPNakEff IN Endpoint NAK Effective Applies to periodic IN endpoints only. This bit can be cleared when the application clears the IN endpoint NAK by writing to DIEPCTLn.CNAK. This interrupt indicates that the core has sampled the NAK bit set (either by the application or by the core). The interrupt indicates that the IN endpoint NAK bit set by the application has taken effect in the core. This interrupt does not guarantee that a NAK handshake is sent on the USB. A STALL bit takes priority over a NAK bit. This bit is applicable only when the endpoint is enabled. Back-to-Back SETUP Packets Received (Back2BackSETup) Applies to Control OUT endpoints only. This bit indicates that the core has received more than three back-to-back SETUP packets for this particular endpoint.
5	W1C	0x0	INTknEPMis IN Token Received with EP Mismatch Applies to non-periodic IN endpoints only. Indicates that the data in the top of the non-periodic TxFIFO belongs to an endpoint other than the one for which the IN token was received. This interrupt is asserted on the endpoint for which the IN token was received. Status Phase Received For Control Write (StsPhseRcvd) This interrupt is valid only for Control OUT endpoints and only in Scatter Gather DMA mode. This interrupt is generated only after the core has transferred all the data that the host has sent during the data phase of a control write transfer, to the system memory buffer. The interrupt indicates to the application that the host has switched from data phase to the status phase of a Control Write transfer. The application can use this interrupt to ACK or STALL the Status phase, after it has decoded the data phase. This is applicable only in case of Scatter Gather DMA mode.

Bit	Attr	Reset Value	Description
4	W1C	0x0	INTknTXFEmp IN Token Received When TxFIFO is Empty Indicates that an IN token was received when the associated TxFIFO periodic/nonperiodic) was empty. This interrupt is asserted on the endpoint for which the IN token was received. OUT Token Received When Endpoint Disabled (OUTTknEPdis) Indicates that an OUT token was received when the endpoint was not yet enabled. This interrupt is asserted on the endpoint for which the OUT token was received.
3	W1C	0x0	TimeOUT Timeout Condition In shared TX FIFO mode, applies to non-isochronous IN endpoints only. In dedicated FIFO mode, applies only to Control IN endpoints. In Scatter/Gather DMA mode, the TimeOUT interrupt is not asserted. Indicates that the core has detected a timeout condition on the USB for the last IN token on this endpoint. SETUP Phase Done (SetUp) Applies to control OUT endpoints only. Indicates that the SETUP phase for the control endpoint is complete and no more back-to-back SETUP packets were received for the current control transfer. On this interrupt, the application can decode the received SETUP data packet.
2	W1C	0x0	AHBErr AHB Error Applies to IN and OUT endpoints. This is generated only in Internal DMA mode when there is an AHB error during an AHB read/write. The application can read the corresponding endpoint DMA address register to get the error address.
1	W1C	0x0	EPDisbld Endpoint Disabled Interrupt Applies to IN and OUT endpoints. This bit indicates that the endpoint is disabled per the application's request.
0	W1C	0x0	XferCompl Transfer Completed Interrupt Applies to IN and OUT endpoints. When Scatter/Gather DMA mode is enabled: For IN endpoint this field indicates that the requested data from the descriptor is moved from external system memory to internal FIFO. For OUT endpoint this field indicates that the requested data from the internal FIFO is moved to external system memory. This interrupt is generated only when the corresponding endpoint descriptor is closed, and the IOC bit for the corresponding descriptor is set. When Scatter/Gather DMA mode is disabled, this field indicates that the programmed transfer is complete on the AHB as well as on the USB, for this endpoint.

USBOTG_DIEPTSIZn
Address: Operational Base + offset (0x0910)
Device endpoint n transfer size register

Bit	Attr	Reset Value	Description
31	RO	0x0	reserved
30:29	RW	0x0	MC Multi Count Applies to IN endpoints only. For periodic IN endpoints, this field indicates the number of packets that must be transmitted per microframe on the USB. The core uses this field to calculate the data PID for isochronous IN endpoints. 2'b01: 1 packet 2'b10: 2 packets 2'b11: 3 packets For non-periodic IN endpoints, this field is valid only in Internal DMA mode. It specifies the number of packets the core must fetch for an IN endpoint before it switches to the endpoint pointed to by the Next Endpoint field of the Device Endpoint-n Control register (DIEPCTLn.NextEp). Received Data PID (RxDPID) Applies to isochronous OUT endpoints only. This is the data PID received in the last packet for this endpoint. 2'b00: DATA0 2'b01: DATA2 2'b10: DATA1 2'b11: MDATA SETUP Packet Count (SUPCnt).Applies to control OUT Endpoints only. This field specifies the number of back-to-back SETUP data packets the endpoint can receive. 2'b01: 1 packet 2'b10: 2 packets 2'b11: 3 packets
28:19	RW	0x000	PktCnt Packet Count Indicates the total number of USB packets that constitute the Transfer Size amount of data for this endpoint. IN Endpoints: This field is decremented every time a packet (maximum size or short packet) is read from the TxFIFO. OUT Endpoints: This field is decremented every time a packet (maximum size or short packet) is written to the RxFIFO.

Bit	Attr	Reset Value	Description
18:0	RW	0x00000	XferSize Transfer Size This field contains the transfer size in bytes for the current endpoint. The core only interrupts the application after it has exhausted the transfer size amount of data. The transfer size can be set to the maximum packet size of the endpoint, to be interrupted at the end of each packet. IN Endpoints: The core decrements this field every time a packet from the external memory is written to the TxFIFO. OUT Endpoints: The core decrements this field every time a packet is read from the RxFIFO and written to the external memory.

USBOTG_DIEPDMAn
Address: Operational Base + offset (0x0914)
Device endpoint-n DMA Address Register

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	DMAAddr DMA Address Holds the start address of the external memory for storing or fetching endpoint data. Note: For control endpoints, this field stores control OUT data packets as well as SETUP transaction data packets. When more than three SETUP packets are received back-to-back, the SETUP data packet in the memory is overwritten. This register is incremented on every AHB transaction. The application can give only a DWORD-aligned address. When Scatter/Gather DMA mode is not enabled, the application programs the start address value in this field. When Scatter/Gather DMA mode is enabled, this field indicates the base pointer for the descriptor list.

USBOTG_DTXFSTSn
Address: Operational Base + offset (0x0918)
Device IN endpoint transmit FIFO status register

Bit	Attr	Reset Value	Description
31:16	RO	0x0	reserved

Bit	Attr	Reset Value	Description
15:0	RW	0x0000	INEPTxFSpcAvail IN Endpoint TxFIFO Space Avail Indicates the amount of free space available in the Endpoint TxFIFO.Values are in terms of 32-bit words. 16'h0: Endpoint TxFIFO is full 16'h1: 1 word available 16'h2: 2 words available 16'hn: n words available (where 0 . n . 32,768) 16'h8000: 32,768 words available Others: Reserved

USBOTG_DIEPDMABn

Address: Operational Base + offset (0x091c)

Device endpoint-n DMA buffer address register

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	DMABufferAddr DMA Buffer Address Holds the current buffer address.This register is updated as and when the data transfer for the corresponding end point is in progress. This register is present only in Scatter/Gather DMA mode. Otherwise this field is reserved.

USBOTG_DIEPCTLn

Address: Operational Base + offset (0x0920)

Device endpoint-n control register

Bit	Attr	Reset Value	Description
31	R/W SC	0x0	EPEna Endpoint Enable Applies to IN and OUT endpoints. When Scatter/Gather DMA mode is enabled, For IN endpoints this bit indicates that the descriptor structure and data buffer with data ready to transmit is setup. For OUT endpoint it indicates that the descriptor structure and data buffer to receive data is setup. When Scatter/Gather DMA mode is enabled-such as for buffer-pointer based DMA mode: For IN endpoints, this bit indicates that data is ready to be transmitted on the endpoint ; For OUT endpoints, this bit indicates that the application has allocated the memory to start receiving data from the USB. The core clears this bit before setting any of the following interrupts on this endpoint: SETUP Phase Done, Endpoint Disabled, Transfer Completed. Note: For control endpoints in DMA mode, this bit must be set to be able to transfer SETUP data packets in memory.

Bit	Attr	Reset Value	Description
30	R/WSC	0x0	EPDis Endpoint Disable Applies to IN and OUT endpoints. The application sets this bit to stop transmitting/receiving data on an endpoint, even before the transfer for that endpoint is complete. The application must wait for the Endpoint Disabled interrupt before treating the endpoint as disabled. The core clears this bit before setting the Endpoint Disabled interrupt. The application must set this bit only if Endpoint Enable is already set for this endpoint.
29	WO	0x0	SetD1PID Set DATA1 PID Applies to interrupt/bulk IN and OUT endpoints only.Writing to this field sets the Endpoint Data PID (DPID) field in this register to DATA1.This field is applicable both for Scatter/Gather DMA mode and non-Scatter/Gather DMA mode. Set Odd (micro)frame (SetOddFr). Applies to isochronous IN and OUTendpoints only. Writing to this field sets the Even/Odd (micro)frame (EO_FrNum) field to odd (micro)frame. This field is not applicable for Scatter/Gather DMA mode.
28	WO	0x0	SetD0PID Set DATA0 PID Applies to interrupt/bulk IN and OUT endpoints only.Writing to this field sets the Endpoint Data PID (DPID) field in this register to DATA0.This field is applicable both for Scatter/Gather DMA mode and non-Scatter/Gather DMA mode. In non-Scatter/Gather DMA mode: Set Even (micro)frame (SetEvenFr) Applies to isochronous IN and OUT endpoints only. Writing to this field sets the Even/Odd (micro)frame (EO_FrNum) field to even (micro)frame. When Scatter/Gather DMA mode is enabled, this field is reserved. The frame number in which to send data is in the transmit descriptor structure. The frame in which to receive data is updated in received descriptor structure.
27	WO	0x0	SNAK Set NAK Applies to IN and OUT endpoints. A write to this bit sets the NAK bit for the endpoint. Using this bit, the application can control the transmission of NAK handshakes on an endpoint. The core can also set this bit for OUT endpoints on a Transfer Completed interrupt, or after a SETUP is received on the endpoint.
26	WO	0x0	CNAK Clear NAK Applies to IN and OUT endpoints. A write to this bit clears the NAK bit for the endpoint.

Bit	Attr	Reset Value	Description
25:22	RW	0x0	TxFNum TxFIFO Number Shared FIFO Operation: non-periodic endpoints must set this bit to zero. Periodic endpoints must map this to the corresponding Periodic TxFIFO number. 4'h0: Non-Periodic TxFIFO; Others: Specified Periodic TxFIFO number. Note: An interrupt IN endpoint can be configured as a non-periodic endpoint for applications such as mass storage. The core treats an IN endpoint as a non-periodic endpoint if the TxFNum field is set to 0. Otherwise, a separate periodic FIFO must be allocated for an interrupt IN endpoint, and the number of this FIFO must be programmed into the TxFNum field. Configuring an interrupt IN endpoint as a non-periodic endpoint saves the extra periodic FIFO area. Dedicated FIFO Operation: these bits specify the FIFO number associated with this endpoint. Each active IN endpoint must be programmed to a separate FIFO number. This field is valid only for IN endpoints.
21	RW	0x0	Stall STALL Handshake Applies to non-control, non-isochronous IN and OUT endpoints only. The application sets this bit to stall all tokens from the USB host to this endpoint. If a NAK bit, Global Non-periodic IN NAK, or Global OUT NAK is set along with this bit, the STALL bit takes priority. Only the application can clear this bit, never the core. Applies to control endpoints only. The application can only set this bit, and the core clears it, when a SETUP token is received for this endpoint. If a NAK bit, Global Non-periodic IN NAK, or Global OUT NAK is set along with this bit, the STALL bit takes priority. Irrespective of this bit's setting, the core always responds to SETUP data packets with an ACK handshake.
20	RW	0x0	Snp Snoop Mode Applies to OUT endpoints only. This bit configures the endpoint to Snoop mode. In Snoop mode, the core does not check the correctness of OUT packets before transferring them to application memory.
19:18	RW	0x0	EPType Endpoint Type Applies to IN and OUT endpoints. This is the transfer type supported by this logical endpoint. 2'b00: Control 2'b01: Isochronous 2'b10: Bulk 2'b11: Interrupt

Bit	Attr	Reset Value	Description
17	RO	0x0	NAKSts NAK Status Applies to IN and OUT endpoints. Indicates the following: 1'b0: The core is transmitting non-NAK handshakes based on the FIFO status. 1'b1: The core is transmitting NAK handshakes on this endpoint. When either the application or the core sets this bit: The core stops receiving any data on an OUT endpoint, even if there is space in the RxFIFO to accommodate the incoming packet. For non-isochronous IN endpoints: The core stops transmitting any data on an IN endpoint, even if there data is available in the TxFIFO. For isochronous IN endpoints: The core sends out a zero-length data packet, even if there data is available in the TxFIFO. Irrespective of this bit's setting, the core always responds to SETUP data packets with an ACK handshake.
16	RO	0x0	DPID Endpoint Data PID Applies to interrupt/bulk IN and OUT endpoints only. Contains the PID of the packet to be received or transmitted on this endpoint. The application must program the PID of the first packet to be received or transmitted on this endpoint, after the endpoint is activated. The applications use the SetD1PID and SetD0PID fields of this register to program either DATA0 or DATA1 PID. 1'b0: DATA0 1'b1: DATA1 This field is applicable both for Scatter/Gather DMA mode and non-Scatter/Gather DMA mode.Even/Odd (Micro)Frame (EO_FrNum) In non-Scatter/Gather DMA mode: Applies to isochronous IN and OUT endpoints only.Indicates the (micro) rame number in which the core transmits/receives isochronous data for this endpoint. The application must program the even/odd (micro) frame number in which it intends to transmit/receive isochronous data for this endpoint using the SetEvnFr and SetOddFr fields in this register. 1'b0: Even (micro)frame 1'b1: Odd (micro)frame When Scatter/Gather DMA mode is enabled, this field is reserved. The frame number in which to send data is provided in the transmit descriptor structure. The frame in which data is received is updated in receive descriptor structure.

Bit	Attr	Reset Value	Description
15	R/WSC	0x0	USBActEP USB Active Endpoint Applies to IN and OUT endpoints.Indicates whether this endpoint is active in the current configuration and interface. The core clears this bit for all endpoints (other than EP 0) after detecting a USB reset. After receiving the SetConfiguration and SetInterface commands, the application must program endpoint registers accordingly and set this bit.
14:11	RW	0x0	NextEp Next Endpoint Applies to non-periodic IN endpoints only.Indicates the endpoint number to be fetched after the data for the current endpoint is fetched. The core can access this field, even when the Endpoint Enable (EPEna) bit is low. This field is not valid in Slave mode operation.Note: This field is valid only for Shared FIFO operations.
10:0	RW	0x000	MPS Maximum Packet Size Applies to IN and OUT endpoints. The application must program this field with the maximum packet size for the current logical endpoint. This value is in bytes.

USBOTG_DOEPCTL0
Address: Operational Base + offset (0x0b00)
Device control OUT endpoint 0 control register

Bit	Attr	Reset Value	Description
31	R/WSC	0x0	EPEna Endpoint Enable When Scatter/Gather DMA mode is enabled, for OUT endpoints this bit indicates that the descriptor structure and data buffer to receive data is setup. When Scatter/Gather DMA mode is disabled?such as for buffer-pointer based DMA mode)-this bit indicates that the application has allocated the memory to start receiving data from the USB. The core clears this bit before setting any of the following interrupts on this endpoint: SETUP Phase Done, Endpoint Disabled, Transfer Completed. *Note: In DMA mode, this bit must be set for the core to transfer SETUP data packets into memory.*
30	WO	0x0	EPDis Endpoint Disable The application cannot disable control OUT endpoint 0.
29:28	RO	0x0	reserved

Bit	Attr	Reset Value	Description
27	WO	0x0	SNAK Set NAK A write to this bit sets the NAK bit for the endpoint. Using this bit, the application can control the transmission of NAK handshakes on an endpoint. The core can also set bit on a Transfer Completed interrupt, or after a SETUP is received on the endpoint.
26	WO	0x0	CNAK Clear NAK A write to this bit clears the NAK bit for the endpoint.
25:22	RO	0x0	reserved
21	R/WSC	0x0	Stall STALL Handshake The application can only set this bit, and the core clears it, when a SETUP token is received for this endpoint. If a NAK bit or Global OUT NAK is set along with this bit, the STALL bit takes priority. Irrespective of this bit's setting, the core always responds to SETUP data packets with an ACK handshake.
20	RW	0x0	Snp Snoop Mode This bit configures the endpoint to Snoop mode. In Snoop mode, the core does not check the correctness of OUT packets before transferring them to application memory.
19:18	RO	0x0	EPType Endpoint Type Hardcoded to 2'b00 for control.
17	RO	0x0	NAKSts NAK Status Indicates the following: 1'b0: The core is transmitting non-NAK handshakes based on the FIFO status. 1'b1: The core is transmitting NAK handshakes on this endpoint. When either the application or the core sets this bit, the core stops receiving data, even if there is space in the RxFIFO to accommodate the incoming packet. Irrespective of this bit setting, the core always responds to SETUP data packets with an ACK handshake.
16	RO	0x0	reserved
15	RO	0x0	USBActEP USB Active Endpoint This bit is always set to 1, indicating that a control endpoint 0 is always active in all configurations and interfaces.
14:2	RO	0x0	reserved

Bit	Attr	Reset Value	Description
1:0	RO	0x0	MPS Maximum Packet Size The maximum packet size for control OUT endpoint 0 is the same as what is programmed in control IN Endpoint 0. 2'b00: 64 bytes 2'b01: 32 bytes 2'b10: 16 bytes 2'b11: 8 bytes

USBOTG_DOEPINTn

Address: Operational Base + offset (0x0b08)

Device endpoint-n control register

Bit	Attr	Reset Value	Description
31:15	RO	0x0	reserved
14	W1C	0x0	NYETIntrpt NYET interrupt The core generates this interrupt when a NYET response is transmitted for a non isochronous OUT endpoint.
13	W1C	0x0	NAKIntrpt NAK interrupt The core generates this interrupt when a NAK is transmitted or received by the device.In case of isochronous IN endpoints the interrupt gets generated when a zero length packet is transmitted due to un-availability of data in the TXFifo.
12	W1C	0x0	BbleErrIntrpt BbleErr (Babble Error) interrupt The core generates this interrupt when babble is received for the endpoint.
11	W1C	0x0	PktDrpSts Packet Dropped Status This bit indicates to the application that an ISOC OUT packet has been dropped. This bit does not have an associated mask bit and does not generate an interrupt. Dependency: This bit is valid in non Scatter/Gather DMA mode when periodic transfer interrupt feature is selected.
10	RO	0x0	reserved
9	W1C	0x0	BNAIntr BNA (Buffer Not Available) Interrupt The core generates this interrupt when the descriptor accessed is not ready for the Core to process, such as Host busy or DMA done. Dependency: This bit is valid only when Scatter/Gather DMA mode is enabled.

Bit	Attr	Reset Value	Description
8	W1C	0x0	TxfifoUndrn FIFO Underrun Applies to IN endpoints only. The core generates this interrupt when it detects a transmit FIFO underrun condition for this endpoint. Dependency: This interrupt is valid only when both of the following conditions are true: Parameter OTG_EN_DED_TX_FIFO==1, Thresholding is enabled, OUT Packet Error (OutPktErr). Applies to OUT endpoints only. This interrupt is asserted when the core detects an overflow or a CRC error for an OUT packet. Dependency: This interrupt is valid only when both of the following conditions are true: Parameter OTG_EN_DED_TX_FIFO==1, Thresholding is enabled.
7	W1C	0x0	TxFEmp Transmit FIFO Empty This bit is valid only for IN Endpoints. This interrupt is asserted when the TxFIFO for this endpoint is either half or completely empty. The half or completely empty status is determined by the TxFIFO Empty Level bit in the Core AHB Configuration register(GAHBCFG.NPTxFEmpLvl)).
6	W1C	0x0	INEPNakEff IN Endpoint NAK Effective Applies to periodic IN endpoints only. This bit can be cleared when the application clears the IN endpoint NAK by writing to DIEPCTLn.CNAK. This interrupt indicates that the core has sampled the NAK bit set (either by the application or by the core). The interrupt indicates that the IN endpoint NAK bit set by the application has taken effect in the core. This interrupt does not guarantee that a NAK handshake is sent on the USB. A STALL bit takes priority over a NAK bit. This bit is applicable only when the endpoint is enabled. Back-to-Back SETUP Packets Received (Back2BackSETup) Applies to Control OUT endpoints only. This bit indicates that the core has received more than three back-to-back SETUP packets for this particular endpoint.
5	W1C	0x0	INTknEPMis IN Token Received with EP Mismatch Applies to non-periodic IN endpoints only. Indicates that the data in the top of the non-periodic TxFIFO belongs to an endpoint other than the one for which the IN token was received. This interrupt is asserted on the endpoint for which the IN token was received. Status Phase Received For Control Write (StsPhseRcvd) This interrupt is valid only for Control OUT endpoints and only in Scatter Gather DMA mode.

Bit	Attr	Reset Value	Description
4	W1C	0x0	INTknTXFEmp IN Token Received When TxFIFO is Empty Indicates that an IN token was received when the associated TxFIFO periodic/ nonperiodic) was empty. This interrupt is asserted on the endpoint for which the IN token was received. OUT Token Received When Endpoint Disabled (OUTTknEPdis) Indicates that an OUT token was received when the endpoint was not yet enabled. This interrupt is asserted on the endpoint for which the OUT token was received.
3	W1C	0x0	TimeOUT Timeout Condition In shared TX FIFO mode, applies to non-isochronous IN endpoints only. In dedicated FIFO mode, applies only to Control IN endpoints. In Scatter/Gather DMA mode, the TimeOUT interrupt is not asserted.Indicates that the core has detected a timeout condition on the USB for the last IN token on this endpoint. SETUP Phase Done (SetUp). Applies to control OUT endpoints only. Indicates that the SETUP phase for the control endpoint is complete and no more back-to-back SETUP packets were received for the current control transfer. On this interrupt, the application can decode the received SETUP data packet.
2	W1C	0x0	AHBErr AHB Error Applies to IN and OUT endpoints. This is generated only in Internal DMA mode when there is an AHB error during an AHB read/write. The application can read the corresponding endpoint DMA address register to get the error address.
1	W1C	0x0	EPDisbld Endpoint Disabled Interrupt Applies to IN and OUT endpoints. This bit indicates that the endpoint is disabled per the application's request.
0	W1C	0x0	XferCompl Transfer Completed Interrupt Applies to IN and OUT endpoints. When Scatter/Gather DMA mode is enabled. For IN endpoint this field indicates that the requested data from the descriptor is moved from external system memory to internal FIFO. For OUT endpoint this field indicates that the requested data from the internal FIFO is moved to external system memory. This interrupt is generated only when the corresponding endpoint descriptor is closed, and the IOC bit for the corresponding descriptor is set. When Scatter/Gather DMA mode is disabled, this field indicates that the programmed transfer is complete on the AHB as well as on the USB, for this endpoint.

USBOTG_DOEPTSIZn

Address: Operational Base + offset (0x0b10)

Device endpoint n transfer size register

Bit	Attr	Reset Value	Description
31	RO	0x0	reserved
30:29	RW	0x0	MC Multi Count Applies to IN endpoints only. For periodic IN endpoints, this field indicates the number of packets that must be transmitted per microframe on the USB. The core uses this field to calculate the data PID for isochronous IN endpoints. 2'b01: 1 packet 2'b10: 2 packets 2'b11: 3 packets For non-periodic IN endpoints, this field is valid only in Internal DMA mode. It specifies the number of packets the core must fetch for an IN endpoint before it switches to the endpoint pointed to by the Next Endpoint field of the Device Endpoint-n Control register (DIEPCTLn.NextEp). Received Data PID (RxDPID) Applies to isochronous OUT endpoints only. This is the data PID received in the last packet for this endpoint. 2'b00: DATA0 2'b01: DATA2 2'b10: DATA1 2'b11: MDATA SETUP Packet Count (SUPCnt).Applies to control OUT Endpoints only. This field specifies the number of back-to-back SETUP data packets the endpoint can receive. 2'b01: 1 packet 2'b10: 2 packets 2'b11: 3 packets
28:19	RW	0x000	PktCnt Packet Count Indicates the total number of USB packets that constitute the Transfer Size amount of data for this endpoint. IN Endpoints: This field is decremented every time a packet (maximum size or short packet) is read from the TxFIFO. OUT Endpoints: This field is decremented every time a packet (maximum size or short packet) is written to the RxFIFO.

Bit	Attr	Reset Value	Description
18:0	RW	0x00000	XferSize Transfer Size This field contains the transfer size in bytes for the current endpoint. The core only interrupts the application after it has exhausted the transfer size amount of data. The transfer size can be set to the maximum packet size of the endpoint, to be interrupted at the end of each packet. IN Endpoints: The core decrements this field every time a packet from the external memory is written to the TxFIFO. OUT Endpoints: The core decrements this field every time a packet is read from the RxFIFO and written to the external memory.

USBOTG_DOEPDMAn
Address: Operational Base + offset (0x0b14)
Device Endpoint-n DMA Address Register

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	DMAAddr DMA Address Holds the start address of the external memory for storing or fetching endpoint data. Note: For control endpoints, this field stores control OUT data packets as well as SETUP transaction data packets. When more than three SETUP packets are received back-to-back, the SETUP data packet in the memory is overwritten. This register is incremented on every AHB transaction. The application can give only a DWORD-aligned address. When Scatter/Gather DMA mode is not enabled, the application programs the start address value in this field. When Scatter/Gather DMA mode is enabled, this field indicates the base pointer for the descriptor list.

USBOTG_DOEPDMABn
Address: Operational Base + offset (0x0b1c)
Device endpoint-n DMA buffer address register

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	DMABufferAddr DMA Buffer Address Holds the current buffer address. This register is updated as and when the data transfer for the corresponding end point is in progress. This register is present only in Scatter/Gather DMA mode. Otherwise this field is reserved.

USBOTG_DOEPCTLn
Address: Operational Base + offset (0x0b20)

Device endpoint-n control register

Bit	Attr	Reset Value	Description
31	R/WSC	0x0	EPEna Endpoint Enable Applies to IN and OUT endpoints. When Scatter/Gather DMA mode is enabled, For IN endpoints this bit indicates that the descriptor structure and data buffer with data ready to transmit is setup. For OUT endpoint it indicates that the descriptor structure and data buffer to receive data is setup. When Scatter/Gather DMA mode is enabled-such as for buffer-pointer based DMA mode: For IN endpoints, this bit indicates that data is ready to be transmitted on the endpoint; For OUT endpoints, this bit indicates that the application has allocated the memory to start receiving data from the USB. The core clears this bit before setting any of the following interrupts on this endpoint: SETUP Phase Done, Endpoint Disabled, Transfer Completed. Note: For control endpoints in DMA mode, this bit must be set to be able to transfer SETUP data packets in memory.
30	R/WSC	0x0	EPDis Endpoint Disable Applies to IN and OUT endpoints. The application sets this bit to stop transmitting/receiving data on an endpoint, even before the transfer for that endpoint is complete. The application must wait for the Endpoint Disabled interrupt before treating the endpoint as disabled. The core clears this bit before setting the Endpoint Disabled interrupt. The application must set this bit only if Endpoint Enable is already set for this endpoint.
29	RO	0x0	SetD1PID Field0001 Abstract Applies to interrupt/bulk IN and OUT endpoints only. Writing to this field sets the Endpoint Data PID (DPID) field in this register to DATA1.This field is applicable both for Scatter/Gather DMA mode and non-Scatter/Gather DMA mode. Set Odd (micro)frame (SetOddFr). Applies to isochronous IN and OUT endpoints only. Writing to this field sets the Even/Odd (micro) frame (EO_FrNum) field to odd (micro) frame. This field is not applicable for Scatter/Gather DMA mode.

Bit	Attr	Reset Value	Description
28	WO	0x0	SetD0PID Set DATA0 PID Applies to interrupt/bulk IN and OUT endpoints only. Writing to this field sets the Endpoint Data PID (DPID) field in this register to DATA0.This field is applicable both for Scatter/Gather DMA mode and non-Scatter/Gather DMA mode. In non-Scatter/Gather DMA mode: Set Even (micro)frame (SetEvenFr) Applies to isochronous IN and OUT endpoints only. Writing to this field sets the Even/Odd (micro) frame (EO_FrNum) field to even (micro) frame. When Scatter/Gather DMA mode is enabled, this field is reserved. The frame number in which to send data is in the transmit descriptor structure. The frame in which to receive data is updated in receive descriptor structure.
27	WO	0x0	SNAK Set NAK Applies to IN and OUT endpoints. A write to this bit sets the NAK bit for the endpoint. Using this bit, the application can control the transmission of NAK handshakes on an endpoint. The core can also set this bit for OUT endpoints on a Transfer Completed interrupt, or after a SETUP is received on the endpoint.
26	WO	0x0	CNAK Clear NAK Applies to IN and OUT endpoints. A write to this bit clears the NAK bit for the endpoint.
25:22	RW	0x0	TxFNum TxFIFO Number Shared FIFO Operation: non-periodic endpoints must set this bit to zero. Periodic endpoints must map this to the corresponding Periodic TxFIFO number. 4'h0: Non-Periodic TxFIFO; Others: Specified Periodic TxFIFO number. Note: An interrupt IN endpoint can be configured as a non-periodic endpoint for applications such as mass storage. The core treats an IN endpoint as a non-periodic endpoint if the TxFNum field is set to 0. Otherwise, a separate periodic FIFO must be allocated for an interrupt IN endpoint, and the number of this FIFO must be programmed into the TxFNum field. Configuring an interrupt IN endpoint as a non-periodic endpoint saves the extra periodic FIFO area. Dedicated FIFO Operation: these bits specify the FIFO number associated with this endpoint. Each active IN endpoint must be programmed to a separate FIFO number. This field is valid only for IN endpoints.

Bit	Attr	Reset Value	Description
21	RW	0x0	Stall STALL Handshake Applies to non-control, non-isochronous IN and OUT endpoints only. The application sets this bit to stall all tokens from the USB host to this endpoint. If a NAK bit, Global Non-periodic IN NAK, or Global OUT NAK is set along with this bit, the STALL bit takes priority. Only the application can clear this bit, never the core. Applies to control endpoints only. The application can only set this bit, and the core clears it, when a SETUP token is received for this endpoint. If a NAK bit, Global Non-periodic IN NAK, or Global OUT NAK is set along with this bit, the STALL bit takes priority. Irrespective of this bit's setting, the core always responds to SETUP data packets with an ACK handshake.
20	RW	0x0	Snp Snoop Mode Applies to OUT endpoints only. This bit configures the endpoint to Snoop mode. In Snoop mode, the core does not check the correctness of OUT packets before transferring them to application memory.
19:18	RW	0x0	EPType Endpoint Type Applies to IN and OUT endpoints. This is the transfer type supported by this logical endpoint. 2'b00: Control 2'b01: Isochronous 2'b10: Bulk 2'b11: Interrupt
17	RO	0x0	NAKSts NAK Status Applies to IN and OUT endpoints. Indicates the following: 1'b0: The core is transmitting non-NAK handshakes based on the FIFO status. 1'b1: The core is transmitting NAK handshakes on this endpoint. When either the application or the core sets this bit: The core stops receiving any data on an OUT endpoint, even if there is space in the RxFIFO to accommodate the incoming packet.

Bit	Attr	Reset Value	Description
16	RO	0x0	DPID Endpoint Data PID Applies to interrupt/bulk IN and OUT endpoints only. Contains the PID of the packet to be received or transmitted on this endpoint. The application must program the PID of the first packet to be received or transmitted on this endpoint, after the endpoint is activated. The applications use the SetD1PID and SetD0PID fields of this register to program either DATA0 or DATA1 PID. 1'b0: DATA0 1'b1: DATA1 This field is applicable both for Scatter/Gather DMA mode and non-Scatter/Gather DMA mode. Even/Odd (Micro)Frame (EO_FrNum). In non-Scatter/Gather DMA mode: Applies to isochronous IN and OUT endpoints only. Indicates the (micro)frame number in which the core transmits/receives isochronous data for this endpoint. The application must program the even/odd (micro) frame number in which it intends to transmit/receive isochronous data for this endpoint using the SetEvnFr and SetOddFr fields in this register. 1'b0: Even (micro)frame 1'b1: Odd (micro)frame When Scatter/Gather DMA mode is enabled, this field is reserved. The frame number in which to send data is provided in the transmit descriptor structure. The frame in which data is received is updated in receive descriptor structure.
15	R/WSC	0x0	USBActEP USB Active Endpoint Applies to IN and OUT endpoints. Indicates whether this endpoint is active in the current configuration and interface. The core clears this bit for all endpoints (other than EP 0) after detecting a USB reset. After receiving the SetConfiguration and SetInterface commands, the application must program endpoint registers accordingly and set this bit.
14:11	RW	0x0	NextEp Next Endpoint Applies to non-periodic IN endpoints only. Indicates the endpoint number to be fetched after the data for the current endpoint is fetched. The core can access this field, even when the Endpoint Enable (EPEna) bit is low. This field is not valid in Slave mode operation. Note: This field is valid only for Shared FIFO operations.
10:0	RW	0x000	MPS Maximum Packet Size Applies to IN and OUT endpoints. The application must program this field with the maximum packet size for the current logical endpoint. This value is in bytes.

USBOTG_PCGCR

Address: Operational Base + offset (0x0b24)

Power and clock gating control register

Bit	Attr	Reset Value	Description
31:14	RW	0x0802e	RestoreValue Restore Value (Applicable only when Hibernation is enabled (OTG_EN_PWROPT=2). Defines port clock select for different speeds. [31] if_dev_mode - 1: Device mode, core restored as device - 0: Host mode, core restored as host [30:29] p2hd_prt_spd (PRT speed) - 00: HS - 01: FS - 10: LS - 11: Reserved [28:27] p2hd_dev_enum_spd (Device enumerated speed) - 00: HS - 01: FS (30/60 MHz clk) - 10: LS - 11: FS (48 MHz clk) [26:20] mac_dev_addr (MAC device address) Device address [19] mac_termselect (Termination selection) - 0: HS_TERM (Program for High Speed) - 1: FS_TERM (Program for Full Speed) [18:17] mac_xcvrselect (Transceiver select) - 00: HS_XCVR (High Speed) - 01: FS_XCVR (Full Speed) - 10: LS_XCVR (Low Speed) - 11: LFS_XCVR (Reserved) [16] sh2pl_prt_ctl[0] - 1: prt_power enabled - 0: prt_power disabled [15:14] prt_clk_sel (Refer prt_clk_sel table)
13	RW	0x0	EssRegRestored Essential Register Values Restored (Applicable only when Hibernation is enabled (OTG_EN_PWROPT=2). When a value of 1 is written to this field, it indicates that register values of essential registers have been restored.
12:10	RO	0x0	reserved

Bit	Attr	Reset Value	Description
9	RO	0x0	RestoreMode Restore Mode (Applicable only when Hibernation is enabled (OTG_EN_PWROPT=2). The application should program this bit to specify the restore mode during RESTORE POINT before programming PCGCCTL.EssRegRest bit is set. Host Mode: 1'b0: Host Initiated Resume, Host Initiated Reset 1'b1: Device Initiated Remote Wake up Device Mode: 1'b0: Device Initiated Remote Wake up 1'b1: Host Initiated Resume, Host Initiated Reset
8	RW	0x0	ResetAfterSusp Reset After Suspend Applicable in Partial power-down mode. In partial power-down mode of operation, this bit needs to be set in host mode before clamp is removed if the host needs to issue reset after suspend. If this bit is not set,then the host issues resume after suspend. This bit is not applicable in device mode and non-partial power-down mode.In Hibernation mode, this bit needs to be set at RESTORE_POINT before PCGCCTL.EssRegRestored is set. In this case, PCGCCTL.restore_mode needs to be set to wait_restore.
7	RO	0x0	L1Suspended Deep Sleep This bit indicates that the PHY is in deep sleep when in L1 state.
6	RO	0x0	PhySleep PHY in Sleep This bit indicates that the PHY is in the Sleep state.
5	RW	0x0	Enbl_L1Gating Enable Sleep Clock Gating When this bit is set, core internal clock gating is enabled in Sleep state if the core cannot assert utmi_l1_suspend_n. When this bit is not set, the PHY clock is not gated in Sleep state.
4	RO	0x0	reserved
3	RW	0x0	RstPdwnModule Reset Power-Down Modules This bit is valid only in Partial Power-Down mode. The application sets this bit when the power is turned off. The application clears this bit after the power is turned on and the PHY clock is up.

Bit	Attr	Reset Value	Description
2	RW	0x0	PwrClmp Power Clamp This bit is valid only in Partial Power-Down mode (OTG_EN_PWROPT = 1). The application sets this bit before the power is turned off to clamp the signals between the power-on modules and the power-off modules. The application clears the bit to disable the clamping before the power is turned on.
1	RW	0x0	GateHclk Gate Hclk The application sets this bit to gate hclk to modules other than the AHB Slave and Master and wakeup logic when the USB is uspended or the session is not valid. The application clears this bit when the USB is resumed or a new session starts.
0	RW	0x0	StopPclk Stop Pclk The application sets this bit to stop the PHY clock (phy_clk) when the USB is suspended, the session is not valid, or the device is disconnected. The application clears this bit when the USB is resumed or a new session starts.

USBOTG_EPBUF0
Address: Operational Base + offset (0x1000)
Device endpoint 0 / host out channel 0 address

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	EPBUF0 From0x1000 to 0x2000, EPBUF for endport0

USBOTG_EPBUF1
Address: Operational Base + offset (0x2000)
Device endpoint 1 / host out channel 1 address

Bit	Attr	Reset Value	Description
31:1	RO	0x0	reserved
0	RW	0x0	EPBUF1 From0x2000 to 0x3000, EPBUF for endport1

USBOTG_EPBUF2
Address: Operational Base + offset (0x3000)
Device endpoint 2 / host out channel 2 address

Bit	Attr	Reset Value	Description
31:1	RO	0x0	reserved
0	RW	0x0	EPBUF2 From0x3000 to 0x4000, EPBUF for endport2

USBOTG_EPBUF3
Address: Operational Base + offset (0x4000)
Device endpoint 3 / host out channel 3 address

Bit	Attr	Reset Value	Description
31:1	RO	0x0	reserved
0	RW	0x0	EPBUF3 From0x4000 to 0x5000, EPBUF for endport3

USBOTG_EPBUF4
Address: Operational Base + offset (0x5000)
Device endpoint 4 / host out channel 4 address

Bit	Attr	Reset Value	Description
31:1	RO	0x0	reserved
0	RW	0x0	EPBUF4 From0x5000 to 0x6000, EPBUF for endport4

USBOTG_EPBUF5
Address: Operational Base + offset (0x6000)
Device endpoint 5 / host out channel 5 address

Bit	Attr	Reset Value	Description
31:1	RO	0x0	reserved
0	RW	0x0	EPBUF5 From0x6000 to 0x7000, EPBUF for endport5

USBOTG_EPBUF6
Address: Operational Base + offset (0x7000)
Device endpoint 6 / host out channel 6 address

Bit	Attr	Reset Value	Description
31:1	RO	0x0	reserved
0	RW	0x0	EPBUF6 From0x7000 to 0x8000, EPBUF for endport6

USBOTG_EPBUF7
Address: Operational Base + offset (0x8000)
Device endpoint 7 / host out channel 7 address

Bit	Attr	Reset Value	Description
31:1	RO	0x0	reserved
0	RW	0x0	EPBUF7 From0x8000 to 0x9000, EPBUF for endport7

13.7 Interface description

Table 13-1 USB OTG 2.0 Interface Description

Module Pin	Direction	Pad Name	Pin mux
USB0PN	A	IO_USB0_PN	-
USBRBIAS	A	IO_USB_RBIAS_0	-

Module Pin	Direction	Pad Name	Pin mux
USB0PP	A	IO_USB0_PP	-
VBUS	A	IO_USB_VBUS_0	-
USB0ID	A	IO_USB0_ID	-
DRVVBUS	D	IO_GPIO3C1	

***Note**: **A**—Analog pad ;**AP**—Analog power; **AG**—Analog ground ;**DP**—Digital power ;**DG**–Digital ground;*

13.8 Application Note

13.8.1 Suspend Mode

When PHY is in suspend state

- COMMONONN = 1'b1, 480M clock invalid
- COMMONONN = 1'b0, 480M clock output available.

Please refer to "Chapter GRF" of part1 for configuration details

13.8.2 Reset a port

Please refer to "Chapter3 CRU" for more details.

13.8.3 Relative GRF Registers

GRF_USBPHY_CON0 ~ GRF_USBPHY_CON15 contain 256 bit registers to configure USB PHY. These bits are used to adjust DP/DM SI.

GRF_UOC0_CON5 contain some bit registers to configure USBOTG Controller. These bits are used to control UTMI+ interface.

GRF_SOC_STATUS0 and GRF_UOC_STATUS0 contain some bit status of USBOTG Controller.

Please refer to "Chapter GRF" of part1 for more details.

Chapter 14 HDMI Tx

14.1 Overview

HDMI TX is fully compliant with HDMI 1.4a specification. It offers a simple implementation for consumer electronics like DVD/player/recorder and camcorder. HDMI TX consists of one HMDI transmitter controller and one HMDI transmitter PHY.

14.1.1 Features

- Very low power operation, less than 60mW in PHY during 1080P HD display
- HDMI 1.4a/b/1.3/1.2/1.1, HDCP 1.4 and DVI 1.0 standard compliant transmitter
- Supports data rate from 25MHz, 1.65bps up to 3.4Gbps over a Single channel HDMI
- Support 3D function defined in HDMI 1.4 a/b spec
- TMDS Tx Drivers with programmable output swing, resister values and pre-emphasis
- Supports all DTV resolutions including 480i/576i/480p/576p/720p/1080i/1080p
- Digital video interface supports a pixel size of 24bits color depth in RGB
- S/PDIF output supports PCM, Dolby Digital, DTS digital audio transmission (32-192kHz Fs) using IEC60958 and IEC 61937
- Multiphase 4MHz fixed bandwidth PLL with low jitter
- DDC Bus I2C master interface at 3.3V
- HDCP encryption and decryption engine contains all the necessary logic to encrypt the incoming audio and video data
- Support HDMI LipSync if needed as additional feature
- Lower power operation with optimal power management feature
- Embedded ESD, scan support logic.
- Library delivered to support all major EDA tools and detailed guide to integrate into pad ring/BGA bumps
- The EDID and CEC function are also supported by HDMI Transmitter Controller
- Optional Monitor Detection supported through Hot Plug
- 3.3V high speed I/O and 1.2V/1.0V core power supply

14.2 Block Diagram

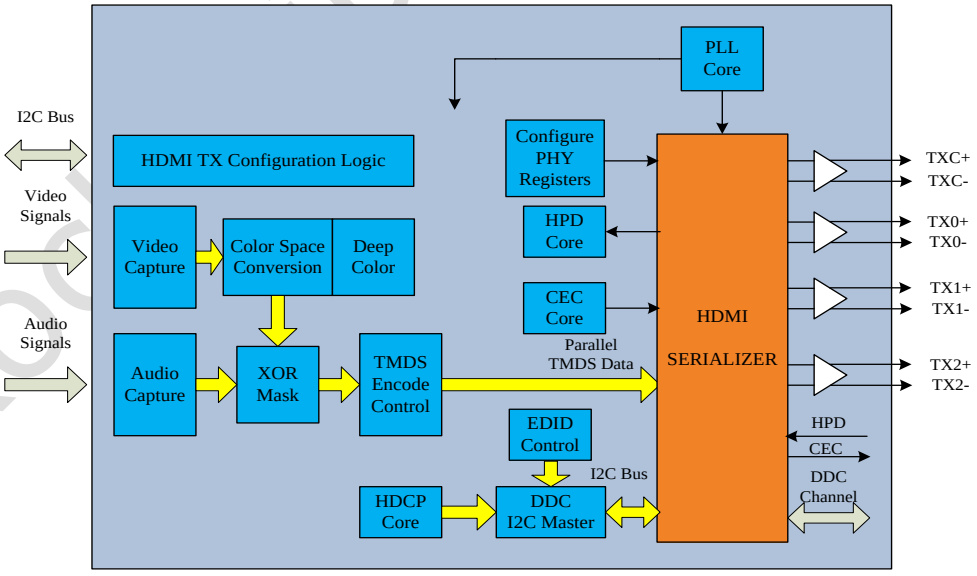


Fig. 14-1 HDMI TX Block Diagram

14.3 Function description

14.3.1 Video Data Processing

The video processing contain video format timings, pixel encodings(RGB to YCbCr, or YCbCr to RGB), colorimetry and corresponding requirements. This function is implemented by

some functional blocks, Video Capture block, Color Space Conversion block, and Deep Color block.

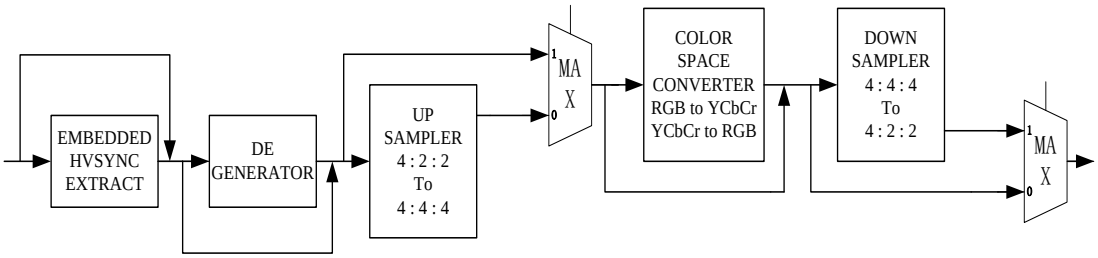


Fig. 14-2 HDMI Video Data Processing

The input video pixels can be encoded in either RGB, YCBCR 4:4:4 or YCBCR 4:2:2 formats by Color Space Conversion block.

The input Video data can have a pixel size of 24, 30, 36 or 48 bits. The deep color block is used to deal with different pixel size. Video at the default 24-bit color depth is carried at a TMDS clock rate equal to the pixel clock rate. Higher color depths are carried using a correspondingly higher TMDS clock rate. HDMI Transmitter support video formats with TMDS rates below 25MHz (e.g. 13.5MHz for 480i/NTSC) that can be transmitted using a pixel-repetition scheme by setting relative registers.

The following interface timing diagram outlines the Video interface signal format. 24 bit data (we also support 36 bit data for deep color) in RGB can be captured by the rising edge of VCLK with 1ns setup time and 1ns hold time requirements. Control signals such DE and VSync/HSync/FSync going with the same timing relationship.

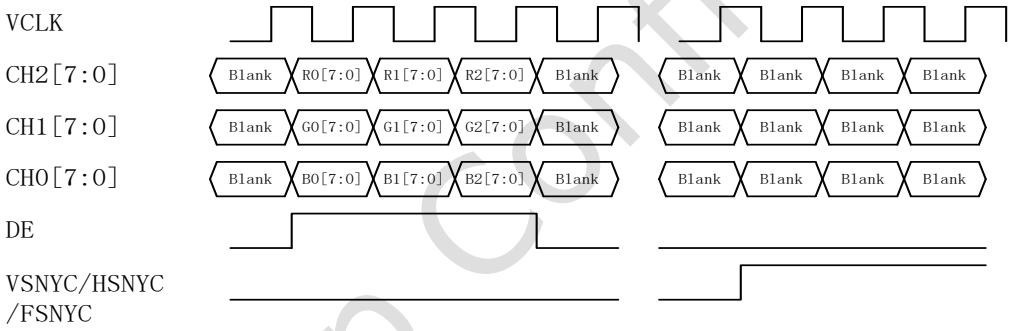


Fig. 14-3 HDMI Video Processing Timing

1. Video Data Capture Logic

HDMI TX support input video data related format table is listed below.

Table 14-1 HDMI Supported Input Video Formats

Color Space	Pixel Encoding	Sync	Channel Width	Pin Nums
RGB	4:4:4	Separate	8	24
RGB	4:4:4	Separate	10	30
RGB	4:4:4	Separate	12	36
YCbCr	4:4:4	Separate	8	24
YCbCr	4:4:4	Separate	10	30
YCbCr	4:4:4	Separate	12	36
YCbCr	4:2:2	Separate	8	16
YCbCr	4:2:2	Separate	10	20
YCbCr	4:2:2	Separate	12	24
YCbCr	4:4:4	Embedded	8	24
YCbCr	4:4:4	Embedded	10	30
YCbCr	4:4:4	Embedded	12	36

Color Space	Pixel Encoding	Sync	Channel Width	Pin Nums
YCbCr	4:2:2	Embedded	8	16
YCbCr	4:2:2	Embedded	10	20
YCbCr	4:2:2	Embedded	12	24

2. Embedded Sync Extraction Module

The module is used to extract Vsync and Hsync signals from input video data stream such as ITU656 format. With setting the relative registers, this functional module can extract correct video sync signals for later processblock using.

3. Data Enable (DE) Generator

HDMI Transmitter has DE signal generator by incoming HSYNCs, VSYNCs and Video clock. External DE is optional and selected by appropriate register settings. This feature is particularly useful when interfacing to MPEG decoders that do not provide a specific DE output signal.

4. Color Space Conversion

HDMI Transmitter Color space conversion (CSC) is available to interface for several MPEG decoders like with YCbCr-only outputs, and to provide full DVI backwards compatibility. The function of this module is to perform color space conversion functionality as listed below.
(1)Convert RGB input Video data to YCbCr Video data.
(2)Convert YCbCr input Video data to RGB Video data.
(3)upsample for YCbCr 4:2:2 to YCbCr 4:4:4
(4)downsample for YCbCr 4:4:4 to YCbCr 4:2:2

14.3.2 Audio Data Processing

The HDMI TX audio process contain audio clock regeneration, placement of audio samples within packets, packet timing control, audio sample rates setting, and channel/speaker assignments. This function is implemented by Audio Capture blocks
The Audio Capture support either SPDIF or four channel I2S input. SPDIF input supports audio sampling rates from 32to 192 KHz. The I2S input supports from 2-channel to 8-channel audio up to 192 KHz.
The scheme of audio processing as shown in the figure below:

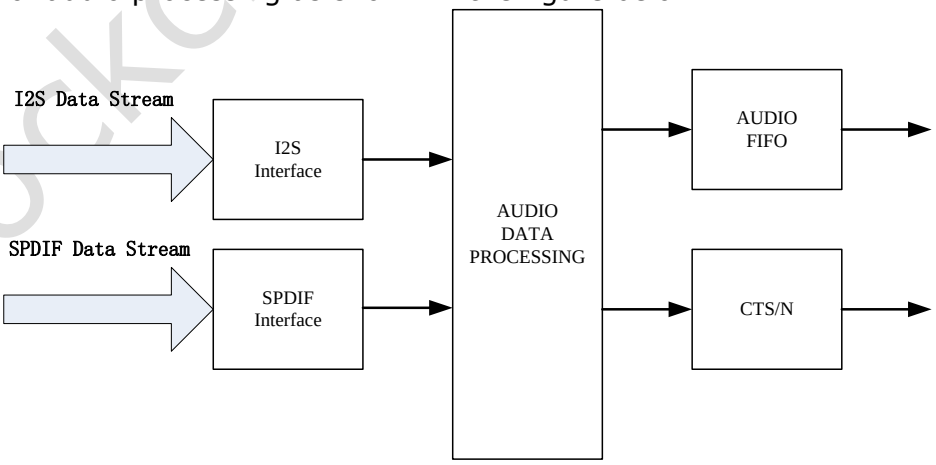


Fig. 14-4 HDMI Audio Data Processing Diagram

1. I2S

The function of this module is to implement I2S audio input feature. The incoming audio stream is captured, processed then transmitted into the TMDS link.Four I2S inputs also allow transmission of DVD-Audio and decoded Dolby Digital to A/V Receivers and high-end

displays. The interface supports from 2-channel to 8-channel audio up to 192 kHz. The I2S pins must also be coherent with mclk. The appropriate registers must be configured to describe the format of audio being input. This information is passed over the HDMI link in the CEA-861D Audio Info (AI) packets. Table shows the I2S 8 channel audio formats that are supported for each of the video formats.

Table 14-2 HDMI TX I2S 2Channel Audio Sampling Frequency at Each Video Format

Video Format	32kHz	44.1kHz	48kHz	88.2kHz	96kHz	176.4kHz	192kHz
720x480p /720x576p	Yes	Yes	Yes	Yes	Yes	Yes	Yes
1440x480i/1440x576i	Yes	Yes	Yes	Yes	Yes	Yes	Yes
720p	Yes	Yes	Yes	Yes	Yes	Yes	Yes
1080i	Yes	Yes	Yes	Yes	Yes	Yes	Yes
1080p	Yes	Yes	Yes	Yes	Yes	Yes	Yes

Table 14-3 HDMI TX I2S 8 Channel Audio Sampling Frequency at Each Video Format

Video Format	32kHz	44.1kHz	48kHz	88.2kHz	96kHz	176.4kHz	192kHz
720x480p /720x576p	Yes	Yes	Yes	No	No	No	No
1440x480i/1440x576i	Yes	Yes	Yes	Yes	No	No	No
720p	Yes	Yes	Yes	Yes	Yes	Yes	Yes
1080i	Yes	Yes	Yes	Yes	Yes	Yes	Yes
1080p	Yes	Yes	Yes	Yes	Yes	Yes	Yes

2. SPDIF

The function of this module is to implement SPDIF audio input feature. The incoming audio stream is captured, processed then transmitted into the TMDS link. SPDIF stream can carry 2-channel uncompressed PCM data (IEC 60958) or a compressed bit stream for multi-channel (IEC 61937) formats. The audio data capture logic forms the audio data into packets in accordance with the HDMI specification. SPDIF input supports audio sampling rates from 32 to 192 KHz. The following shows the SPDIF audio formats that are supported for each of the video formats

Table 14-4 HDMI SPDIF Sampling Frequency at Each Video Format

Video Format	32kHz	44.1kHz	48kHz	88.2kHz	96kHz	176.4kHz	192kHz
720x480p /720x576p	Yes	Yes	Yes	Yes	Yes	No	No
1440x480i/1440x576i	Yes	Yes	Yes	Yes	Yes	No	No
720p	Yes	Yes	Yes	Yes	Yes	Yes	Yes
1080i	Yes	Yes	Yes	Yes	Yes	Yes	Yes
1080p	Yes	Yes	Yes	Yes	Yes	Yes	Yes

3. Audio Sample Clock Capture and Regeneration

Audio data being carried across the HDMI link, which is driven by a TMDS clock running at a rate corresponding to the video pixel rate, does not retain the original audio sample clock. The task of recreating this clock at the Sink is called Audio Clock Regeneration. The HDMI Transmitter determine the fractional relationship between the TMDS clock and an audio reference clock (128 audio sample rate [fs]) and pass the numerator and denominator of that fraction to the HDMI Sink across the HDMI link. The Sinkthen re-create

the audio clock from the TMDS clock by using a clock divider and a clock multiplier.
The exact relationship between the two clocks will be.
128·fS = fTMDS_clock · N / CTS.
The scheme of the Audio Sample Clock Capture and Regeneration as shown below:

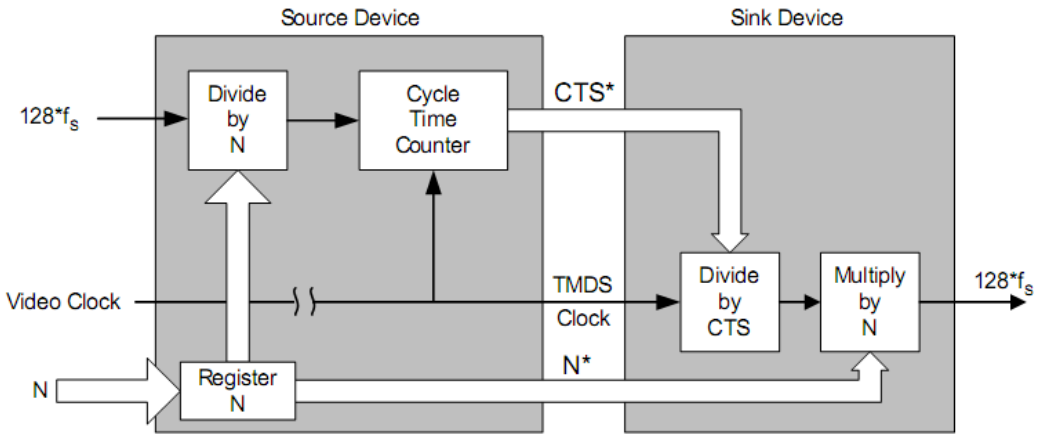


Fig. 14-5 HDMI Audio Clock Regeneration Model

14.3.3 DDC

The DDC functional block is used for configuration and status exchange between the HDMI Source and HDMI Sink. HMDI TransmitterController has I2C Master Interface for DDC transactions. It enables for host controller to read EDID, HDCP authentication by issuing simple register access. The I2C bus speed is limited by DDC specification. DDC bus access frequency can be controlled.

14.3.4 EDID

Extended Display Identification Data (EDID) was created by VESA to enable plug and play capabilities of monitors. This data, which is stored in the sink device, describes video formats that the DTV Monitor is capable of receiving and rendering. The information is supplied to the source device, over the interface, upon the request of the source device. The source device then chooses its output format, taking into account the format of the original video stream and the formats supported by the DTV Monitor. The function of this module is to implement EDID feature.

14.3.5 HDCP

HDMI Transmitter has a capability for HDCP authentication by hardware. The function of this module is to implement HDCP encryption feature. This feature can be turned on or off depending on register setting.

14.3.6 Hot Plug Detect

HDMI Transmitter has a capability for detecting the Sink plug in or plug out, and launch an interrupt and registers state indicating for software controlling.

14.3.7 TMDS encoder

The TMDS encoder converts the 2/4/8 bits data into the 10 bit DC-balanced TMDS data. HDMI TX put the TMDS encoding on the audio /video /aux data received from the HDCP XOR mask. This data is output onto three TMDS differential data lines along with a TMDS differential clock.

14.3.8 CEC

The CEC functional block provides high-level control functions between all of the various audiovisual products in a user's environment through one line.

14.3.9 Optional Lipsync

HDMI sinks and repeaters can now declare audio/video latency information in their EDIDs. HDMI TX can read the audio/video latency information from HDMI sinks, than it may delay audio or video to compensate for latencies in downstream devices.

14.4 Register Description

14.4.1 Register Summary

Name	Offset	Size	Reset Value	Description
HDMI_reg00	0x0000	B	0x67	Register00
HDMI_reg01	0x0004	B	0x01	Register01
HDMI_reg02	0x0008	B	0x30	Register02
HDMI_reg04	0x0010	B	0x08	Register04
HDMI_reg05	0x0014	B	0x00	Register05
HDMI_reg08	0x0020	B	0x00	Register08
HDMI_reg09	0x0024	B	0x00	Register09
HDMI_reg0a	0x0028	B	0x00	Register0a
HDMI_reg0b	0x002c	B	0x00	Register0b
HDMI_reg0c	0x0030	B	0x00	Register0c
HDMI_reg0d	0x0034	B	0x00	Register0d
HDMI_reg0e	0x0038	B	0x00	Register0e
HDMI_reg0f	0x003c	B	0x00	Register0f
HDMI_reg10	0x0040	B	0x00	Register10
HDMI_reg11	0x0044	B	0x00	Register11
HDMI_reg12	0x0048	B	0x00	Register12
HDMI_reg13	0x004c	B	0x00	Register13
HDMI_reg14	0x0050	B	0x00	Register14
HDMI_reg15	0x0054	B	0x00	Register15
HDMI_reg35	0x00d4	B	0x01	Register35
HDMI_reg37	0x00dc	B	0x00	Register37
HDMI_reg38	0x00e0	B	0x3c	Register38
HDMI_reg39	0x00e4	B	0x00	Register39
HDMI_reg3a	0x00e8	B	0x00	Register3a
HDMI_reg3f	0x00fc	B	0x00	Register3f
HDMI_reg40	0x0100	B	0x18	Register40
HDMI_reg41	0x0104	B	0x00	Register41
HDMI_reg45	0x0114	B	0x00	Register45
HDMI_reg46	0x0118	B	0x00	Register46
HDMI_reg47	0x011c	B	0x00	Register47
HDMI_reg4a	0x0128	B	0x00	Register4a
HDMI_reg4b	0x012c	B	0x40	Register4b
HDMI_reg4c	0x0130	B	0x00	Register4c
HDMI_reg4d	0x0134	B	0x00	Register4d
HDMI_reg4e	0x0138	B	0x00	Register4e
HDMI_reg4f	0x013c	B	0x00	Register4f
HDMI_reg50	0x0140	B	0x00	Register50
HDMI_reg52	0x0148	B	0x12	Register52
HDMI_reg53	0x014c	B	0x04	Register53
HDMI_reg57	0x015c	B	0x20	Register57

Name	Offset	Size	Reset Value	Description
HDMI_reg58	0x0160	B	0x00	Register58
HDMI_reg63	0x018c	B	0x26	Register63
HDMI_reg65	0x0194	B	0x00	Register65
HDMI_reg66	0x0198	B	0x00	Register66
HDMI_reg67	0x019c	B	0x00	Register67
HDMI_reg68	0x01a0	B	0x00	Register68
HDMI_reg69	0x01a4	B	0x00	Register69
HDMI_reg6a	0x01a8	B	0x00	Register6a
HDMI_reg6c	0x01b0	B	0x00	Register6c
HDMI_reg95	0x0254	B	0x00	Register95
HDMI_reg96	0x0258	B	0x00	Register96
HDMI_reg97	0x025c	B	0x00	Register97
HDMI_reg98	0x0260	B	0x03	Register98
HDMI_reg9c	0x0270	B	0x00	Register9c
HDMI_reg9e	0x0278	B	0x01	Register9e
HDMI_reg9f	0x027c	B	0x00	Register9f
HDMI_rega0	0x0280	B	0x00	Registera0
HDMI_rega1	0x0284	B	0x00	Registera1
HDMI_rega2	0x0288	B	0x00	Registera2
HDMI_rega3	0x028c	B	0x00	Registera3
HDMI_rega4	0x0290	B	0x00	Registera4
HDMI_rega5	0x0294	B	0x00	Registera5
HDMI_rega6	0x0298	B	0x00	Registera6
HDMI_rega7	0x029c	B	0x00	Registera7
HDMI_rega8	0x02a0	B	0x00	Registera8
HDMI_rega9	0x02a4	B	0x00	Registera9
HDMI_regaa	0x02a8	B	0x00	Registeraa
HDMI_regab	0x02ac	B	0x00	Registerab
HDMI_regac	0x02b0	B	0x00	Registerac
HDMI_regad	0x02b4	B	0x00	Registerad
HDMI_regae	0x02b8	B	0x00	Registerae
HDMI_regaf	0x02bc	B	0x00	Registeraf
HDMI_regb0	0x02c0	B	0x00	Registerb0
HDMI_regb1	0x02c4	B	0x00	Registerb1
HDMI_regb2	0x02c8	B	0x00	Registerb2
HDMI_regb3	0x02cc	B	0x00	Registerb3
HDMI_regb4	0x02d0	B	0x00	Registerb4
HDMI_regb5	0x02d4	B	0x00	Registerb5
HDMI_regb6	0x02d8	B	0x00	Registerb6
HDMI_regb7	0x02dc	B	0x00	Registerb7
HDMI_regb8	0x02e0	B	0x00	Registerb8
HDMI_regb9	0x02e4	B	0x00	Registerb9
HDMI_regba	0x02e8	B	0x00	Registerba

Name	Offset	Size	Reset Value	Description
HDMI_regbb	0x02ec	B	0x00	Registerbb
HDMI_regbc	0x02f0	B	0x00	Registerbc
HDMI_regbd	0x02f4	B	0x00	Registerbd
HDMI_regbe	0x02f8	B	0x00	Registerbe
HDMI_regc0	0x0300	B	0xc0	Registerc0
HDMI_regc1	0x0304	B	0x00	Registerc1
HDMI_regc2	0x0308	B	0x78	Registerc2
HDMI_regc3	0x030c	B	0x00	Registerc3
HDMI_regc4	0x0310	B	0x00	Registerc4
HDMI_regc5	0x0314	B	0x00	Registerc5
HDMI_regc8	0x0320	B	0x00	Registerc8
HDMI_regc9	0x0324	B	0x50	Registerc9
HDMI_regce	0x0338	B	0x01	Registerce
HDMI_regd0	0x0340	B	0x00	Registerd0
HDMI_regd1	0x0344	B	0x00	Registerd1
HDMI_regd2	0x0348	B	0x00	Registerd2
HDMI_regd3	0x034c	B	0x00	Registerd3
HDMI_regd4	0x0350	B	0x03	Registerd4
HDMI_regd5	0x0354	B	0x09	Registerd5
HDMI_regd6	0x0358	B	0x03	Registerd6
HDMI_regd7	0x035c	B	0x00	Registed7
HDMI_regd8	0x0360	B	0xff	Registerd8
HDMI_regd9	0x0364	B	0xff	Registerd9
HDMI_regda	0x0368	B	0x00	Registerda
HDMI_regdb	0x036c	B	0x00	Registerdb
HDMI_regdc	0x0370	B	0xd0	Registerdc
HDMI_regdd	0x0374	B	0x20	Registerdd
HDMI_regde	0x0378	B	0x00	Registerde
HDMI_rege0	0x0380	B	0x23	Registere0
HDMI_rege1	0x0384	B	0x0f	Registere1
HDMI_rege2	0x0388	B	0xaa	Registere2
HDMI_rege3	0x038c	B	0x0f	Registere3
HDMI_rege7	0x039c	B	0x1e	Registere7
HDMI_rege8	0x03a0	B	0x00	Registere8
HDMI_reged	0x03b4	B	0x03	Registered

14.4.2 Detail Register Description

HDMI_reg00

Address: Operational Base + offset (0x0000)

Register00

Bit	Attr	Reset Value	Description
7	RW	0x0	reserved

Bit	Attr	Reset Value	Description
6	RW	0x1	sw_reset_ana Soft reset for analog part 1'b0: reset 1'b1: not
5	RW	0x1	sw_reset_dig Soft reset for digital part 1'b0: reset 1'b1: not
4	RW	0x0	reserved
3	RW	0x0	vclk_inv Vclk invert select 1'b0: not invert 1'b1: invert
2	RW	0x0	reserved
1	RW	0x1	pd_dig System power down for digital function 1'b0: not 1'b1: power down
0	RW	0x1	interrupt_polarity Interrupt polarity 1'b1: Active High 1'b0: Active low

HDMI_reg01
Address: Operational Base + offset (0x0004)
Register01

Bit	Attr	Reset Value	Description
7:4	RW	0x0	reserved

Bit	Attr	Reset Value	Description
3:1	RW	0x0	input_video_fmt Input video format 3'b000: RGB and YCbCr 4:4:4 3'b101: DDR RGB 4:4:4 or YCbCr 4:4:4 3'b110: DDR YCbCr 4:2:2
0	RW	0x1	de_sel DE select 1'b0: internal DE 1'b1: external DE

HDMI_reg02
Address: Operational Base + offset (0x0008)
Register02

Bit	Attr	Reset Value	Description
7:6	RW	0x0	video_output_fmt Video output format 2'b00: RGB 4:4:4 2'b01: YCbCr 4:4:4 2'b10: YCbCr 4:2:2
5:4	RW	0x3	data_width_422 Data width for 4:2:2 input 2'b00: 12 bits 2'b01: 10 bits 2'b11: 8 bits
3:1	RW	0x0	reserved
0	RW	0x0	Video_in_color_space Video input color space 1'b0: RGB 1'b1: YCbCr

HDMI_reg04
Address: Operational Base + offset (0x0010)
Register04

Bit	Attr	Reset Value	Description
7:4	RW	0x0	reserved

Bit	Attr	Reset Value	Description
3	RW	0x1	sof_sel After 1st SOF(the first edge of the Vsync) for external DE sample 1'b0: after SOF 1'b1: Not
2:1	RW	0x0	reserved
0	RW	0x0	csc_en CSC (Color Space Convert) enable. 1'b0: no CSC 1'b1: enable CSC

HDMI_reg05
Address: Operational Base + offset (0x0014)
Register05

Bit	Attr	Reset Value	Description
7	RW	0x0	clear_avmute Clear avmute
6	RW	0x0	set_avmute Set avmute
5:2	RW	0x0	reserved
1	RW	0x0	audio_mute Audio mute
0	RW	0x0	video_black Video black

HDMI_reg08
Address: Operational Base + offset (0x0020)
Register08

Bit	Attr	Reset Value	Description
7:4	RW	0x0	reserved
3	RW	0x0	vs_polarity VSYNC polarity 1'b0: Negative 1'b1: Positive

Bit	Attr	Reset Value	Description
2	RW	0x0	hs_polarity HSYNC polarity 1'b0: Negative 1'b1: Positive
1	RW	0x0	interlace_progressiv_sel Interlace/Progressive 1'b0: Progressive 1'b1: Interlace
0	RW	0x0	ext_video_para_set_en External video parameter setting enable 1'b0: disable 1'b1: enable

HDMI_reg09
Address: Operational Base + offset (0x0024)
Register09

Bit	Attr	Reset Value	Description
7:0	RW	0x00	ext_hs_total_part1 External horizontal total part1. It is not need to be set at normal resolution. Just for special resolution or test purpose.

HDMI_reg0a
Address: Operational Base + offset (0x0028)
Register0a

Bit	Attr	Reset Value	Description
7:5	RW	0x0	reserved
4:0	RW	0x00	ext_hs_total_part2 External horizontal total part2. It is not need to be set at normal resolution. Just for special resolution or test purpose.

HDMI_reg0b
Address: Operational Base + offset (0x002c)
Register0b

Bit	Attr	Reset Value	Description
7:0	RW	0x00	ext_hs_blank_part1 External horizontal blank part1. It is not need to be set at normal resolution. Just for special resolution or test purpose.

HDMI_reg0c
Address: Operational Base + offset (0x0030)
Register0c

Bit	Attr	Reset Value	Description
7:3	RW	0x0	reserved
2:0	RW	0x0	ext_hs_blank_part2 External horizontal blank part2. It is not need to be set at normal resolution. Just for special resolution or test purpose.

HDMI_reg0d
Address: Operational Base + offset (0x0034)
Register0d

Bit	Attr	Reset Value	Description
7:0	RW	0x00	Ext_hsync_part1 External horizontal hsync part1. It is not need to be set at normal resolution. Just for special resolution or test purpose.

HDMI_reg0e
Address: Operational Base + offset (0x0038)
Register0e

Bit	Attr	Reset Value	Description
7:2	RW	0x0	reserved
1:0	RW	0x0	Ext_hsync_part2 External horizontal hsync part2. It is not need to be set at normal resolution. Just for special resolution or test purpose.

HDMI_reg0f
Address: Operational Base + offset (0x003c)
Register0f

Bit	Attr	Reset Value	Description
7:0	RW	0x00	ext_hs_sync_part1 External horizontal duration part1. It is not need to be set at normal resolution. Just for special resolution or test purpose

HDMI_reg10
Address: Operational Base + offset (0x0040)
Register10

Bit	Attr	Reset Value	Description
7:2	RW	0x0	reserved
1:0	RW	0x0	ext_hs_sync_part2 External horizontal duration part2. It is not need to be set at normal resolution. Just for special resolution or test purpose

HDMI_reg11
Address: Operational Base + offset (0x0044)
Register11

Bit	Attr	Reset Value	Description
7:0	RW	0x00	ext_vertical_total_part1 External vertical total part1. It is not need to be set at normal resolution. Just for special resolution or test purpose

HDMI_reg12
Address: Operational Base + offset (0x0048)
Register12

Bit	Attr	Reset Value	Description
7:4	RW	0x0	reserved
3:0	RW	0x0	ext_vertical_total_part2 External vertical total part2. It is not need to be set at normal resolution. Just for special resolution or test purpose

HDMI_reg13
Address: Operational Base + offset (0x004c)
Register13

Bit	Attr	Reset Value	Description
7	RW	0x0	reserved

Bit	Attr	Reset Value	Description
6:0	RW	0x00	ext_vertical_blank External vertical blank. It is not need to be set at normal resolution. Just for special resolution or test purpose.

HDMI_reg14
Address: Operational Base + offset (0x0050)
Register14

Bit	Attr	Reset Value	Description
7	RW	0x0	reserved
6:0	RW	0x00	ext_vertical_vsync External vertical vsync. It is not need to be set at normal resolution. Just for special resolution or test purpose.

HDMI_reg15
Address: Operational Base + offset (0x0054)
Register15

Bit	Attr	Reset Value	Description
7:6	RW	0x0	reserved
5:0	RW	0x00	ext_vertical_duration External vertical duration. It is not need to be set at normal resolution. Just for special resolution or test purpose.

HDMI_reg35
Address: Operational Base + offset (0x00d4)
Register35

Bit	Attr	Reset Value	Description
7	RW	0x0	cts__sel CTS source select 1'b0: internal CTS 1'b1: external CTS
6:5	RW	0x0	reserved
4:3	RW	0x0	audio_type_sel Audio type select 2'b00: I2S 2'b01: S/PDIF

Bit	Attr	Reset Value	Description
2	RW	0x0	mclk_sel MCLK select
1:0	RW	0x1	mclk_ratio MCLK ratio 2'b00: 128 fs 2'b01: 256 fs 2'b10: 384 fs 2'b11: 512 fs

HDMI_reg37
Address: Operational Base + offset (0x00dc)
Register37

Bit	Attr	Reset Value	Description
7:4	RW	0x0	reserved
3:0	RW	0x0	sample_freq_sel Sampling frequency for I2S audio 4'b0011: 32 kHz 4'b0000: 44.1 kHz 4'b0010: 48 kHz 4'b1000: 88.2 kHz 4'b1010: 96 kHz 4'b1100: 176.4 kHz 4'b1110: 192 kHz

HDMI_reg38
Address: Operational Base + offset (0x00e0)
Register38

Bit	Attr	Reset Value	Description
7:6	RW	0x0	reserved
5:2	RW	0xf	i2s_sel I2S enable for the four I2S pins 4'b0001: I2S0 4'b0010: I2S1 4'b0100: I2S2 4'b1000: I2S3

Bit	Attr	Reset Value	Description
1:0	RW	0x0	i2s_fmt I2S format 2'b00: standard I2S mode 2'b01: right-justified I2S mode 2'b10: left-justified I2S mode

HDMI_reg39
Address: Operational Base + offset (0x00e4)
Register39

Bit	Attr	Reset Value	Description
7:6	RW	0x0	audio_channel3_input_sel Audio channel3 input 2'b00: I2S3 2'b01: I2S2 2'b10: I2S1 2'b11: I2S0
5:4	RW	0x0	audio_channel2_input_sel Audio channel2 input 2'b00: I2S2 2'b01: I2S1 2'b10: I2S0 2'b11: I2S3
3:2	RW	0x0	audio_channel1_input_sel Audio channel1 input 2'b00: I2S1 2'b01: I2S0 2'b10: I2S3 2'b11: I2S2
1:0	RW	0x0	audio_channel0_input_sel Audio channel0 input 2'b00: I2S0 2'b01: I2S3 2'b10: I2S2 2'b11: I2S1

HDMI_reg3a
Address: Operational Base + offset (0x00e8)
Register3a

Bit	Attr	Reset Value	Description
7	RW	0x0	lr_swap_ch7_8 Left/Right data swap for ch7/8
6	RW	0x0	lr_swap_ch5_6 Left/Right data swap for ch5/6
5	RW	0x0	lr_swap_ch3_4 Left/Right data swap for ch3/4
4	RW	0x0	lr_swap_ch1_2 Left/Right data swap for ch1/2
3:0	RW	0x0	spdif_sample_freq S/PDIF sampling frequency 4'b0011: 32 kHz. 4'b0000: 44.1 kHz. 4'b0010: 48 kHz. 4'b1000: 88.2 kHz. 4'b1010: 96 kHz. 4'b1100: 176.4 kHz. 4'b1110: 192 kHz.

HDMI_reg3f
Address: Operational Base + offset (0x00fc)
Register3f

Bit	Attr	Reset Value	Description
7:4	RW	0x0	reserved
3:0	RW	0x0	ncts_part1 20-bit N-CTS Value for audio clock generation at Sink end. N-CTS=reg3f[3:0],reg40[7:0],reg41[7:0] Use this function please refer to the HDMI specification at "Recommended N and Expected CTS Values"

HDMI_reg40
Address: Operational Base + offset (0x0100)
Register40

Bit	Attr	Reset Value	Description
7:0	RW	0x18	ncts_part2 20-bit N-CTS Value for audio clock generation at Sink end. N-CTS=reg3f[3:0],reg40[7:0],reg41[7:0] Use this function please refer to the HDMI specification at "Recommended N and Expected CTS Values"

HDMI_reg41
Address: Operational Base + offset (0x0104)
Register41

Bit	Attr	Reset Value	Description
7:0	RW	0x00	ncts_part3 20-bit N-CTS Value for audio clock generation at Sink end. N-CTS=reg3f[3:0],reg40[7:0],reg41[7:0] Use this function please refer to the HDMI specification at "Recommended N and Expected CTS Values"

HDMI_reg45
Address: Operational Base + offset (0x0114)
Register45

Bit	Attr	Reset Value	Description
7:4	RW	0x00	reserved
3:0	RW	0x00	cts_part3 When use external CTS (reg0x35[7]=1),then set these three regs. CTS = {reg45[3:0], reg46[7:0], reg47[7:0]}

HDMI_reg46
Address: Operational Base + offset (0x0118)
Register46

Bit	Attr	Reset Value	Description
7:0	RW	0x00	cts_part3 When use external CTS (reg0x35[7]=1),then set these three regs. CTS = {reg45[3:0], reg46[7:0], reg47[7:0]}

HDMI_reg47
Address: Operational Base + offset (0x011c)

Register47

Bit	Attr	Reset Value	Description
7:0	RW	0x00	cts_part3 When use external CTS (reg0x35[7]=1),then set these three regs. CTS = {reg45[3:0], reg46[7:0], reg47[7:0]}

HDMI_reg4a

Address: Operational Base + offset (0x0128)

Register4a

Bit	Attr	Reset Value	Description
7:2	RW	0x0	reserved
1	RO	0x0	ddc_sda_bus_statue DDC SDA bus status
0	RO	0x0	ddc_scl_bus_statue DDC SCL bus status

HDMI_reg4b

Address: Operational Base + offset (0x012c)

Register4b

Bit	Attr	Reset Value	Description
7:0	RW	0x40	ddc_bus_access_freq_lsb DDC bus access frequency (LSB)

HDMI_reg4c

Address: Operational Base + offset (0x0130)

Register4c

Bit	Attr	Reset Value	Description
7:0	RW	0x00	ddc_bus_access_freq_msb DDC bus access frequency (MSB)

HDMI_reg4d

Address: Operational Base + offset (0x0134)

Register4d

Bit	Attr	Reset Value	Description
7:0	RW	0x00	edid_segment_pointer EDID segment pointer

HDMI_reg4e
Address: Operational Base + offset (0x0138)
Register4e

Bit	Attr	Reset Value	Description
7:0	RW	0x00	edid_word_addr EDID word address

HDMI_reg4f
Address: Operational Base + offset (0x013c)
Register4f

Bit	Attr	Reset Value	Description
7:0	RW	0x00	edid_fifo_rd_addr EDID FIFO read address

HDMI_reg50
Address: Operational Base + offset (0x0140)
Register50

Bit	Attr	Reset Value	Description
7:0	RO	0x00	edid_rd_access_window EDID reading access window

HDMI_reg52
Address: Operational Base + offset (0x0148)
Register52

Bit	Attr	Reset Value	Description
7	RW	0x0	authentication_start Authentication start
6	RW	0x0	bksv_not_in_blacklist BKSV is not in the blacklist
5	RW	0x0	bksv_in_blacklist BKSV is in the blacklist
4	RW	0x1	encypte_en current frame encypted en 1'b0: current frame should not be encrypted 1'b1: current frame should be encrypted
3	RW	0x0	authentication_stop Authentication Stop

Bit	Attr	Reset Value	Description
2	RW	0x0	advanced_mode_sel 1'b0: do not use advanced cipher mode 1'b1: use advanced cipher mode
1	RW	0x1	mode_sel mode selection 1'b0: DVI mode 1'b1: HDMI mode
0	RW	0x0	hdcp_reset HDCP reset

HDMI_reg53
Address: Operational Base + offset (0x014c)
Register53

Bit	Attr	Reset Value	Description
7	RW	0x0	disable_127_chk Disable 127 check
6	RW	0x0	skip_bksv_blacklist_chk Skip BKSV blacklist check
5	RW	0x0	pj_chk_en Enable Pj check
4	RW	0x0	disable_dev_num_check Disable device number check
3	RW	0x1	dly_ri_chk Delay Ri check by one clock
2	RW	0x0	use_preset Use preset An value
1:0	RW	0x0	key_comb Key combination

HDMI_reg57
Address: Operational Base + offset (0x015c)
Register57

Bit	Attr	Reset Value	Description
7	RW	0x0	Authenticated_en 1'b0: Authenticated 1'b0: Not authenticated
6	RW	0x0	av_encrypte_en AV content is encrypted enable 1'b0: AV content is not encrypted 1'b1: AV content is encrypted
5	RW	0x1	reserved
4	RW	0x0	HDCP_work 1'b0: HDCP is working 1'b1: HDCP is not working
3	RW	0x0	advanced_cipher_en advanced cipher enable 1'b0: Advanced cipher is not enabled 1'b1: Advanced cipher is enabled
2:0	RW	0x0	reserved

HDMI_reg58
Address: Operational Base + offset (0x0160)
Register58

Bit	Attr	Reset Value	Description
7:0	RW	0x00	rx_bcaps_value RX Bcaps value

HDMI_reg63
Address: Operational Base + offset (0x018c)
Register63

Bit	Attr	Reset Value	Description
7:5	RW	0x00	reserved
4:0	RW	0x06	timer_100ms Registers for timer 100ms

HDMI_reg65
Address: Operational Base + offset (0x0194)
Register65

Bit	Attr	Reset Value	Description
7:4	RW	0x0	reserved
3	RW	0x0	ddc_channel_no_ack DDC channels no acknowledge
2	RW	0x0	pj_mismatch Pj mismatch
1	RW	0x0	ri_mismatch Ri mismatch
0	RW	0x0	bksv_wrong Bksv is wrong

HDMI_reg66
Address: Operational Base + offset (0x0198)
Register66

Bit	Attr	Reset Value	Description
7:0	RW	0x00	Bksv value with least significant byte first

HDMI_reg67
Address: Operational Base + offset (0x019c)
Register67

Bit	Attr	Reset Value	Description
7:0	RW	0x00	Bksv value with least significant byte first

HDMI_reg68
Address: Operational Base + offset (0x01a0)
Register68

Bit	Attr	Reset Value	Description
7:0	RW	0x00	Bksv value with least significant byte first

HDMI_reg69
Address: Operational Base + offset (0x01a4)
Register69

Bit	Attr	Reset Value	Description
7:0	RW	0x00	Bksv value with least significant byte first

HDMI_reg6a
Address: Operational Base + offset (0x01a8)
Register6a

Bit	Attr	Reset Value	Description
7:0	RW	0x00	Bksv value with least significant byte first

HDMI_reg6c
Address: Operational Base + offset (0x01b0)
Register6c

Bit	Attr	Reset Value	Description
7:0	RW	0x00	seed_an An seed for generate An

HDMI_reg95
Address: Operational Base + offset (0x0254)
Register95

Bit	Attr	Reset Value	Description
7:0	RW	0x00	hdcp_fifo_first_wr_byte_addr HDCP KEY FIFO first write byte num address

HDMI_reg96
Address: Operational Base + offset (0x0258)
Register96

Bit	Attr	Reset Value	Description
7:0	RW	0x00	hdcp_fifo_first_rd_byte_addr HDCP KEY FIFO first read byte num address

HDMI_reg97
Address: Operational Base + offset (0x025c)
Register97

Bit	Attr	Reset Value	Description
7:2	RO	0x0	reserved
1	RW	0x0	hdcp_fifo_rd_addr8 HDCP KEY FIFO first read byte num address[8]
0	RW	0x0	hdcp_fifo_wr_addr8 HDCP KEY FIFO first write byte num address[8]

HDMI_reg98
Address: Operational Base + offset (0x0260)
Register98

Bit	Attr	Reset Value	Description
7:0	RW	0x03	hdcp_fifo_wr_rd_addr HDCP KEY FIFO write or read point address

HDMI_reg9c
Address: Operational Base + offset (0x0270)
Register9c

Bit	Attr	Reset Value	Description
7	RW	0x0	general_control_packet General control packet 1'b1:enable 1'b0:disable
6	RW	0x0	mpeg_source_frame_packet MPEG source InfoFrame packet 1'b1:enable 1'b0:disable
5	RW	0x0	source_descriptor_frame_packet Source product descriptor InfoFrame packet 1'b1:enable 1'b0:disable
4	RW	0x0	vendor_specific_frame_packet Vendor specific InfoFrame packet 1'b1:enable 1'b0:disable
3	RW	0x0	gamut_metadata_packet Gamut metadata packet 1'b1:enable 1'b0:disable
2	RW	0x0	isrc1_packet ISRC1 packet 1'b1:enable 1'b0:disable
1	RW	0x0	acp_packet ACP packet 1'b1:enable 1'b0:disable

Bit	Attr	Reset Value	Description
0	RW	0x0	generic_packet Generic packet 1'b1:enable 1'b0:disable

HDMI_reg9e
Address: Operational Base + offset (0x0278)
Register9e

Bit	Attr	Reset Value	Description
7:1	RW	0x0	reserved
0	RW	0x1	auto_checksum_frame Auto checksum for InfoFrame 1'b1:enable 1'b0:disable It enables checksum calculation for any InfoFrame packets by hardware.

HDMI_reg9f
Address: Operational Base + offset (0x027c)
Register9f

Bit	Attr	Reset Value	Description
7:0	RW	0x00	packet_offset packet offset register for 0xa0~0xbe 8'h0: Generic packet 8'h1: ACP packet 8'h2: ISRC1 packet 8'h3: ISRC2 packet 8'h4: Gamut metadata packet 8'h5: Vendor specific InfoFrame 8'h6: AVI InfoFrame 8'h7: Source product descriptor InfoFrame packet 8'h8: Audio InfoFrame packet 8'h9: MPEG source InfoFrame

HDMI_rega0
Address: Operational Base + offset (0x0280)
Registera0

Bit	Attr	Reset Value	Description
7:0	RW	0x00	After configure the packet offset register, these registers can configure any packet shown following. Use this function please refer to the HDMI specification at "Data Island Packet Definitions" Generic packet ACP packet ISRC1 packet ISRC2 packet Gamut metadata packet Vendor specific InfoFrame packet AVI InfoFrame packet Source product descriptor InfoFrame packet Audio InfoFrame packet MPEG source InfoFrame packet

HDMI_rega1
Address: Operational Base + offset (0x0284)
Registera1

Bit	Attr	Reset Value	Description
7:0	RW	0x00	After configure the packet offset register, these registers can configure any packet shown following. Use this function please refer to the HDMI specification at "Data Island Packet Definitions" Generic packet ACP packet ISRC1 packet ISRC2 packet Gamut metadata packet Vendor specific InfoFrame packet AVI InfoFrame packet Source product descriptor InfoFrame packet Audio InfoFrame packet MPEG source InfoFrame packet

HDMI_rega2
Address: Operational Base + offset (0x0288)
Registera2

Bit	Attr	Reset Value	Description
7:0	RW	0x00	After configure the packet offset register, these registers can configure any packet shown following. Use this function please refer to the HDMI specification at "Data Island Packet Definitions" Generic packet ACP packet ISRC1 packet ISRC2 packet Gamut metadata packet Vendor specific InfoFrame packet AVI InfoFrame packet Source product descriptor InfoFrame packet Audio InfoFrame packet MPEG source InfoFrame packet

HDMI_rega3
Address: Operational Base + offset (0x028c)
Registera3

Bit	Attr	Reset Value	Description
7:0	RW	0x00	After configure the packet offset register, these registers can configure any packet shown following. Use this function please refer to the HDMI specification at "Data Island Packet Definitions" Generic packet ACP packet ISRC1 packet ISRC2 packet Gamut metadata packet Vendor specific InfoFrame packet AVI InfoFrame packet Source product descriptor InfoFrame packet Audio InfoFrame packet MPEG source InfoFrame packet

HDMI_rega4
Address: Operational Base + offset (0x0290)
Registera4

Bit	Attr	Reset Value	Description
7:0	RW	0x00	After configure the packet offset register, these registers can configure any packet shown following. Use this function please refer to the HDMI specification at "Data Island Packet Definitions" Generic packet ACP packet ISRC1 packet ISRC2 packet Gamut metadata packet Vendor specific InfoFrame packet AVI InfoFrame packet Source product descriptor InfoFrame packet Audio InfoFrame packet MPEG source InfoFrame packet

HDMI_rega5
Address: Operational Base + offset (0x0294)
Registera5

Bit	Attr	Reset Value	Description
7:0	RW	0x00	After configure the packet offset register, these registers can configure any packet shown following. Use this function please refer to the HDMI specification at "Data Island Packet Definitions" Generic packet ACP packet ISRC1 packet ISRC2 packet Gamut metadata packet Vendor specific InfoFrame packet AVI InfoFrame packet Source product descriptor InfoFrame packet Audio InfoFrame packet MPEG source InfoFrame packet

HDMI_rega6
Address: Operational Base + offset (0x0298)
Registera6

Bit	Attr	Reset Value	Description
7:0	RW	0x00	After configure the packet offset register, these registers can configure any packet shown following. Use this function please refer to the HDMI specification at "Data Island Packet Definitions" Generic packet ACP packet ISRC1 packet ISRC2 packet Gamut metadata packet Vendor specific InfoFrame packet AVI InfoFrame packet Source product descriptor InfoFrame packet Audio InfoFrame packet MPEG source InfoFrame packet

HDMI_rega7
Address: Operational Base + offset (0x029c)
Registera7

Bit	Attr	Reset Value	Description
7:0	RW	0x00	After configure the packet offset register, these registers can configure any packet shown following. Use this function please refer to the HDMI specification at "Data Island Packet Definitions" Generic packet ACP packet ISRC1 packet ISRC2 packet Gamut metadata packet Vendor specific InfoFrame packet AVI InfoFrame packet Source product descriptor InfoFrame packet Audio InfoFrame packet MPEG source InfoFrame packet

HDMI_rega8
Address: Operational Base + offset (0x02a0)
Registera8

Bit	Attr	Reset Value	Description
7:0	RW	0x00	After configure the packet offset register, these registers can configure any packet shown following. Use this function please refer to the HDMI specification at "Data Island Packet Definitions" Generic packet ACP packet ISRC1 packet ISRC2 packet Gamut metadata packet Vendor specific InfoFrame packet AVI InfoFrame packet Source product descriptor InfoFrame packet Audio InfoFrame packet MPEG source InfoFrame packet

HDMI_rega9
Address: Operational Base + offset (0x02a4)
Registera9

Bit	Attr	Reset Value	Description
7:0	RW	0x00	After configure the packet offset register, these registers can configure any packet shown following. Use this function please refer to the HDMI specification at "Data Island Packet Definitions" Generic packet ACP packet ISRC1 packet ISRC2 packet Gamut metadata packet Vendor specific InfoFrame packet AVI InfoFrame packet Source product descriptor InfoFrame packet Audio InfoFrame packet MPEG source InfoFrame packet

HDMI_regaa
Address: Operational Base + offset (0x02a8)
Registeraa

Bit	Attr	Reset Value	Description
7:0	RW	0x00	After configure the packet offset register, these registers can configure any packet shown following. Use this function please refer to the HDMI specification at "Data Island Packet Definitions" Generic packet ACP packet ISRC1 packet ISRC2 packet Gamut metadata packet Vendor specific InfoFrame packet AVI InfoFrame packet Source product descriptor InfoFrame packet Audio InfoFrame packet MPEG source InfoFrame packet

HDMI_regab
Address: Operational Base + offset (0x02ac)
Registerab

Bit	Attr	Reset Value	Description
7:0	RW	0x00	After configure the packet offset register, these registers can configure any packet shown following. Use this function please refer to the HDMI specification at "Data Island Packet Definitions" Generic packet ACP packet ISRC1 packet ISRC2 packet Gamut metadata packet Vendor specific InfoFrame packet AVI InfoFrame packet Source product descriptor InfoFrame packet Audio InfoFrame packet MPEG source InfoFrame packet

HDMI_regac
Address: Operational Base + offset (0x02b0)
Registerac

Bit	Attr	Reset Value	Description
7:0	RW	0x00	After configure the packet offset register, these registers can configure any packet shown following. Use this function please refer to the HDMI specification at "Data Island Packet Definitions" Generic packet ACP packet ISRC1 packet ISRC2 packet Gamut metadata packet Vendor specific InfoFrame packet AVI InfoFrame packet Source product descriptor InfoFrame packet Audio InfoFrame packet MPEG source InfoFrame packet

HDMI_regad
Address: Operational Base + offset (0x02b4)
Registerad

Bit	Attr	Reset Value	Description
7:0	RW	0x00	After configure the packet offset register, these registers can configure any packet shown following. Use this function please refer to the HDMI specification at "Data Island Packet Definitions" Generic packet ACP packet ISRC1 packet ISRC2 packet Gamut metadata packet Vendor specific InfoFrame packet AVI InfoFrame packet Source product descriptor InfoFrame packet Audio InfoFrame packet MPEG source InfoFrame packet

HDMI_regae
Address: Operational Base + offset (0x02b8)
Registerae

Bit	Attr	Reset Value	Description
7:0	RW	0x00	After configure the packet offset register, these registers can configure any packet shown following. Use this function please refer to the HDMI specification at "Data Island Packet Definitions" Generic packet ACP packet ISRC1 packet ISRC2 packet Gamut metadata packet Vendor specific InfoFrame packet AVI InfoFrame packet Source product descriptor InfoFrame packet Audio InfoFrame packet MPEG source InfoFrame packet

HDMI_regaf
Address: Operational Base + offset (0x02bc)
Registeraf

Bit	Attr	Reset Value	Description
7:0	RW	0x00	After configure the packet offset register, these registers can configure any packet shown following. Use this function please refer to the HDMI specification at "Data Island Packet Definitions" Generic packet ACP packet ISRC1 packet ISRC2 packet Gamut metadata packet Vendor specific InfoFrame packet AVI InfoFrame packet Source product descriptor InfoFrame packet Audio InfoFrame packet MPEG source InfoFrame packet

HDMI_regb0
Address: Operational Base + offset (0x02c0)
Registerb0

Bit	Attr	Reset Value	Description
7:0	RW	0x00	After configure the packet offset register, these registers can configure any packet shown following. Use this function please refer to the HDMI specification at "Data Island Packet Definitions" Generic packet ACP packet ISRC1 packet ISRC2 packet Gamut metadata packet Vendor specific InfoFrame packet AVI InfoFrame packet Source product descriptor InfoFrame packet Audio InfoFrame packet MPEG source InfoFrame packet

HDMI_regb1
Address: Operational Base + offset (0x02c4)
Registerb1

Bit	Attr	Reset Value	Description
7:0	RW	0x00	After configure the packet offset register, these registers can configure any packet shown following. Use this function please refer to the HDMI specification at "Data Island Packet Definitions" Generic packet ACP packet ISRC1 packet ISRC2 packet Gamut metadata packet Vendor specific InfoFrame packet AVI InfoFrame packet Source product descriptor InfoFrame packet Audio InfoFrame packet MPEG source InfoFrame packet

HDMI_regb2
Address: Operational Base + offset (0x02c8)
Registerb2

Bit	Attr	Reset Value	Description
7:0	RW	0x00	After configure the packet offset register, these registers can configure any packet shown following. Use this function please refer to the HDMI specification at "Data Island Packet Definitions" Generic packet ACP packet ISRC1 packet ISRC2 packet Gamut metadata packet Vendor specific InfoFrame packet AVI InfoFrame packet Source product descriptor InfoFrame packet Audio InfoFrame packet MPEG source InfoFrame packet

HDMI_regb3
Address: Operational Base + offset (0x02cc)
Registerb3

Bit	Attr	Reset Value	Description
7:0	RW	0x00	After configure the packet offset register, these registers can configure any packet shown following. Use this function please refer to the HDMI specification at "Data Island Packet Definitions" Generic packet ACP packet ISRC1 packet ISRC2 packet Gamut metadata packet Vendor specific InfoFrame packet AVI InfoFrame packet Source product descriptor InfoFrame packet Audio InfoFrame packet MPEG source InfoFrame packet

HDMI_regb4
Address: Operational Base + offset (0x02d0)
Registerb4

Bit	Attr	Reset Value	Description
7:0	RW	0x00	After configure the packet offset register, these registers can configure any packet shown following. Use this function please refer to the HDMI specification at "Data Island Packet Definitions" Generic packet ACP packet ISRC1 packet ISRC2 packet Gamut metadata packet Vendor specific InfoFrame packet AVI InfoFrame packet Source product descriptor InfoFrame packet Audio InfoFrame packet MPEG source InfoFrame packet

HDMI_regb5
Address: Operational Base + offset (0x02d4)
Registerb5

Bit	Attr	Reset Value	Description
7:0	RW	0x00	After configure the packet offset register, these registers can configure any packet shown following. Use this function please refer to the HDMI specification at "Data Island Packet Definitions" Generic packet ACP packet ISRC1 packet ISRC2 packet Gamut metadata packet Vendor specific InfoFrame packet AVI InfoFrame packet Source product descriptor InfoFrame packet Audio InfoFrame packet MPEG source InfoFrame packet

HDMI_regb6
Address: Operational Base + offset (0x02d8)
Registerb6

Bit	Attr	Reset Value	Description
7:0	RW	0x00	After configure the packet offset register, these registers can configure any packet shown following. Use this function please refer to the HDMI specification at "Data Island Packet Definitions" Generic packet ACP packet ISRC1 packet ISRC2 packet Gamut metadata packet Vendor specific InfoFrame packet AVI InfoFrame packet Source product descriptor InfoFrame packet Audio InfoFrame packet MPEG source InfoFrame packet

HDMI_regb7
Address: Operational Base + offset (0x02dc)
Registerb7

Bit	Attr	Reset Value	Description
7:0	RW	0x00	After configure the packet offset register, these registers can configure any packet shown following. Use this function please refer to the HDMI specification at "Data Island Packet Definitions" Generic packet ACP packet ISRC1 packet ISRC2 packet Gamut metadata packet Vendor specific InfoFrame packet AVI InfoFrame packet Source product descriptor InfoFrame packet Audio InfoFrame packet MPEG source InfoFrame packet

HDMI_regb8
Address: Operational Base + offset (0x02e0)
Registerb8

Bit	Attr	Reset Value	Description
7:0	RW	0x00	After configure the packet offset register, these registers can configure any packet shown following. Use this function please refer to the HDMI specification at "Data Island Packet Definitions" Generic packet ACP packet ISRC1 packet ISRC2 packet Gamut metadata packet Vendor specific InfoFrame packet AVI InfoFrame packet Source product descriptor InfoFrame packet Audio InfoFrame packet MPEG source InfoFrame packet

HDMI_regb9
Address: Operational Base + offset (0x02e4)
Registerb9

Bit	Attr	Reset Value	Description
7:0	RW	0x00	After configure the packet offset register, these registers can configure any packet shown following. Use this function please refer to the HDMI specification at "Data Island Packet Definitions" Generic packet ACP packet ISRC1 packet ISRC2 packet Gamut metadata packet Vendor specific InfoFrame packet AVI InfoFrame packet Source product descriptor InfoFrame packet Audio InfoFrame packet MPEG source InfoFrame packet

HDMI_regba
Address: Operational Base + offset (0x02e8)
Registerba

Bit	Attr	Reset Value	Description
7:0	RW	0x00	After configure the packet offset register, these registers can configure any packet shown following. Use this function please refer to the HDMI specification at "Data Island Packet Definitions" Generic packet ACP packet ISRC1 packet ISRC2 packet Gamut metadata packet Vendor specific InfoFrame packet AVI InfoFrame packet Source product descriptor InfoFrame packet Audio InfoFrame packet MPEG source InfoFrame packet

HDMI_regbb
Address: Operational Base + offset (0x02ec)
Registerbb

Bit	Attr	Reset Value	Description
7:0	RW	0x00	After configure the packet offset register, these registers can configure any packet shown following. Use this function please refer to the HDMI specification at "Data Island Packet Definitions" Generic packet ACP packet ISRC1 packet ISRC2 packet Gamut metadata packet Vendor specific InfoFrame packet AVI InfoFrame packet Source product descriptor InfoFrame packet Audio InfoFrame packet MPEG source InfoFrame packet

HDMI_regbc
Address: Operational Base + offset (0x02f0)
Registerbc

Bit	Attr	Reset Value	Description
7:0	RW	0x00	After configure the packet offset register, these registers can configure any packet shown following. Use this function please refer to the HDMI specification at "Data Island Packet Definitions" Generic packet ACP packet ISRC1 packet ISRC2 packet Gamut metadata packet Vendor specific InfoFrame packet AVI InfoFrame packet Source product descriptor InfoFrame packet Audio InfoFrame packet MPEG source InfoFrame packet

HDMI_regbd
Address: Operational Base + offset (0x02f4)
Registerbd

Bit	Attr	Reset Value	Description
7:0	RW	0x00	After configure the packet offset register, these registers can configure any packet shown following. Use this function please refer to the HDMI specification at "Data Island Packet Definitions" Generic packet ACP packet ISRC1 packet ISRC2 packet Gamut metadata packet Vendor specific InfoFrame packet AVI InfoFrame packet Source product descriptor InfoFrame packet Audio InfoFrame packet MPEG source InfoFrame packet

HDMI_regbe
Address: Operational Base + offset (0x02f8)
Registerbe

Bit	Attr	Reset Value	Description
7:0	RW	0x00	After configure the packet offset register, these registers can configure any packet shown following. Use this function please refer to the HDMI specification at "Data Island Packet Definitions" Generic packet ACP packet ISRC1 packet ISRC2 packet Gamut metadata packet Vendor specific InfoFrame packet AVI InfoFrame packet Source product descriptor InfoFrame packet Audio InfoFrame packet MPEG source InfoFrame packet

HDMI_regc0
Address: Operational Base + offset (0x0300)
Registerc0

Bit	Attr	Reset Value	Description
7:0	RW	0x00	mask_int Mask for Interrupt 1'b0: masked 1'b1: not mask [5] Interrupt for active Vsync edge [2] Interrupt for EDID Ready

HDMI_regc1
Address: Operational Base + offset (0x0304)
Registerc1

Bit	Attr	Reset Value	Description
7:0	RW	0x00	int_statu Interrupt status [5] Interrupt for active Vsync edge This bit will active when the mask for this bit to be set 1.When Interrupt occurs, this bit will be equal to 1. Clear interrupt: write 1 to this bit. [2] Interrupt for EDID Ready This bit will active when the mask for this bit to be set 1.When Interrupt occurs, this bit will be equal to 1. Clear interrupt: write 1 to this bit.

HDMI_regc2
Address: Operational Base + offset (0x0308)
Registerc2

Bit	Attr	Reset Value	Description
7:3	RW	0x0f	int_mask Mask for interrupt: 1'b0:masked 1'b1:not masked [7] for HDCP error [6] for bksv fifo ready [5] for bksv update [4] for authentication success [3] for ready to start authentication
2:0	RW	0x0	reserved

HDMI_regc3
Address: Operational Base + offset (0x030c)
Registerc3

Bit	Attr	Reset Value	Description
7:3	RW	0x00	int_status Interrupt status [7] HDCP error Interrupt This bit will active when the mask for this bit to be set 1.When Interrupt occurs, this bit will be equal to 1. Clear interrupt: write 1 to this bit. [6] bksv fifo ready This bit will active when the mask for this bit to be set 1.When Interrupt occurs, this bit will be equal to 1. Clear interrupt: write 1 to this bit. [5] bksv update This bit will active when the mask for this bit to be set 1.When Interrupt occurs, this bit will be equal to 1. Clear interrupt: write 1 to this bit. [4] authentication success This bit will active when the mask for this bit to be set 1.When Interrupt occurs, this bit will be equal to 1. Clear interrupt: write 1 to this bit. [3] ready to start authentication This bit will active when the mask for this bit to be set 1. When Interrupt occurs, this bit will be equal to 1. Clear interrupt: write 1 to this bit.
2:0	RW	0x0	reserved

HDMI_regc4
Address: Operational Base + offset (0x0310)
Registerc4

Bit	Attr	Reset Value	Description
7	RW	0x00	reserved

Bit	Attr	Reset Value	Description
6:0	RW	0x00	int_mask Mask for interrupt: 1'b0:masked 1'b1:not masked [6] for HDCP authentication M0 ready [5] for first frame arrive [4] for HDCP An ready [3] [2] for HDCP encrypted [1] for HDCP not encrypted (av mute) [0] for HDCP not encrypted (no av mute)

HDMI_regc5
Address: Operational Base + offset (0x0314)
Registerc5

Bit	Attr	Reset Value	Description
7	RW	0x00	reserved

Bit	Attr	Reset Value	Description
6:0	RW	0x00	int_status_group3 Interrupt status Group3 This bit will active when the mask for this bit to be set 1.When Interrupt occurs, this bit will be equal to 1. Clear interrupt: write 1 to this bit. [6] HDCP authentication M0 ready This bit will active when the mask for this bit to be set 1.When Interrupt occurs, this bit will be equal to 1. Clear interrupt: write 1 to this bit. [5] irst frame arrive This bit will active when the mask for this bit to be set 1.When Interrupt occurs, this bit will be equal to 1. Clear interrupt: write 1 to this bit. [4] HDCP An ready This bit will active when the mask for this bit to be set 1.When Interrupt occurs, this bit will be equal to 1. Clear interrupt: write 1 to this bit. [2] HDCP encrypted This bit will active when the mask for this bit to be set 1.When Interrupt occurs, this bit will be equal to 1. Clear interrupt: write 1 to this bit. [1] HDCP not encrypted (av mute) This bit will active when the mask for this bit to be set 1.When Interrupt occurs, this bit will be equal to 1. Clear interrupt: write 1 to this bit. [0] HDCP not encrypted (no av mute) This bit will active when the mask for this bit to be set 1.When Interrupt occurs, this bit will be equal to 1. Clear interrupt: write 1 to this bit.

HDMI_regc8
Address: Operational Base + offset (0x0320)
Registerc8

Bit	Attr	Reset Value	Description
7	RO	0x0	hot_plug_pin_status Hot plug pin status 1'b1: Plug in 1'b0: Plug out
6	RO	0x0	reserved
5	RW	0x1	Mask_hpd_int Mask for hot plug detect(HPG) interrupt 1'b0：masked 1'b1：not mask
4:2	RW	0x4	reserved
1	RW	0x0	Int_hpd Interrupt for hot plug detect(HPD) This bit will active when the mask for this bit to be set 1.When interrupt occurs,this bit will be equal to 1. Clear interrupt：write 1 to this bit.
0	RW	0x0	reserved

HDMI_regc9
Address: Operational Base + offset (0x0324)
Registerc9

Bit	Attr	Reset Value	Description
7:6	RW	0x1	video_bist_mode Video BIST mode 2'b00: normal color bar. 2'b01: special color bar. 2'b10: black
5	RW	0x0	flush_en Flush enable. After enable it the screen will flush from color bar to black again and again. 1'b1: enable 1'b0: disable

Bit	Attr	Reset Value	Description
4	RW	0x1	disable_colorbar_bist_test Disable color bar BIST test 1'b1: disable 1'b0: enable
3:0	RW	0x0	reserved

HDMI_regce
Address: Operational Base + offset (0x0338)
Registerce

Bit	Attr	Reset Value	Description
7:1	RW	0x0	reserved
0	RW	0x1	tmds_channel_sync The synchronization for TMDS channels The synchronization for TMDS channels is automatic after Reset the HDMI. But you can synchronization for TMDS channels manually. The operation for this function is: first send 0 to this bit, then send 1, and keep 1 into this bit.

HDMI_regd0
Address: Operational Base + offset (0x0340)
Registerd0

Bit	Attr	Reset Value	Description
7	RW	0x0	reserved
6	RW	0x0	modify_start_bit_timing_hisense_tv Modify the start bit timing for Hisense TV 1'b1: Modify the start bit timing for Hisense TV 1'b0: Not modify
5	RW	0x0	reject_rec_cec_broadcast_message Reject receive CEC broadcast message 1'b1: Reject receive CEC broadcast message 1'b0: Not reject
4:3	RW	0x0	reserved

Bit	Attr	Reset Value	Description
2	RW	0x0	CEC_bus_free_time_cnt_en Enable count CEC bus free time 1'b1: Enable count CEC bus free time 1'b0: Not enable
1	RW	0x0	forbid_receive_cec_message Forbid receive CEC message 1'b1: Forbid receive CEC message 1'b0: Not forbid
0	RW	0x0	start_transmit_cec_message Start transmit CEC message 1'b1: Start transmit CEC message 1'b0: Not start

HDMI_regd1
Address: Operational Base + offset (0x0344)
Registerd1

Bit	Attr	Reset Value	Description
7:0	RW	0x00	cec_rx_tx_data_in_fifo CEC RX/TX data in FIFO

HDMI_regd2
Address: Operational Base + offset (0x0348)
Registerd2

Bit	Attr	Reset Value	Description
7:5	RW	0x0	reserved
4:0	RW	0x00	wr_cec_tx_data_addr Point address for write CEC TX data

HDMI_regd3
Address: Operational Base + offset (0x034c)
Registerd3

Bit	Attr	Reset Value	Description
7:5	RW	0x0	reserved
4:0	RW	0x00	cec_rx_data_addr Point address for CEC RX data

HDMI_regd4
Address: Operational Base + offset (0x0350)
Registerd4

Bit	Attr	Reset Value	Description
7:0	RW	0x03	cec_clk_freq CEC working clock frequency register1

HDMI_regd5
Address: Operational Base + offset (0x0354)
Registerd5

Bit	Attr	Reset Value	Description
7:0	RW	0x09	cec_work_clk_freq CEC working clock frequency register1

HDMI_regd6
Address: Operational Base + offset (0x0358)
Registerd6

Bit	Attr	Reset Value	Description
7:5	RW	0x0	reserved
4:0	RW	0x03	tx_block_length Length of TX blocks

HDMI_regd7
Address: Operational Base + offset (0x035c)
Registed7

Bit	Attr	Reset Value	Description
7:5	RW	0x0	reserved
4:0	RW	0x00	rx_block_length Length of RX blocks

HDMI_regd8
Address: Operational Base + offset (0x0360)
Registerd8

Bit	Attr	Reset Value	Description
7:0	RW	0xff	cec_tx_int_mask Mask for CEC TX Interrupts 1'b0: masked 1'b1: not mask [3] Interrupt for TX done [2] Interrupt for TX no ACK from follower [1] Interrupt for TX broadcast rejected [0] Interrupt for CEC line free time not satisfied

HDMI_regd9
Address: Operational Base + offset (0x0364)
Registerd9

Bit	Attr	Reset Value	Description
7:0	RW	0xff	cec_rx_int_mask Mask for CEC RX Interrupts 1'b0: masked 1'b1: not mask [4] Interrupt for RX receive logic address error [3] Interrupt for RX receive glitch error [2] Interrupt for RX ACK detect overtime [1] Interrupt for RX sending broadcast reject [0] Interrupt for RX done

HDMI_regda
Address: Operational Base + offset (0x0368)
Registerda

Bit	Attr	Reset Value	Description
7:0	RW	0x00	cec_tx_int CEC TX Interrupts [3] Interrupt for TX done [2] Interrupt for TX no ACK from follower [1] Interrupt for TX broadcast rejected [0] Interrupt for CEC line free time not satisfied

HDMI_regdb
Address: Operational Base + offset (0x036c)
Registerdb

Bit	Attr	Reset Value	Description
7:0	RW	0x00	cec_tx_int CEC RX Interrupts [4] Interrupt for RX receive logic address error [3] Interrupt for RX receive glitch error [2] Interrupt for RX ACK detect overtime [1] Interrupt for RX sending broadcast reject [0] Interrupt for RX done

HDMI_regdc
Address: Operational Base + offset (0x0370)
Registerdc

Bit	Attr	Reset Value	Description
7:0	RW	0xd0	signal_free_time_l Signal free time length_L

HDMI_regdd
Address: Operational Base + offset (0x0374)
Registerdd

Bit	Attr	Reset Value	Description
7:0	RW	0x20	signal_free_time_h Signal free time length_H

HDMI_regde
Address: Operational Base + offset (0x0378)
Registerde

Bit	Attr	Reset Value	Description
7:4	RW	0x0	reserved
3:0	RW	0x0	dev_addr Device logic address

HDMI_rege0
Address: Operational Base + offset (0x0380)
Registere0

Bit	Attr	Reset Value	Description
7:5	RW	0x0	reserved
4	RW	0x1	TMDS_clk_sel 1'b0: select TMDS clock from PLL; 1'b1: select TMDS clock generated by serializer

Bit	Attr	Reset Value	Description
3	RW	0x0	TMDS_phase_sel 1'b0: select default phase for TMDS clock; 1'b1: select synchronized phase for TMDS clock with regard to TMDS data
2	RW	0x0	HDMI_band_gap_pd HDMI Band gap power down. 1:Active
1	RW	0x0	HDMI_pll_pd HDMI PLL power down. 1:Active
0	RW	0x0	tmds_channel_pd TMDS channel power down. 1:Active

HDMI_rege1
Address: Operational Base + offset (0x0384)
Registere1

Bit	Attr	Reset Value	Description
7:4	RW	0x0	reserved
3	RW	0x1	clk_channel_driver_en Clock channel driver enable 1'b0: disable 1'b1: enable
2:0	RW	0x7	data_channels_driver_en Three data channels driver enable [2]-ch2 [1]-ch1 [0]-ch0 1'b0: disable 1'b1: enable

HDMI_rege2
Address: Operational Base + offset (0x0388)

Registere2

Bit	Attr	Reset Value	Description
7:4	RW	0xa	clk_driver_strength Clock channel main-driver strength 0000~1111: the strength from low to high
3:0	RW	0xa	data_driver_strength Three data channels main-driver strength 0000~1111: the strength from low to high

HDMI_rege3

Address: Operational Base + offset (0x038c)

Registere3

Bit	Attr	Reset Value	Description
7	RW	0x0	reserved
6:4	RW	0x0	pre_emphasis_strength Pre-emphasis strength 000~111: the strength from 0 to high
3:2	RW	0x3	clk_driver_strength Clock channel pre-driver strength Slew rate from low to high 00~11: the strength from low to high
1:0	RW	0x3	data_driver_strength Three data channels pre-driver strength Slew rate from low to high 00~11: the strength from low to high

HDMI_rege7

Address: Operational Base + offset (0x039c)

Registere7

Bit	Attr	Reset Value	Description
7:0	RW	0x78	feedback_dividing_ratio_part1 The value of feedback dividing ratio

HDMI_rege8

Address: Operational Base + offset (0x03a0)

Registere8

Bit	Attr	Reset Value	Description
7:1	RW	0x0	reserved
0	RW	0x0	feedback_dividing_ratio_part2 The value of feedback dividing ratio

HDMI_reged
Address: Operational Base + offset (0x03b4)
Registered

Bit	Attr	Reset Value	Description
7:5	RW	0x0	reserved
4:0	RW	0x0c	pre_dividing_ratio The value of pre-dividing ratio

Note1: There are some bits of registers have not description in the registers table above, are reserved for other using, please keep it into default value.

14.5 Interface Description

14.6 Video Input Source

The HDMI TX video source comes from VOP.

14.7 Audio Input Source

The HDMI TX audio source comes from I2S_8CH.

14.8 ApplicationNotes

This chapter describes how to bring up HDMI transmitter in your system. As shown few examples below, these introduce the basically HDMI transmitter application, likes, the Hot Plug Detect, EDID read back, multiple audio format input and different video resolution displaying.
You can easily configure these functions with proper registers value setting by HDMI TX APB BUS.

14.8.1 Initial Operation

The default HDMI transmitter is configured to 24bit RGB 1080P resolution video with 8 channel 48K sample I2S format audio input. It is easily for customer to turn on HDMI transmitter without doing more complex operation. Just do the step, reset the HDMI TX.

14.8.2 Hot Plug Detection

Hot Plug Detect is a special feature for HDMI transmitter spying the state on the HDMI port. You can control this function by using the interrupt signal and proper registers from the HDMI transmitter with few operations. The following is a step by step instruction for detecting the hot plug in and out.

Hot Plug in Steps:

Step1: Se nd 0 x63 to register 0x00.
Step2: Plug HDMI receiver in.
Step3: Check the interrupt from signal pin_int.
If the pin_int shows high, that means the HDMI transmitter interrupt have generated.
Step4: Check the interrupt.
Read the bit1 o f the register 0 xc8. If bit1=1'b1 that means the hot plug interrupt valid, otherwise there is other interrupt. The n s end the 1'b1 to bit1 of register 0xc8to clean up the hot plug interrupt.
Step5: Check if really HDMI receiver plug in.
Read the bit7 of the register 0xc8. If bit7=1'b1 that means hot plug in was really happen, otherwise there is ma y be a glitch o n the HPD port.

14.8.3 Hot Plug out Steps:

Step1: HDMI transmitter at working state.
Step2: Plug HDMI receiver out.
Step3: Check the interrupt from signal pin_int.
If the pin_int shows high, that means the HDMI transmitter interrupt have generated.
Step4: Check the interrupt.
Read the bit1 of the register 0xc8. If bit1=1'b1 that means the hot plug interrupt valid, otherwise there is other interrupt. The n s end the 1'b1 to bit1 of register 0xc8to clean up the hot plug interrupt.
Step5: Check if really HDMI receiver plug out.
Read the bit7 of the register 0xc8. If bit7=1'b0 that means hot plug out was really happen, otherwise there is ma y be a glitch o n the HPD port.

14.8.4 Reading EDID

Read EDID is a function that can make the HDMI transmitter to read the HDMI receiver's Extended Display Identification Data (EDID) in order to discover the HDMI receiver's configuration and capabilities. HDMI transmitter can choose the appropriate audio and video format for playing and displaying by the HDMI receiver through the use of the EDID. Besides, HDMI transmitter support the reading Enhanced Extended Display Identification Data (E-EDID) if HDMI receiver have this enhanced structure.
The following describes how to read E-EDID through HDMI transmitter. The total E-EDID is 512bytes data, which is divided into 2 segments. Each segment has 256bytes data. The Read E-EDID function is only read 128bytes data from HDMI receiver at each time. So, you must read 4 times that can read total 512bytes data back.

Prepare read E-EDID512bytes Steps:
Step1: Send 0 x63 to register 0x00.
System power down mode.
Step2: HDMI transmitter detected Hot Plug in.
Step3: Define DDC clock (SCL) frequency Fscl = 100K.
Fscl = Freg_clk/(4*X). X={register 0x4c, register 0x4b}.
Note: Freg_clk is the frequency o f the tmds clock.
Assume the tmds clock is 148.5MHz.
Send 0 x73 to register 0x4b, Send 0x01 to register 0x4c.
Step4: Enable EDID Ready interrupt.
Send 0 x04 to register 0xc0
Read E-EDIDsegment 0x00 256bytes Steps:
Step1: Set EDID FIFO initial address.
Send 0x00 to register 0x4f
Step2: Set EDID first word address, read first 128bytes.
Send 0x00 to register 0x4e
Step3: Set EDID segment address.
Send 0x00 to register 0x4d

Step4: Waiting and check EDID interrupt from pin_int.
Check the EDID Ready state from bit2 o f register 0xc1. If bit2 = 1'b1 that means EDID data ready. The n send the 1'b1 to bit2 of register 0xc1 to clean up the EDID Ready interrupt.
Step5: Read first 128bytes EDID data from EDID FIFO.
for (i = 0, i < 128, i = i + 1)
Read data from register 0x50.
Step6: Set EDID FIFO initial address again.
Send 0x00 to register 0x4f
Step7: Set EDID first word address, read last 128bytes.
Send 0x80 to register 0x4e
Step8: Set EDID segment address.
Send 0x00 to register 0x4d
Step9: Waiting and check EDID interrupt from pin_int.
Check the EDID Ready state from bit2 of register 0xc1. If bit2 = 1'b1 that means EDID data ready. The n send the 1'b1 to bit2 of register 0xc1 to clean up the EDID Ready interrupt.
Step10: Read last 128bytes EDID data from EDID FIFO.
for (i = 0, i < 128, i = i + 1)
Read data from register 0x50.

Read E-EDIDsegment 0x01 256bytes Steps:

Step1: Set EDID FIFO initial address.
Send 0x00 to register 0x4f
Step2: Set EDID first word address, read first 128bytes.
Send 0x00 to register 0x4e
Step3: Set EDID segment address.
Send 0x01 to register 0x4d
Step4: Waiting and check EDID interrupt from pin_int.
Check the EDID Ready state from bit2 of register 0xc1. If bit2 =1'b1 that means EDID data ready. The n send the 1'b1 to bit2 of register 0xc1 to clean up the EDID Ready interrupt.
Step5: Read first 128bytes EDID data from EDID FIFO.
for (i = 0, i < 128, i = i + 1)
Read data from register 0x50.
Step6: Set EDID FIFO initial address again.
Send 0x00 to register 0x4f
Step7: Set EDID first word address, read last 128bytes.
Send 0x80 to register 0x4e
Step8: Set EDID segment address.
Send 0x01 to register 0x4d
Step9: Waiting and check EDID interrupt from pin_int.
Check the EDID Ready state from bit2 of register 0xc1. If bit2 =1'b1 that means EDID data ready. The n send the 1'b1 to bit2 of register 0xc1 to clean up the EDID Ready interrupt.
Step10: Read last 128bytes EDID data from EDID FIFO.
for (i = 0, i < 128, i = i + 1)
Read data from register 0x50.

Note: The segment address must be sent at each time when read 128bytes E-EDID data.

14.8.5 Audio input configuration

HDMI transmitteraudio support either SPDIF or four channel I2S input. SPDIF input supports audio sampling rates from 32to 192 KHz. The I2S input supports from 2-channel to 8-channel audio up to 192 KHz. The default audio format is I2S input with 8 channels. The audio sample rate is 48K.
The following describes how to configure audio input format into I2S input with 2 channels and the sample rate is 44.1K.

Audio input requirement:
SD0 input (2 Channels I2S input)
WS (fs) = 44.1 KHz
SCK = 64fs
MCLK = 256fs

Configure Audio Input Format Steps:
Step1: Select I2S input.
Send 0 x01 to register 0x35
Step2: Select SD0 input.
Send 0 x04 to register 0x38
Step3: Select N/CTS for sample rate = 44.1 KHz.
Send 0 x18 to register 0x40
Send 0x80 to register 0x41

The following describes how to configure audio input format into I2S input with 8 channels and the sample rate is 44.1K.

Audio input requirement:
SD[3 :0] input (8 C ha nnels I2S input)
WS (fs) = 44.1 KHz
SCK = 64fs
MCLK = 256fs

Configure Audio Input Format Steps :
Step1: Select I2S input.
Send 0x01 to register 0x35
Step2: Select SD[3:0] input.
Send 0x3c to register 0x38
Send 0x00 to register 0x39
Step3: Select N/CTS for sample rate = 44.1 KHz.
Send 0x18 to register 0x40
Send 0x80 to register 0x41

14.8.6 Video input configuration

HDMI transmitter support RGB/YCbCr 24/30/36bit video input with different resolution. The default video format is RGB24bit input at resolution of 1080P@60. The following describes how to configure video input format into RGB24bit input at resolution of 480P@60, 720P@60 or 1080P@60.
VOP dclk cannot get invert.

Video input requirement:
24bit RGB 4:4:4 Source.
Resolution is 480P@60, 720P@60 or 1080P@60.
Configure Video Input Format Steps:
Step1: Select configure for video format.
Send 0x06 to register 0x9f
Step2: Configure the AVI info to HDMI RX.
Send 0x82 to register 0xa0 //HB0
Send 0x02 to register 0xa1 //HB1
Send 0x0d to register 0xa2 //HB3
Send 0x00 to register 0xa3 //PB0: checksum is auto set, needn’t set this register
Send 0x00 to register 0xa4 //PB1
Send 0x08 to register 0xa5 //PB2
Send 0x70 to register 0xa6 //PB3

Send 0x10 to register 0xa7 //PB4: VID 1920*1080P@60
Send 0x40 to register 0xa8 //PB5
Note: Select correct VID for displaying.
Send 0x02 to register 0xa7 for 720*480P@60.
Send 0x04 to register 0xa7 for 1280*720P@60.
Send 0x05 to register 0xa7 for 1920*1080I@60.
Send 0x10 to register 0xa7 for 1920*1080P@60.
Send 0x11 to register 0xa7 for 720*576P@50.
Send 0x13 to register 0xa7 for 1280*720P@50.
Send 0x14 to register 0xa7 for 1920*1080I@50.
Send 0 x1 f to register 0xa7 for 1920*1080P@50.

The detail configuration for AVI information, please refer to the HDMI specification (8.2.1) and CEA-861-D (6.3).

14.8.7 Low Power Mode

HDMI Transmitter can be configured into Low Power Mode at special customer works mode. The following is a step by step instruction to describe how to configure our HDMI Transmitter into this mode.

Low Power Enter Steps:

Step1:HDMI Transmitter working.
Step2:Low Power analog module.
Send 0x00 to register 0xe1.//Drive disable
Send 0x17 to register 0xe0.
Step3:Assign pin_vclk = 1'b0 or 1'b1.

Low Power Quit Steps:

Step1: Reset system by pin_rst_n.
Ste p2: Wait Hot Plug.
Step3: Read EDID.
Step4: Active vclk through p in_vclk
Step5: Bring up analog module.
Send 0x15 to register 0xe0. //PLL power on
Send 0x14 to register 0xe0. //TX power on
Send 0x10 to register 0xe0. //BG power on
Send 0 x0f to register 0xe1. //driver enable
Step6: Configuration mode reg at addr 0x00 if needed.
Step7: Configuration Video format if needed.
Step8: Configuration Audio format if needed.
Stap9: Configuration mode reg, power on digital part and select tmds_clk for configuration.
Send 0 x61 to register 0x00.
Stap10: Synchronize analog module.
Send 0x00 to register 0xce.
Send 0x01 to register 0xce.

14.8.8 CEC OPERATION

The CEC line is used for high-level user control of HDMI-connected devices.
You can control this function by using the interrupt signal and proper registers from the HDMI transmitter with few operations.
The following is a step by step instruction for CEC TX operations and CEC RXoperations.

CEC TX steps

(Send the Command : Standby (0x36) for example)

Step1: Set logic address for CEC (logic address = 1).
Send 0x01 to register 0xde.
Step2: Configure the divide numbers for internal reference clock.
Send 0 x03 to register 0xd4 and 0x09 to register 0 xd5. See Note1 for details.
Step3: Configure the value for signal free time counter.
Send 0xd0 to register 0xdc and 0x20 to register 0xdd. See Note2 for details.
Step4: Configure point address for write CEC TX data.
Send 0x00 to register 0xd2.
Step5: Configure TX data FIFO (the Command o f Standby).
Send 0 x01 to register 0xd1, then, send 0x36 to register 0xd1.
The data 0x01 is for the Header Block, in which the high 4bits is for the initiator, the low 4bits is for the destination. The data 0x36 is for the Data Block, it indicate the command for standby.
Step6: Configure TX data length of this packet. The length is 2.
Send 0x02 to register 0xd6.
Step7: Enable count CEC b us free time.
Send 0x04 to register 0xd0.
Step8: Waiting 16.8ms+4 us for CEC bus free.
If Interrupt happened for CEC line free time not satisfied. You should change to receive coming data from opposite. After this you can do CEC TX steps again.
(Note that the time for 16.8ms+4us should be compute by MCU itself)
Step9: Begin TX.
Send 0x05 to register 0xd0.
Step10: Waiting and c he ck interrupt o f C EC TX do ne.
Waiting the interrupt, the n, read the value of register 0xda, if the value is 0x08 that means CEC command was transmitted successfully. At last, send 0x08 to register 0xda to clear up this interrupt.

CEC RX steps:
Step1: Set logic address for CEC (logic address = 1).
Send 0x01 to register 0xde.
Step2: Configure the divide numbers for internal reference clock. See Note1 for details.
Send 0 x03 to register 0xd4 and 0x09 to re gis ter 0 xd5.
Step3: Configure the point address fo r CEC RX data.
Send 0x00 to register 0xd3.
Step4: Waiting CEC TX send CEC packet to o ur HDMI Transmitter, then check interrupt of CEC.Read the value of register 0xdb, if the value is 0x80, thatmeans CEC command was received successfully. Then send 0x80 to register 0xdb to clear up this interrupt.
Step5: Read the RX packet le ngth.
Read the value of the register 0xd7.
Step6: Read the received CEC packet.
Read the value of the register 0 xd1 according to the RX packet length. If the length is 2, please read twice from the register 0xd1.

Note1: The system clock (Default value Fsys = 20MHz from pin_sys_clk), registe r 0xd4 and 0xd5 are used to configure the CEC logic to generate a reference clock (Fref=0.5 MHz, Tref = 2us), which is us e d to control t he CEC signal's generation. The following is the formula for generating reference clock 0.5MHz. Fref = Fsys / ((register 0xd4 + 1)(register 0xd5 + 1)). Under the default 20MHz system clock, the values of register 0xd4 and 0xd5 are 0x03 and 0x09.*
*Note2: Before attempting to transmit or re-transmit a frame, a device shall ensure that the CEC line has be e n inactive for a number of bit periods . For example , the signal free time is 7*2.4ms=16.8ms (the bit time is 2.4 ms). According to the 16.8ms/2us = {register 0xdd, register 0xdc}, so the register 0xdd=0x20 and register 0xdc= 0xd0.*

14.8.9 HDCP OPERATION

HDCP is designed to protect the transmission of Audiovisual Content between an HDCP Transmitter and an HDCP Receiver. You can control this function by using the interrupt signal and proper registers from the HDMI transmitter with few operations.

The following is a step by step instruction for HDCP operation.

HDCP operation steps（skip BKSV black list check）:

Example Condition:Video Format：1920×1080p@59.94/60Hz
Step1:Power on HDMI_TX.
Send 0x61 to register 0x00.
Step2: Write HDCP transmitter's Akey to the chip.See Note1 for details.

```
for(i =0; i <= 39 ; i = i+ 1 )
begin
hdcp_wdata_temp = akey[i];
    for ( j=0 ; j < 7   ; j = j + 1 )
begin
Send hdcp_wdata_temp[7:0] to register 0x98;
hdcp_wdata_temp = hdcp_wdata_temp >>8;
end
end
```

Step3: Write HDCP transmitter's AKSV1 to the chip.See Note1 for details.

```
hdcp_wdata_temp = pre_ksv1
for (i = 0, i<5, i = i + 1)
begin
Send hdcp_wdata_temp[7:0] to register 0x98;
hdcp_wdata_temp = hdcp_wdata_temp >>8;
end
```

Step4: Write HDCP transmitter's AKSV2 to the chip.See Note1 for details.

```
hdcp_wdata_temp = pre_ksv2
for (i = 0, i<5, i = i + 1)
begin
Send hdcp_wdata_temp[7:0] to register 0x98;
hdcp_wdata_temp = hdcp_wdata_temp >>8;
end
```

Step5: Check the key_ready signal.
Read the value of register 0x54, if the value is 0x01,that means the Akey and AKSV were load into the chip successfully.
Step6: Configure the DDC frequency.See Note2 for details.
Send 0x73 to register 0x4b.
Send 0x01 to register 0x4c.
Step7: Configure the HDCP function.
Send 0x77 to register 0x53.
Step8: Open the mask for HDCP interrupt.
Send 0xff to register 0xc2.
Send 0xff to register 0xc4.
Step9: Start the HDCP authentication.
Send 0x96 to register 0x52.
Step10: Checking HDCP interrupt.
Read the value of register 0xc3 and register 0xc5, if the value of register 0xc3 is 0x10 and the value of register 0xc5 is 0x65, that is means the HDCP authenticationwas successfully finished.
Step11: Integrity check: After the completion of the first part of the authentication protocol, the frame counter count periodic from 1 to 128, and the Ri value is updated for every 128th frame counter increment, starting with the 128th.The integrity check Ri start at the vertical blanking interval of the next 2nd frame or the next 1st frame, depending on the register 0x53[3] is set to 1 or 0, occurs every 128 frames. Read the value of register 0x57, if reg57[7]=0, indicated that the authentication integrity was unsuccessful, otherwise indicated that the authentication was successful. If an interrupt happened after authentication success, please read back the value of register 0xc3, and check whether regc3[7]=1, if indeed it indicated that there are some errors of hdcp, then, checking register 0x65 for more information is necessary.

Note: reg0xc3[4] is the interrupt register, only =1 indicated that the authentication state changed from unsuccessful to successful. It should be cleared after checking every time. But, the authentication information can be obtained all the time if needed from the reg0x57[7].

HDCP operation steps（not skip BKSV black list check）:

Example Condition: Video Format : 1920×1080p@59.94/60Hz
Step1:Power on HDMI_TX.
Send 0x61 to register 0x00.
Step2: Write HDCP transmitter's Akey to the chip.See Note1 for details.

```
for(i =0; i <= 39 ; i = i+ 1 )
begin
hdcp_wdata_temp = akey[i];
    for ( j=0 ; j < 7   ; j = j + 1 )
begin
Send hdcp_wdata_temp[7:0] to register 0x98;
hdcp_wdata_temp = hdcp_wdata_temp >>8;
end
end
```

Step3: Write HDCP transmitter's AKSV1 to the chip.See Note1 for details.

```
hdcp_wdata_temp = pre_ksv1
for (i = 0, i<5, i = i + 1)
begin
Send hdcp_wdata_temp[7:0] to register 0x98;
hdcp_wdata_temp = hdcp_wdata_temp >>8;
end
```

Step4: Write HDCP transmitter's AKSV2 to the chip.See Note1 for details.

```
hdcp_wdata_temp = pre_ksv2
for (i = 0, i<5, i = i + 1)
begin
Send hdcp_wdata_temp[7:0] to register 0x98;
hdcp_wdata_temp = hdcp_wdata_temp >>8;
end
```

Step5: Check the key_ready signal.
Read the value of register 0x54, if the value is 0x01,that means the Akey and AKSV were load into the chip successfully.
Step6: Configure the DDC frequency.See Note2 for details.
Send 0x73 to register 0x4b.
Send 0x01 to register 0x4c.
Step7: Configure the HDCP function.
Send 0x37 to register 0x53(do not skip the BKSV blacklist check).
Step8: Open the mask for HDCP interrupt.
Send 0xff to register 0xc2.
Send 0xff to register 0xc4.
Step9: Start the HDCP authentication.
Send 0x96 to register 0x52.
Step10: Check the BKSV.
Wait for INT from registerc3 bit5(bksv_update),than read register66~register6a for BKSV values to check whether the BKSV is in the revocation list. If the BKSV is not in the revocation list, write 1 to register52 bit6(bksv_pass) , and authentication will continue. Else if the BKSV is in the revocation list, write 1 to register52 bit5(bksv_fail), and authentication will stop.
Step11: Checking HDCP interrupt.
Read the value of register 0xc3 and register 0xc5, if the value of register 0xc3 is 0x10 and the value of register 0xc5 is 0x65, that is means the HDCP authenticationwas successfully finished.
Step12: Integrity check: After the completion of the first part of the authentication protocol, the frame counter count periodic from 1 to 128, and the Ri value is updated for every 128th frame counter increment, starting with the 128th.The integrity check Ri start

at the vertical blanking interval of the next 2nd frame or the next 1st frame, depending on the register 0x53[3] is set to 1 or 0, occurs every 128 frames. Read the value of register 0x57, if reg57[7]=0, indicated that the authentication integrity was unsuccessful, otherwise indicated that the authentication was successful. If an interrupt happened after authentication success, please read back the value of register 0xc3, and check whether regc3[7]=1, if indeed it indicated that there are some errors of hdcp, then, checking register 0x65 for more information is necessary.

Note: reg0xc3[4] is the interrupt register, only =1 indicated that the authentication state changed from unsuccessful to successful. It should be cleared after checking every time. But, the authentication information can be obtained all the time if needed from the reg0x57[7].

Note1: For HDCP's akey and KSV write, use the TEST KEYS in HDCP specification for example

```
akey[0] =56'h4da4588f131e69;
akey[1] =56'h1f823558e65009;
akey[2] =56'h8a6a47abb9980d;
akey[3] =56'hf3181b52cbc5ca;
akey[4] =56'hfb147f6896d8b4;
akey[5] =56'he08bc978488f81;
akey[6] =56'ha0d064c8112c41;
akey[7] =56'hb39d5a28242044;
akey[8] =56'hb928b2bdad566b;
akey[9] =56'h91a47b4a6ce4f6;
akey[10] =56'h5600f8205e9d58;
akey[11] =56'h8c7fb706ee3fa0;
akey[12] =56'hc02d8c9d7cbc28;
akey[13] =56'h561261e54b9f05;
akey[14] =56'h74f0de8ccac1cb;
akey[15] =56'h3bb8f60efcdb6a;
akey[16] =56'ha02bbb16b22fd7;
akey[17] =56'h482f8e46785498;
akey[18] =56'h66ae2562274738;
akey[19] =56'h3d4952a323ddf2;
akey[20] =56'he2d231767b3a54;
akey[21] =56'h4d581aede66125;
akey[22] =56'h326082bf7b22f7;
akey[23] =56'hf61b463530ce6b;
akey[24] =56'h360409f0d7976b;
akey[25] =56'ha1e105618d49f9;
akey[26] =56'hc98e9dd1053406;
akey[27] =56'h20c36794426190;
akey[28] =56'h964451ceac4fc3;
akey[29] =56'h3e904504e18c8a;
akey[30] =56'h290010579c2dfc;
akey[31] =56'hd7943b69e5b180;
akey[32] =56'h54c7ea5bdd7b43;
akey[33] =56'h74fb5887c790ba;
akey[34] =56'h935cfa364e1de0;
akey[35] =56'h03075e159a11ae;
akey[36] =56'h05d3408a78fb01;
akey[37] =56'h0059a5d7a04db3;
akey[38] =56'h373b634a2c9e40;
akey[39] =56'h2573bbb4562041;

pre_ksv1 = 40'hb70361f714;
pre_ksv2 = 40'hb70361f714;
```

Note2: For DDC frequency configuration

In the HDCP specification, the frequency value of DDC(Fddc) support only up to 100Kbps, and now we select the TMDS clock as the reference clock of DDC , and under the condition of 1920× 1080p@59.94/60Hz, the frequency of TMDS(FTMDS) is 148.5MHz. Pay attention, internally we use the 4 times of DDC frequency to generate the SCL. So X = FTMDS/Fddc/4 = 'h173, So the values of register 0x4b and 0x4c are 0x73 and 0x01.

14.8.10 NOMAL OPERATION EXAMPLE AT 1080P

After the description of HDMI Transmitter configuration above, the following example shows an entire configuration for HDMI Transmitter works on the 1080P mode.

Audio input requirement:

SD0 input (2 Channels I2S input)
WS (fs) = 48 KHz
SCK = 64fs
MCLK = 256fs

Video input requirement:

24bit RGB 4:4:4 Source.
Resolution is 1080P@60.

Setup Steps:

Step1:Power on HDMI Transmitter.
Step2: Configure the input signals.
Assign pin_test_en = 1'b0.
Step3:Reset HDMI Transmitter.
Assign 0 to the signal pin_rst_n and then assign 1.
Step4:System power down.
Send 0x63 to register 0x00.
Step5:Detect Hot Plug In.
Step6:Read EDID.
Step7: Configure video input format.
Step8:Configure audio input format.
Step9:Assign video and audio source to HDMI Transmitter.
Step10:System power on.
Send 0x61 to register 0x00.
Step11: Now, HDMI Transmitter is ready to go. Start your operation.

14.9 Electrical Specification

HDMI Transmitter contains tunable source termination and pre-emphasis to enable high speed operation. The Transmitter meets the AC specifications below across all operating conditions specified. Rise and fall times are defined as the signal transition time between 20% and 80% of the nominal swing voltage (Vswing) of the device under test. The Transmitter intra-pair skew is the maximum allowable time difference (on both low-to-high and high-to-low transitions) as measured at TP1, between the true and complement signals of a given differential pair. This time difference is measured at the midpoint on the single-ended signal swing of the true and complement signals. The Transmitter inter-pair skew is the maximum allowable time difference (on both low-to-high and high-to-low transitions) as measured at TP1.

Table 14-5 Normal Operating Conditions

Symbol	Parameter	Min	Typ	Max	Units
VCC	Supply Voltage	3.0	3.3	3.6	V
VREF	Reference Voltage	3.0	3.3	3.6	V
VCCN	Supply Voltage Noise			100	mVp. p
TA	Ambient Temperature (with power applied)	0	25	70	℃

Table 14-6 Absolute Maximum Conditions

Symbol	Parameter	Min	Typ	Max	Units
VCC1.2	Supply Voltage 3.3V	-0.3		4.0	V
V11.2	Input Voltage	-0.3		VCC+0.3	V
V01.2	Output Voltage	-0.3		VCC+0.3	V
VJ1.2	Junction Temperature (with power applied)			125	℃

Table 14-7 Transmitter DC Characteristics at TP1

Item	Value
Single-ended high level output voltage, VH	AVcc-10mV< VH < AVcc+10mV when sink <= 165Mhz
	AVcc-200mV< VH < AVcc+10mV when sink >165Mhz
Single-ended low level output voltage, VL	(AVcc – 600mV) < VL < (AVcc – 400mV) when sink <=165Mhz
	(AVcc – 700mV) < VL < (AVcc – 400mV) when sink > 165Mhz
Single-ended output swing voltage, Vswing	400mV < Vswing< 600mV
Single-ended standby (off) output voltage, VOFF	AVcc+-10mV
Single-ended standby (off) output current, IOFF	| IOFF|< 10uA

Table 14-8 Transmitter AC Characteristics at TP1

Item	Value
Risetime/falltime (20%-80%)	75psec< Risetime/falltime < 0.4 Tbit
Overshoot, max	15% of full differential amplitude (Vswing *2)
Undershoot, max	25% of full differential amplitudee (Vswing *2)
Intra-Pair Skew at Transmitter Connector, max	0.15 Tbit
Inter-Pair Skew at Transmitter Connector, max	0.20 Tpixel
TMDS Differential Clock Jitter, max	0.25 Tbit
Clock duty cycle	40% to 60%

Table 14-9 Programmable Output Resistance and Output Equalization Level

Item	Value
Output Equalization level	10% to 60%
Termination Resistance	50ohm and 75 ohms selectable with fine steps
OutputSwingRanges	4 different levels

14.9.1 CONTROL SIGNAL - DDC

DDC (Display Data Channel) control signals follow the I2C Bus specifications. More details to be filled out from I2C specs.

Item	Value
High Voltage Level (Vih)	-3.0V < Vih < 3.8V
Low Voltage Level (Vil)	-0.5V < Vil < 1.5V
SCL clock frequency	100KHz (max)
Rise time	< 1000ns
Fall time	< 300ns
Capacitive load on bus line	400pF

14.9.2 CONTROL SIGNAL - HPD

HPD (Hot Plug Detect) signal is used by the source to read the sink's E-DID. The sink needs to meet the following requirements for reliable detection.
For Sink

Item	Value
High Voltage Level (Vih)	2.4V < Vih < 5.3V
Low Voltage Level (Vil)	0V < Vil < 0.4V

For Source

Item	Value
High Voltage Level (Voh)	2.0V < Voh < 5.3V
Low Voltage Level (Vol)	0V < Vol < 0.8V

Chapter 15 LVDS

15.1 Overview

LVDS IP is integrated into MIPI D-PHY. LVDS transmitter converts a CMOS signal into a low-voltage differential signal. Using a differential signal reduces the system's susceptibility to noise and EMI emissions. In addition, using a differential signal can deliver high speeds. This results in a very cost-effective solution to some of the greatest bandwidth bottlenecks in many transmission applications.
LVDS supports the following features:

- 150MHz clock support
- LVDS 24bits or 18bits color data output
- PLL requires no external components
- Combine LVTTL IO, support LVDS/LVTTL data output
- Comply with the Standard TIA/EIA-644-A LVDS standard
- Support 8bit format-1, format-2, format-3 display mode, Support 6bit display mode.
- Display mode can be select by input MUX
- Consumes Less Than 1mW When Disabled

15.2 Block Diagram

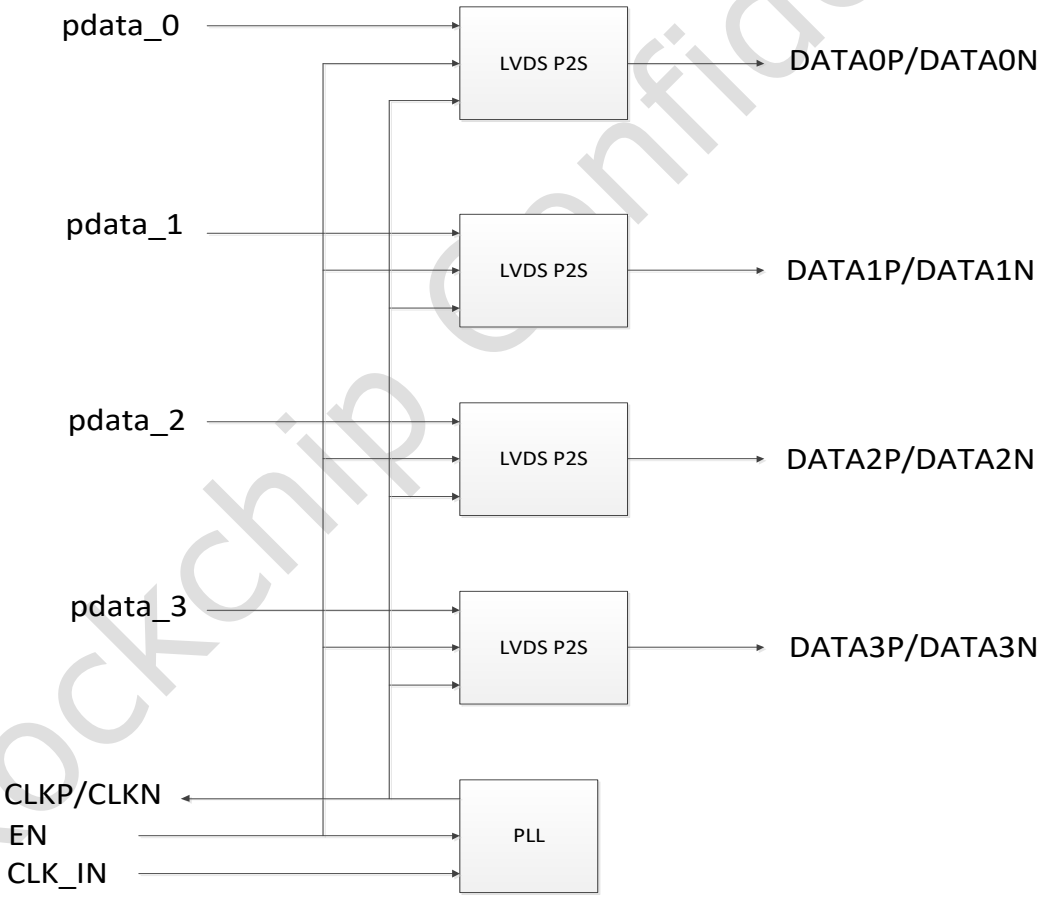


Fig. 15-1 LVDS TX Block Diagram

15.3 Function Description

15.3.1 Video Data Processing

LVDS PHY implements LVDS TIA/EIA protocol. LVDS PHY contains four 7-bit parallel-load serial-out shift registers, a 7X clock PLL, and five Low Voltage Differential Signaling (LVDS) line drivers in a single integrated circuit. These functions allow 28 bits of single-ended LVTTL data to be synchronously transmitted over five balanced-pair conductors for receipt by a compatible receiver.
When transmitting, parallel data, pdata0/1/2/3 are each loaded into registers upon the

edge of the input clock signal (CLKIN). The frequency of CLKIN is multiplied seven times, and then used to unload the data registers in7-bit slices and serially. The four serial streams and a phase-locked clock (CLKOUT) are then output to LVDS output drivers. The frequency of CLKOUT is the same as the input clock, CLKIN.

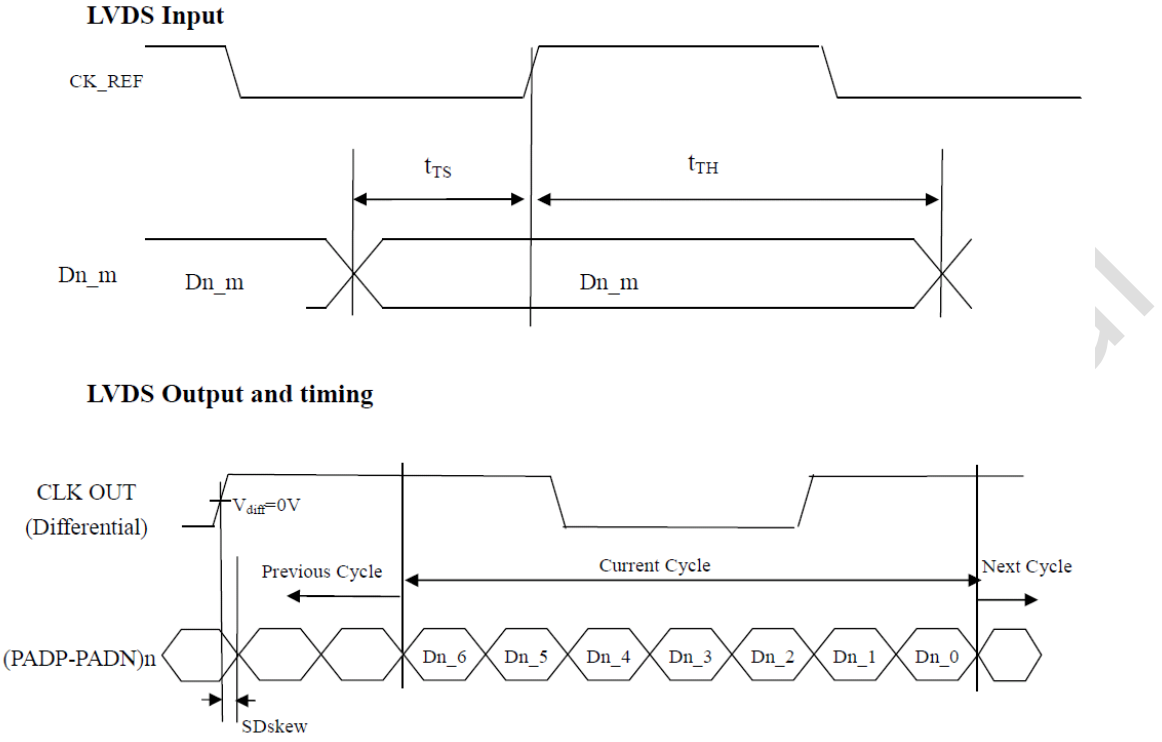


Fig. 15-2 LVDS TX Interface Timing

15.3.2 LVDS Format

LVDS FORMAT converts LCDC RGB interface to LVDS format data, only support 8-bit/6-bit mode. Table 15-1 is LVDSFORMAT input data format.

Table 15-2 LVDS Address Mapping

Serial channel	Data Bits	8-Bit			6-Bit	4-Bit	
		Format-1	Format-2	Format-3		Non-Linear Step Size	Linear Step Size
Y0	D0	R0	R2	R2	R0	R2	VCC
	D1	R1	R3	R3	R1	R3	GND
	D2	R2	R4	R4	R2	R0	R0
	D3	R3	R5	R5	R3	R1	R1
	D4	R4	R6	R6	R4	R2	R2
	D6	R5	R7	R7	R5	R3	R3
	D7	G0	G2	G2	G0	G2	VCC
Y1	D8	G1	G3	G3	G1	G3	GND
	D9	G2	G4	G4	G2	G0	G0
	D12	G3	G5	G5	G3	G1	G1
	D13	G4	G6	G6	G4	G2	G2
	D14	G5	G7	G7	G5	G3	G3
	D15	B0	B2	B2	B0	B2	VCC
	D18	B1	B3	B3	B1	B3	GND
Y2	D19	B2	B4	B4	B2	B0	B0
	D20	B3	B5	B5	B3	B1	B1
	D21	B4	B6	B6	B4	B2	B2

	D22	B5	B7	B7	B5	B3	B3
	D24	HSYNC	HSYNC	HSYNC	HSYNC	HSYNC	HSYNC
	D25	VSYNC	VSYNC	VSYNC	VSYNC	VSYNC	VSYNC
	D26	ENABLE	ENABLE	ENABLE	ENABLE	ENABLE	ENABLE
Y3	D27	R6	R0	GND	GND	GND	GND
	D5	R7	R1	GND	GND	GND	GND
	D10	G6	G0	GND	GND	GND	GND
	D11	G7	G1	GND	GND	GND	GND
	D16	B6	B0	GND	GND	GND	GND
	D17	B7	B1	GND	GND	GND	GND
	D23	RSVD	RSVD	GND	GND	GND	GND
CLKOUIT	CLKIN	CLK	CLK	CLK	CLK	CLK	CLK

15.4 Register Description

15.4.1 Register Summary

Name	Offset	Size	Reset Value	Description
MIPI_reg03	0x000c	B	0x03	Register03
MIPI_reg04	0x0010	B	0x7d	Register04
MIPI_rege0	0x0380	B	0x45	Registere0
MIPI_rege1	0x0384	B	0x12	Registere1
MIPI_rege3	0x038c	B	0x01	Registere3
GRF_LVDS_CON0	0X0150	W	0x0100	GRF_LVDS_CON0

*Notes:Size:**B**- Byte (8 bits) access, **HW**- Half WORD (16 bits) access, **W**-WORD (32 bits) access*

15.4.2 Detail Register Description

MIPI_reg03
Address: Operational Base + offset (0x000c)
Register03

Bit	Attr	Reset Value	Description
7:6	RW	0x0	reserved
5	RW	0x0	Reg_fbdiv[8] PLL input reference clock divider
4:0	RW	0x3	Reg_prediv[4:0] Integer value programmed into feedback divider

MIPI_reg04
Address: Operational Base + offset (0x0010)
Register04

Bit	Attr	Reset Value	Description
7:0	RW	0x7d	Reg_fbdiv[7:0] PLL input reference clock divider

MIPI_rege0
Address: Operational Base + offset (0x0380)
Registere0

Bit	Attr	Reset Value	Description
7:3	RW	0x00	reserved
2	RW	0x1	Digital internal reset Active low, default high
1	RW	0x0	reserved
0	RW	0x1	Selection for MSB and LSB 1'b1: MSB 1'b0: LSB

MIPI_rege1
Address: Operational Base + offset (0x0384)
Registere1

Bit	Attr	Reset Value	Description
7	RW	0x0	Digital internal enable Active high, default low.
6:0	RW	0x12	reserved

MIPI_rege3
Address: Operational Base + offset (0x038c)
Registere3

Bit	Attr	Reset Value	Description
7:3	RW	0x0	reserved
2	RW	0x0	TTL mode enable 1'b1: enable TTL mode 1'b0: disable TTL mode
1	RW	0x0	LVDS mode enable 1'b1: enable LVDS mode 1'b0: disable LVDS mode
0	RW	0x1	Mipi mode enable 1'b1: enable mipi mode 1'b0: disable mipi mode

GRF_LVDS_CON0
Address: Operational Base + offset (0x0150)
GRF_LVDS_CON0

Bit	Attr	Reset Value	Description
31:16	RW	0x0	Write enable When bit 16=1, bit 0 can be written by software . When bit 16=0, bit 0 cannot be written by software; When bit 17=1, bit 1 can be written by software . When bit 17=0, bit 1 cannot be written by software; ..... When bit 31=1, bit 15 can be written by software . When bit 31=0, bit 15 cannot be written by software;
13	RW	0x0	Mipiphylane3forcexmode 1'b1: enable 1'b0: disable
12	RW	0x0	Mipiphylane2 forcexmode 1'b1: enable 1'b0: disable
11	RW	0x0	Mipiphylane1 forcexmode 1'b1: enable 1'b0: disable
10	RW	0x0	Mipiphylane0 forcexmode 1'b1: enable 1'b0: disable
9	RW	0x0	Mipidsiforcexmode 1'b1: enable 1'b0: disable
8	RW	0x0	Mipiphylane0turndisable 1'b1: enable 1'b0: disable
7	RW	0x0	Mipiphyttlmode 1'b1: enable 1'b0: disable
6	RW	0x0	lvds_mode 1'b1: enable 1'b0: disable
5	RW	0x0	Mipictrldpicoloorm 1'b1: enable 1'b0: disable
4	RW	0x0	Mipictrldpishutdown 1'b1: enable 1'b0: disable
3	RW	0x0	Lvdsmsbsel 1'b0: MSB is on D0 1'b1: MSB is on D7

Bit	Attr	Reset Value	Description
2:1	RW	0x0	Lvdsselect 2’b00: 8bit mode format-1 2’b01: 8bit mode format-2 2’b10: 8bit mode format-3 2’b11: 6bit mode
0	RW	0x0	Ebc and lcdc_datasel 1'b0 : lcdc_data[9:0] 1'b1 : ebc

15.5 Interface Description

15.5.1 Video Input Source

In this chip, the LVDS TX video source comes from VOP.

15.6 Application Notes

15.6.1 LVDS mode

When used in LVDS mode, LVDS transmitter source from VOP, vop_dclk need get invert, then input to LVDS.
When lvds panel is LSB receive mode, lvds_msbsel =1, otherwiselvds_msbsel =0.
When LVDS output format is 8bit mode format-1/8bit mode format-2, configure grf_lvds_con0 in 24-bit color mode.
When LVDS output format is 8bit mode format-3/6bit mode,configure grf_lvds_con0 in 18-bit color mode.

Step1:
configure GRF_LVDS_CON0
configure MIPI PHY:
Step2:
configure PLL
MIPI_reg03 = 0x01;
MIPI_reg04 = 0x07;
Step3:
MIPI_rege0 = 0x45;
MIPI_rege1 = 0x92;
MIPI_rege3 = 0x02;

15.6.2 TTL mode

When used in lvttl mode, lvds_dclk and rgb_dclk need get invert.
ebc_sel = 1’b0: LVDS transmitter source from lcdc_data[9:0].
ebc_sel = 1’b1: LVDS transmitter source from EBC.

Step1: configure GRF_LVDS_CON0
Step2: configure PLL
MIPI_reg03 = 0x01;
MIPI_reg04 = 0x07;
Step3: MIPI_rege0 = 0x45;
MIPI_rege1 = 0x92;
MIPI_rege3 = 0x04;

Chapter 16 MIPI D-PHY

16.1 Overview

The MIPI D-PHY integrates a MIPI® V1.0 compatible PHY that supports up to 1GHz high speed data receiver, plus a MIPI® low-power low speed transceiver that supports data transfer in the bi-directional mode. It supports the full specifications described in V1.0 of the D-PHY spec. The D-PHY is built in with a standard digital interface to talk to MIPI Host controller. The architecture supports connection of multiple data lanes in parallel – up to 4 data lanes can be connected to increase the total through-put, customizable to user determined configurations. The MIPI D-PHY supports the electrical portion of MIPI D-PHY V1.0 standard, covering all transmission modes (ULP/LP/HS).
The MIPI D-PHY supports the following features:

- Mixed-signal D-PHY mixed-signal hard-macro- LS Transmitter and LS/HS Receiver solution
- Designed to MIPI® v1.0 Specifications
- Integrated PHY Protocol Interface (PPI) supports interface to CSI, DSI and UniPro™ MIPI® protocols
- 1.0GHz maximum data transfer rate per lane
- Expandable to support 4 data lanes, providing up to 4Gbps transfer rate
- HS, LP and ULPS modes supported
- 10Mbps per lane in low-power mode
- Unidirectional and bi-directional modes supported
- Automatic termination control for HS and LP modes
- Low-Power dissipation: HS less than 3mA/Lane
- Tx/Rx Buffers with tunable On-Die-Termination and advanced equalization.
- Embedded ESD, boundary scan support logic.

16.2 Block Diagram

The MIPI D-PHY configuration contains one Clock Lane Module and four Data Lane Modules. Each of these PHY Lane Modules communicates via two Lines to a complementary part at the other side of the Lane Interconnect. The following diagram shows the D-PHY architecture.

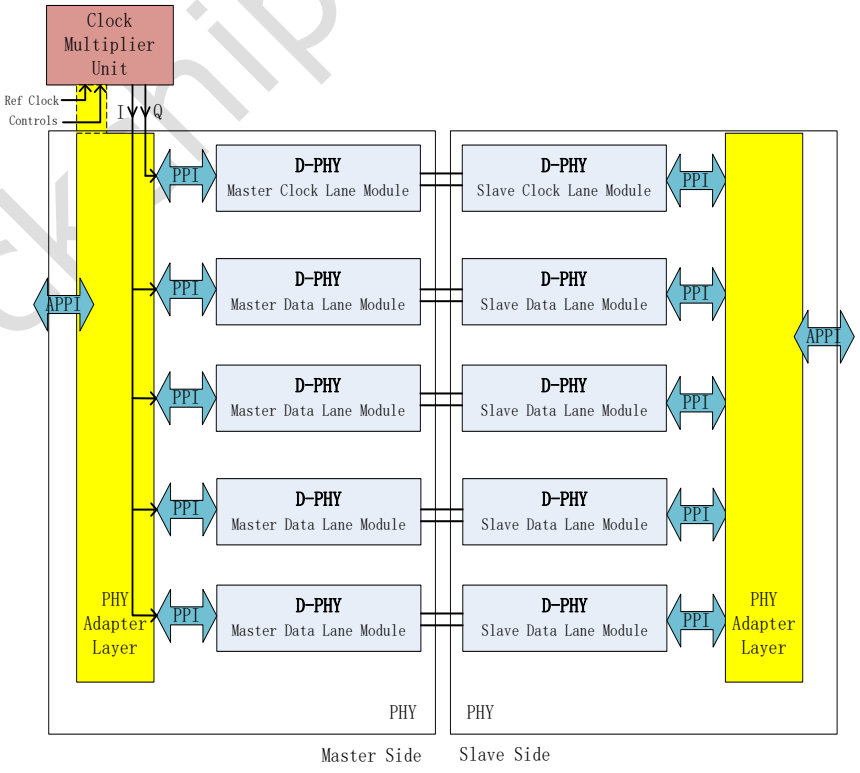


Fig. 16-1 MIPI D-PHY simplified Block diagram with master to slave

The following diagram shows a Universal Lane Module Diagram with a global overview of internal functionality of the CIL function. This Universal Module can be used for all Lane types. The requirements for the 'Control and Interface Logic' (CIL) function depend on the Lane type and Lane side.

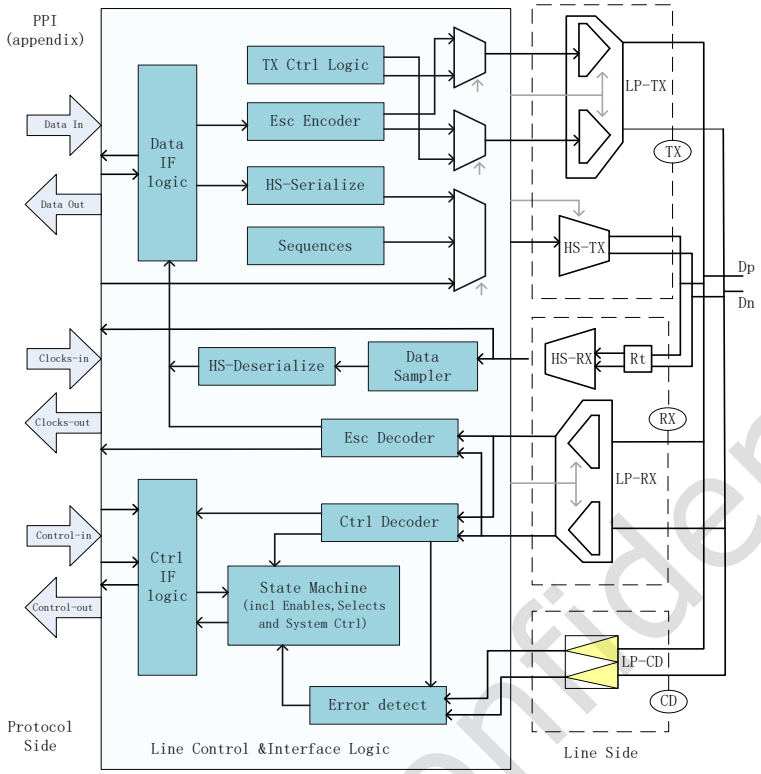


Fig. 16-2 MIPI D-PHY V1.0 detailed block diagram

16.3 Function Description

The MIPI D-PHY transceiver is designed to reliably transmit HS and LP/ULP data/clock over the channel and recover the MIPI LP data stream from any MIPI input signal. It consists of 4 data transceiver paths and 1 clock transmitting path. For each data lane a HS transmitter and a LP transceiver is necessary to transmit/recover the data streams, for the clock lane, a HS/LP transmitter is designed to output the high speed clock signal over the channel.
A HS differential signal driven on the Dp and Dn pins is generated by a differential output driver. For reference, Dp is considered as the positive side and Dn as the negative side.
High speed current switches are designed to output data streams over the channels.
The Low-Power receiver is an un-terminated, single-ended receiver circuit. LP receiver is used to detect the Low-Power state on each pin. For high robustness, the LP receiver can filter out noise pulses and RF interference. Furthermore, any spikes with a pulse width smaller than 20ns will be rejected.
Contention Detector (LP-CD) is designed in Data Lane to monitor the line voltage on each Low-Power signal. The LP-CD is used to detect an LP low fault when the LP transmitter is driving low and the pin voltage is greater than 450mV.
The Low-Power transmitter is a slew-rate controlled push-pull driver. The minimum pull-down and pull-up impedance of LP driver is 110 ohm. At the same time tunable slew rate control logic is available for eye pattern requirement.

16.4 Register Description

This section describes the control/status registers of the design. While you are reading this chapter please note that the offset address[7:0] is distributed two parts, one from the bit7 to bit5 is the first address, the other from the bit4 to bit0 is the second address. When you configure the registers, you must set both of them. The Clock Lane and Data Lane use the

same registers with the same second address, but the first address is different. Its apb base address is 0xc00.

16.4.1 Registers Summary

Name	Offset	Size	Reset Value	Description
MIPIPHY_MIPIPHY_REG0	0x0000	W	0x00000001	mipi phy register 0
MIPIPHY_MIPIPHY_REG1	0x0004	W	0x00000003	mipi phy register 1
MIPIPHY_MIPIPHY_REG3	0x000c	W	0x00000003	mipi phy register 3
MIPIPHY_MIPIPHY_REG4	0x0010	W	0x0000007d	mipi phy register 4
MIPIPHY_MIPIPHY_REG20	0x0080	W	0x00000001	mipi phy register 20
MIPIPHY_MIPIPHY_REG40	0x0100	W	0x0000000b	mipi phy register 40
MIPIPHY_MIPIPHY_REG45	0x0114	W	0x00000005	mipi phy register 45
MIPIPHY_MIPIPHY_REG46	0x0118	W	0x00000000	mipi phy register 46
MIPIPHY_MIPIPHY_REG47	0x011c	W	0x00000000	mipi phy register 47
MIPIPHY_MIPIPHY_REG48	0x0120	W	0x00000000	mipi phy register 48
MIPIPHY_MIPIPHY_REG49	0x0124	W	0x00000000	mipi phy register 49
MIPIPHY_MIPIPHY_REG4A	0x0128	W	0x00000000	mipi phy register 4a
MIPIPHY_MIPIPHY_REG4B	0x012c	W	0x00000000	mipi phy register 4b
MIPIPHY_MIPIPHY_REG4C	0x0130	W	0x00000000	mipi phy register 4c
MIPIPHY_MIPIPHY_REG4D	0x0134	W	0x00000000	mipi phy register 4d
MIPIPHY_MIPIPHY_REG4E	0x0138	W	0x00000000	mipi phy register 4e
MIPIPHY_MIPIPHY_REG50	0x0140	W	0x00000000	mipi phy register 50
MIPIPHY_MIPIPHY_REG51	0x0144	W	0x00000000	mipi phy register 51
MIPIPHY_MIPIPHY_REG52	0x0148	W	0x00000000	mipi phy register 52
MIPIPHY_MIPIPHY_REG60	0x0180	W	0x0000000b	mipi phy register 60
MIPIPHY_MIPIPHY_REG65	0x0194	W	0x00000005	mipi phy register 65
MIPIPHY_MIPIPHY_REG66	0x0198	W	0x00000000	mipi phy register 66
MIPIPHY_MIPIPHY_REG67	0x019c	W	0x00000000	mipi phy register 67
MIPIPHY_MIPIPHY_REG68	0x01a0	W	0x00000000	mipi phy register 68
MIPIPHY_MIPIPHY_REG69	0x01a4	W	0x00000000	mipi phy register 69
MIPIPHY_MIPIPHY_REG6A	0x01a8	W	0x00000000	mipi phy register 6a
MIPIPHY_MIPIPHY_REG6B	0x01ac	W	0x00000000	mipi phy register 6b
MIPIPHY_MIPIPHY_REG6C	0x01b0	W	0x00000000	mipi phy register 6c
MIPIPHY_MIPIPHY_REG6D	0x01b4	W	0x00000000	mipi phy register 6d
MIPIPHY_MIPIPHY_REG6E	0x01b8	W	0x00000000	mipi phy register 6e
MIPIPHY_MIPIPHY_REG70	0x01c0	W	0x00000000	mipi phy register 70
MIPIPHY_MIPIPHY_REG71	0x01c4	W	0x00000000	mipi phy register 71
MIPIPHY_MIPIPHY_REG72	0x01c8	W	0x00000000	mipi phy register 72
MIPIPHY_MIPIPHY_REG80	0x0200	W	0x0000000b	mipi phy register 80
MIPIPHY_MIPIPHY_REG85	0x0214	W	0x00000005	mipi phy register 85
MIPIPHY_MIPIPHY_REG86	0x0218	W	0x00000000	mipi phy register 86
MIPIPHY_MIPIPHY_REG87	0x021c	W	0x00000000	mipi phy register 87
MIPIPHY_MIPIPHY_REG88	0x0220	W	0x00000000	mipi phy register 88
MIPIPHY_MIPIPHY_REG89	0x0224	W	0x00000000	mipi phy register 89
MIPIPHY_MIPIPHY_REG8A	0x0228	W	0x00000000	mipi phy register 8a

Name	Offset	Size	Reset Value	Description
MIPIPHY_MIPIPHY_REG8B	0x022c	W	0x00000000	mipi phy register 8b
MIPIPHY_MIPIPHY_REG8C	0x0230	W	0x00000000	mipi phy register 8c
MIPIPHY_MIPIPHY_REG8D	0x0234	W	0x00000000	mipi phy register 8d
MIPIPHY_MIPIPHY_REG8E	0x0238	W	0x00000000	mipi phy register 8e
MIPIPHY_MIPIPHY_REG90	0x0240	W	0x00000000	mipi phy register 90
MIPIPHY_MIPIPHY_REG91	0x0244	W	0x00000000	mipi phy register 91
MIPIPHY_MIPIPHY_REG92	0x0248	W	0x00000000	mipi phy register 92
MIPIPHY_MIPIPHY_REGA0	0x0280	W	0x0000000b	mipi phy register a0
MIPIPHY_MIPIPHY_REGA5	0x0294	W	0x00000005	mipi phy register a5
MIPIPHY_MIPIPHY_REGA6	0x0298	W	0x00000000	mipi phy register a6
MIPIPHY_MIPIPHY_REGA7	0x029c	W	0x00000000	mipi phy register a7
MIPIPHY_MIPIPHY_REGA8	0x02a0	W	0x00000000	mipi phy register a8
MIPIPHY_MIPIPHY_REGA9	0x02a4	W	0x00000000	mipi phy register a9
MIPIPHY_MIPIPHY_REGAA	0x02a8	W	0x00000000	mipi phy register aa
MIPIPHY_MIPIPHY_REGAB	0x02ac	W	0x00000000	mipi phy register ab
MIPIPHY_MIPIPHY_REGAC	0x02b0	W	0x00000000	mipi phy register ac
MIPIPHY_MIPIPHY_REGAD	0x02b4	W	0x00000000	mipi phy register ad
MIPIPHY_MIPIPHY_REGAE	0x02b8	W	0x00000000	mipi phy register ae
MIPIPHY_MIPIPHY_REGB0	0x02c0	W	0x00000000	mipi phy register b0
MIPIPHY_MIPIPHY_REGB1	0x02c4	W	0x00000000	mipi phy register b1
MIPIPHY_MIPIPHY_REGB2	0x02c8	W	0x00000000	mipi phy register b2
MIPIPHY_MIPIPHY_REGC0	0x0300	W	0x0000000b	mipi phy register c0
MIPIPHY_MIPIPHY_REGC5	0x0314	W	0x00000005	mipi phy register c5
MIPIPHY_MIPIPHY_REGC6	0x0318	W	0x00000000	mipi phy register c6
MIPIPHY_MIPIPHY_REGC7	0x031c	W	0x00000000	mipi phy register c7
MIPIPHY_MIPIPHY_REGC8	0x0320	W	0x00000000	mipi phy register c8
MIPIPHY_MIPIPHY_REGC9	0x0324	W	0x00000000	mipi phy register c9
MIPIPHY_MIPIPHY_REGCA	0x0328	W	0x00000000	mipi phy register ca
MIPIPHY_MIPIPHY_REGCB	0x032c	W	0x00000000	mipi phy register cb
MIPIPHY_MIPIPHY_REGCC	0x0330	W	0x00000000	mipi phy register cc
MIPIPHY_MIPIPHY_REGCD	0x0334	W	0x00000000	mipi phy register cd
MIPIPHY_MIPIPHY_REGCE	0x0338	W	0x00000000	mipi phy register ce
MIPIPHY_MIPIPHY_REGD0	0x0340	W	0x00000000	mipi phy register d0
MIPIPHY_MIPIPHY_REGD1	0x0344	W	0x00000000	mipi phy register d1
MIPIPHY_MIPIPHY_REGD2	0x0348	W	0x00000000	mipi phy register d2
MIPIPHY_MIPIPHY_REGE0	0x0380	W	0x00000000	mipi phy register e0
MIPIPHY_MIPIPHY_REGEA	0x03a8	W	0x00000000	mipi phy register ea

*Notes:Size:**B**- Byte (8 bits) access, **HW**- Half WORD (16 bits) access, **W**-WORD (32 bits) access*

16.4.2 Detail Register Description

MIPIPHY_MIPIPHY_REG0

Address: Operational Base + offset (0x0000)
mipi phy register 0

Bit	Attr	Reset Value	Description
31:7	RO	0x0	reserved

Bit	Attr	Reset Value	Description
6	RW	0x0	lane_en_ck lane_en_ck 1: enable 0: disable
5	RW	0x0	lane_en_3 lane_en_3 1: enable 0: disable
4	RW	0x0	lane_en_2 lane_en_2 1: enable 0: disable
3	RW	0x0	lane_en_1 lane_en_1 1: enable 0: disable
2	RW	0x0	lane_en_0 lane_en_0 1: enable 0: disable
1	RW	0x0	reserved1 reserved
0	RO	0x1	reserved reserved

MIPIPHY_MIPIPHY_REG1
Address: Operational Base + offset (0x0004)
mipi phy register 1

Bit	Attr	Reset Value	Description
31:3	RO	0x0	reserved
2	RW	0x0	reg_da_syncrst reg_da_syncrst 1: reset 0: normal
1	RW	0x1	reg_da_ldopd reg_da_ldopd 1: power down 0: power on
0	RW	0x1	reg_da_pllpd reg_da_pllpd 1: power down 0: power on

MIPIPHY_MIPIPHY_REG3
Address: Operational Base + offset (0x000c)

mipi phy register 3

Bit	Attr	Reset Value	Description
31:6	RO	0x0	reserved
5	RW	0x0	reg_fbdiv reg_fbdiv[8] PLL input reference clock divider
4:0	RW	0x03	reg_prediv reg_prediv[4:0] Integer value programmed into feedback divider

MIPIPHY_MIPIPHY_REG4

Address: Operational Base + offset (0x0010)

mipi phy register 4

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:0	RW	0x7d	reg_fbdiv reg_fbdiv[7:0] PLL input reference clock divider

MIPIPHY_MIPIPHY_REG20

Address: Operational Base + offset (0x0080)

mipi phy register 20

Bit	Attr	Reset Value	Description
31:1	RO	0x0	reserved
0	RW	0x1	reg_dig_rstn reg_dig_rstn 1: normal 0: reset

MIPIPHY_MIPIPHY_REG40

Address: Operational Base + offset (0x0100)

mipi phy register 40

Bit	Attr	Reset Value	Description
31:4	RO	0x0	reserved

Bit	Attr	Reset Value	Description
3:0	RW	0xb	reg_ths_settle Clock Lane Configure the count time of the THS-SETTLE by protocol. After the count done, D-PHY will begin to receive the high speed data. (Can be configured from 4'h0 to 4'hd) 4'b0000 80-110 MHz 4'b0001 110-150 MHz 4'b0010 150-200 MHz 4'b0011 200-250 MHz 4'b0100 250-300 MHz 4'b0101 300-400 MHz 4'b0110 400-500 MHz 4'b0111 500-600 MHz 4'b1000 600-700 MHz 4'b1001 700-800 MHz 4'b1010 800-1000 MHz 4'b1011 additional adjust 4'b1100 additional adjust 4'b1101 additional adjust 4'b1110 additional adjust

MIPIPHY_MIPIPHY_REG45
Address: Operational Base + offset (0x0114)
mipi phy register 45

Bit	Attr	Reset Value	Description
31:6	RO	0x0	reserved
5:0	RW	0x05	reg_hs_tlpx Clock Lane The value of counter for HS Tlpx Time （>=Tlpx） = Tpin_txbyteclkhs * value

MIPIPHY_MIPIPHY_REG46
Address: Operational Base + offset (0x0118)
mipi phy register 46

Bit	Attr	Reset Value	Description
31:7	RO	0x0	reserved
6:0	RW	0x00	reg_hs_ths_prepare Clock Lane The value of counter for HS Ths-prepare For clock lane, Ths-prepare(38ns~95ns) For data lane, Ths-prepare(40ns+4UI~85ns+6UI) =Ttxddrclkhs*value

MIPIPHY_MIPIPHY_REG47
Address: Operational Base + offset (0x011c)
mipi phy register 47

Bit	Attr	Reset Value	Description
31:6	RO	0x0	reserved
5:0	RW	0x00	reg_hs_the_zero Clock Lane The value of counter for HS Ths-zero For clock lane, Ths-prepare+Ths-zero (>=300ns) For data lane, Ths-prepare+Ths-zero (>= 145 ns + 10*UI) = Tpin_txbyteclkhs*value For clock lane, S_HSTXTHSZERO[5:0] = 6'b100000 For data lane, S_HSTXTHSZERO[5:0] = 6'b001001 Frequency(1/UI) Value(Decimal) 80 -110 MHz 3 110-150 MHz 4 150-200 MHz 4 200-250 MHz 5 250-300 MHz 6 300-400 MHz 7 400-500 MHz 8 500-600 MHz 10 600-700 MHz 11 700-800 MHz 12 800-1000 MHz 15

MIPIPHY_MIPIPHY_REG48
Address: Operational Base + offset (0x0120)
mipi phy register 48

Bit	Attr	Reset Value	Description
31:7	RO	0x0	reserved

Bit	Attr	Reset Value	Description
6:0	RW	0x00	reg_hs_ths_trail Clock Lane The value of counter for HS Ths-trail For clock lane, Ths-trail (>=60ns) For data lane,Ths-trail (>=max(8UI, 60ns+4UI) = Tad_txddrclkhs_i* value For clock lane, S_HXTXTHSTRAIL[6:0] = 7'b0100111 For data lane, S_HXTXTHSTRAIL[6:0] = 7'b0100111 Frequency(1/UI) Value(Decimal) 80 -110 MHz 12 110-150 MHz 13 150-200 MHz 17 200-250 MHz 20 250-300 MHz 24 300-400 MHz 29 400-500 MHz 35 500-600 MHz 41 600-700 MHz 49 700-800 MHz 52 800-1000 MHz 64

MIPIPHY_MIPIPHY_REG49
Address: Operational Base + offset (0x0124)
mipi phy register 49

Bit	Attr	Reset Value	Description
31:5	RO	0x0	reserved
4:0	RW	0x00	reg_hs_ths_exit Clock Lane The value of counter for HS Ths-exit Ths-exit = Tpin_txbyteclkhs*value

MIPIPHY_MIPIPHY_REG4A
Address: Operational Base + offset (0x0128)
mipi phy register 4a

Bit	Attr	Reset Value	Description
31:4	RO	0x0	reserved
3:0	RW	0x0	reg_hs_tclk_post Clock Lane The value of counter for HS Tclk-post Tclk-post =Tpin_txbyteclkhs*value

MIPIPHY_MIPIPHY_REG4B
Address: Operational Base + offset (0x012c)
mipi phy register 4b

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:0	RW	0x00	reserved Clock Lane reserved

MIPIPHY_MIPIPHY_REG4C
Address: Operational Base + offset (0x0130)
mipi phy register 4c

Bit	Attr	Reset Value	Description
31:2	RO	0x0	reserved
1:0	RW	0x0	reg_hs_twakup Clock Lane The value[9:8] of counter for HS Twakup also see REG4D

MIPIPHY_MIPIPHY_REG4D
Address: Operational Base + offset (0x0134)
mipi phy register 4d

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:0	RW	0x00	reg_hs_twakup Clock Lane The value[7:0] of counter for HS Twakup Twakup for ulpm, Twakup = Tpin_sys_clk*value[9:0]

MIPIPHY_MIPIPHY_REG4E
Address: Operational Base + offset (0x0138)
mipi phy register 4e

Bit	Attr	Reset Value	Description
31:4	RO	0x0	reserved
3:0	RW	0x0	reg_hs_tclk_pre Clock Lane The value of counter for HS Tclk-pre Tclk-pre for clock lane Tclk-pre = Tpin_txbyteclkhs*value

MIPIPHY_MIPIPHY_REG50
Address: Operational Base + offset (0x0140)
mipi phy register 50

Bit	Attr	Reset Value	Description
31:6	RO	0x0	reserved

Bit	Attr	Reset Value	Description
5:0	RW	0x00	reg_hs_tta_go Clock Lane The value of counter for HS Tta-go Tta-go for turnaround Tta-go = Ttxclkesc*value

MIPIPHY_MIPIPHY_REG51
Address: Operational Base + offset (0x0144)
mipi phy register 51

Bit	Attr	Reset Value	Description
31:6	RO	0x0	reserved
5:0	RW	0x00	reg_hs_tta_sure Clock Lane The value of counter for HS Tta-sure Tta-sure for turnaround Tta-sure = Ttxclkesc*value

MIPIPHY_MIPIPHY_REG52
Address: Operational Base + offset (0x0148)
mipi phy register 52

Bit	Attr	Reset Value	Description
31:6	RO	0x0	reserved
5:0	RW	0x00	reg_hs_tta_wait Clock Lane The value of counter for HS Tta-wait Tta-wait for turnaround Interval from receiving ppi turnaround request to sending esc request. Tta-wait = Ttxclkesc*value

MIPIPHY_MIPIPHY_REG60
Address: Operational Base + offset (0x0180)
mipi phy register 60

Bit	Attr	Reset Value	Description
31:4	RO	0x0	reserved

Bit	Attr	Reset Value	Description
3:0	RW	0xb	reg_ths_settle Data0 Lane Configure the count time of the THS-SETTLE by protocol. After the count done, D-PHY will begin to receive the high speed data. (Can be configured from 4'h0 to 4'hd) 4'b0000 80-110 MHz 4'b0001 110-150 MHz 4'b0010 150-200 MHz 4'b0011 200-250 MHz 4'b0100 250-300 MHz 4'b0101 300-400 MHz 4'b0110 400-500 MHz 4'b0111 500-600 MHz 4'b1000 600-700 MHz 4'b1001 700-800 MHz 4'b1010 800-1000 MHz 4'b1011 additional adjust 4'b1100 additional adjust 4'b1101 additional adjust 4'b1110 additional adjust

MIPIPHY_MIPIPHY_REG65
Address: Operational Base + offset (0x0194)
mipi phy register 65

Bit	Attr	Reset Value	Description
31:6	RO	0x0	reserved
5:0	RW	0x05	reg_hs_tlpx Data0 Lane The value of counter for HS Tlpx Time （>=Tlpx） = Tpin_txbyteclkhs * value

MIPIPHY_MIPIPHY_REG66
Address: Operational Base + offset (0x0198)
mipi phy register 66

Bit	Attr	Reset Value	Description
31:7	RO	0x0	reserved
6:0	RW	0x00	reg_hs_ths_prepare Data0 Lane The value of counter for HS Ths-prepare For clock lane, Ths-prepare(38ns~95ns) For data lane, Ths-prepare(40ns+4UI~85ns+6UI) =Ttxddrclkhs*value

MIPIPHY_MIPIPHY_REG67
Address: Operational Base + offset (0x019c)
mipi phy register 67

Bit	Attr	Reset Value	Description
31:6	RO	0x0	reserved
5:0	RW	0x00	reg_hs_the_zero Data0 Lane The value of counter for HS Ths-zero For clock lane, Ths-prepare+Ths-zero (>=300ns) For data lane, Ths-prepare+Ths-zero (>= 145 ns + 10*UI) = Tpin_txbyteclkhs*value For clock lane, S_HSTXTHSZERO[5:0] = 6'b100000 For data lane, S_HSTXTHSZERO[5:0] = 6'b001001 Frequency(1/UI) Value(Decimal) 80 -110 MHz 3 110-150 MHz 4 150-200 MHz 4 200-250 MHz 5 250-300 MHz 6 300-400 MHz 7 400-500 MHz 8 500-600 MHz 10 600-700 MHz 11 700-800 MHz 12 800-1000 MHz 15

MIPIPHY_MIPIPHY_REG68
Address: Operational Base + offset (0x01a0)
mipi phy register 68

Bit	Attr	Reset Value	Description
31:7	RO	0x0	reserved

Bit	Attr	Reset Value	Description
6:0	RW	0x00	reg_hs_ths_trail Data0 Lane The value of counter for HS Ths-trail For clock lane, Ths-trail (>=60ns) For data lane,Ths-trail (>=max(8UI, 60ns+4UI) = Tad_txddrclkhs_i* value For clock lane, S_HXTXTHSTRAIL[6:0] = 7'b0100111 For data lane, S_HXTXTHSTRAIL[6:0] = 7'b0100111 Frequency(1/UI) Value(Decimal) 80 -110 MHz 12 110-150 MHz 13 150-200 MHz 17 200-250 MHz 20 250-300 MHz 24 300-400 MHz 29 400-500 MHz 35 500-600 MHz 41 600-700 MHz 49 700-800 MHz 52 800-1000 MHz 64

MIPIPHY_MIPIPHY_REG69
Address: Operational Base + offset (0x01a4)
mipi phy register 69

Bit	Attr	Reset Value	Description
31:5	RO	0x0	reserved
4:0	RW	0x00	reg_hs_ths_exit Data0 Lane The value of counter for HS Ths-exit Ths-exit = Tpin_txbyteclkhs*value

MIPIPHY_MIPIPHY_REG6A
Address: Operational Base + offset (0x01a8)
mipi phy register 6a

Bit	Attr	Reset Value	Description
31:4	RO	0x0	reserved
3:0	RW	0x0	reg_hs_tclk_post Data0 Lane The value of counter for HS Tclk-post Tclk-post =Tpin_txbyteclkhs*value

MIPIPHY_MIPIPHY_REG6B
Address: Operational Base + offset (0x01ac)
mipi phy register 6b

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:0	RW	0x00	reserved Data0 Lane

MIPIPHY_MIPIPHY_REG6C
Address: Operational Base + offset (0x01b0)
mipi phy register 6c

Bit	Attr	Reset Value	Description
31:2	RO	0x0	reserved
1:0	RW	0x0	reg_hs_twakup Data0 Lane The value[9:8] of counter for HS Twakup also see REG6D

MIPIPHY_MIPIPHY_REG6D
Address: Operational Base + offset (0x01b4)
mipi phy register 6d

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:0	RW	0x00	reg_hs_twakup Data0 Lane The value[7:0] of counter for HS Twakup Twakup for ulpm, Twakup = Tpin_sys_clk*value[9:0]

MIPIPHY_MIPIPHY_REG6E
Address: Operational Base + offset (0x01b8)
mipi phy register 6e

Bit	Attr	Reset Value	Description
31:4	RO	0x0	reserved
3:0	RW	0x0	reg_hs_tclk_pre Data0 Lane The value of counter for HS Tclk-pre Tclk-pre for clock lane Tclk-pre = Tpin_txbyteclkhs*value

MIPIPHY_MIPIPHY_REG70
Address: Operational Base + offset (0x01c0)
mipi phy register 70

Bit	Attr	Reset Value	Description
31:6	RO	0x0	reserved

Bit	Attr	Reset Value	Description
5:0	RW	0x00	reg_hs_tta_go Data0 Lane The value of counter for HS Tta-go Tta-go for turnaround Tta-go = Ttxclkesc*value

MIPIPHY_MIPIPHY_REG71
Address: Operational Base + offset (0x01c4)
mipi phy register 71

Bit	Attr	Reset Value	Description
31:6	RO	0x0	reserved
5:0	RW	0x00	reg_hs_tta_sure Data0 Lane The value of counter for HS Tta-sure Tta-sure for turnaround Tta-sure = Ttxclkesc*value

MIPIPHY_MIPIPHY_REG72
Address: Operational Base + offset (0x01c8)
mipi phy register 72

Bit	Attr	Reset Value	Description
31:6	RO	0x0	reserved
5:0	RW	0x00	reg_hs_tta_wait Data0 Lane The value of counter for HS Tta-wait Tta-wait for turnaround Interval from receiving ppi turnaround request to sending esc request. Tta-wait = Ttxclkesc*value

MIPIPHY_MIPIPHY_REG80
Address: Operational Base + offset (0x0200)
mipi phy register 80

Bit	Attr	Reset Value	Description
31:4	RO	0x0	reserved

Bit	Attr	Reset Value	Description
3:0	RW	0xb	reg_ths_settle Data1 Lane Configure the count time of the THS-SETTLE by protocol. After the count done, D-PHY will begin to receive the high speed data. (Can be configured from 4'h0 to 4'hd) 4'b0000 80-110 MHz 4'b0001 110-150 MHz 4'b0010 150-200 MHz 4'b0011 200-250 MHz 4'b0100 250-300 MHz 4'b0101 300-400 MHz 4'b0110 400-500 MHz 4'b0111 500-600 MHz 4'b1000 600-700 MHz 4'b1001 700-800 MHz 4'b1010 800-1000 MHz 4'b1011 additional adjust 4'b1100 additional adjust 4'b1101 additional adjust 4'b1110 additional adjust

MIPIPHY_MIPIPHY_REG85
Address: Operational Base + offset (0x0214)
mipi phy register 85

Bit	Attr	Reset Value	Description
31:6	RO	0x0	reserved
5:0	RW	0x05	reg_hs_tlpx Data1 Lane The value of counter for HS Tlpx Time （>=Tlpx） = Tpin_txbyteclkhs * value

MIPIPHY_MIPIPHY_REG86
Address: Operational Base + offset (0x0218)
mipi phy register 86

Bit	Attr	Reset Value	Description
31:7	RO	0x0	reserved
6:0	RW	0x00	reg_hs_ths_prepare Data1 Lane The value of counter for HS Ths-prepare For clock lane, Ths-prepare(38ns~95ns) For data lane, Ths-prepare(40ns+4UI~85ns+6UI) =Ttxddrclkhs*value

MIPIPHY_MIPIPHY_REG87
Address: Operational Base + offset (0x021c)
mipi phy register 87

Bit	Attr	Reset Value	Description
31:6	RO	0x0	reserved
5:0	RW	0x00	reg_hs_the_zero Data1 Lane The value of counter for HS Ths-zero For clock lane, Ths-prepare+Ths-zero (>=300ns) For data lane, Ths-prepare+Ths-zero (>= 145 ns + 10*UI) = Tpin_txbyteclkhs*value For clock lane, S_HSTXTHSZERO[5:0] = 6'b100000 For data lane, S_HSTXTHSZERO[5:0] = 6'b001001 Frequency(1/UI) Value(Decimal) 80 -110 MHz 3 110-150 MHz 4 150-200 MHz 4 200-250 MHz 5 250-300 MHz 6 300-400 MHz 7 400-500 MHz 8 500-600 MHz 10 600-700 MHz 11 700-800 MHz 12 800-1000 MHz 15

MIPIPHY_MIPIPHY_REG88
Address: Operational Base + offset (0x0220)
mipi phy register 88

Bit	Attr	Reset Value	Description
31:7	RO	0x0	reserved

Bit	Attr	Reset Value	Description
6:0	RW	0x00	reg_hs_ths_trail Data1 Lane The value of counter for HS Ths-trail For clock lane, Ths-trail (>=60ns) For data lane,Ths-trail (>=max(8UI, 60ns+4UI) = Tad_txddrclkhs_i* value For clock lane, S_HXTXTHSTRAIL[6:0] = 7'b0100111 For data lane, S_HXTXTHSTRAIL[6:0] = 7'b0100111 Frequency(1/UI) Value(Decimal) 80 -110 MHz 12 110-150 MHz 13 150-200 MHz 17 200-250 MHz 20 250-300 MHz 24 300-400 MHz 29 400-500 MHz 35 500-600 MHz 41 600-700 MHz 49 700-800 MHz 52 800-1000 MHz 64

MIPIPHY_MIPIPHY_REG89
Address: Operational Base + offset (0x0224)
mipi phy register 89

Bit	Attr	Reset Value	Description
31:5	RO	0x0	reserved
4:0	RW	0x00	reg_hs_ths_exit Data1 Lane The value of counter for HS Ths-exit Ths-exit = Tpin_txbyteclkhs*value

MIPIPHY_MIPIPHY_REG8A
Address: Operational Base + offset (0x0228)
mipi phy register 8a

Bit	Attr	Reset Value	Description
31:4	RO	0x0	reserved
3:0	RW	0x0	reg_hs_tclk_post Data1 Lane The value of counter for HS Tclk-post Tclk-post =Tpin_txbyteclkhs*value

MIPIPHY_MIPIPHY_REG8B
Address: Operational Base + offset (0x022c)
mipi phy register 8b

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:0	RW	0x00	reserved Data1 Lane reserved

MIPIPHY_MIPIPHY_REG8C
Address: Operational Base + offset (0x0230)
mipi phy register 8c

Bit	Attr	Reset Value	Description
31:2	RO	0x0	reserved
1:0	RW	0x0	reg_hs_twakup Data1 Lane The value[9:8] of counter for HS Twakup also see REG8D

MIPIPHY_MIPIPHY_REG8D
Address: Operational Base + offset (0x0234)
mipi phy register 8d

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:0	RW	0x00	reg_hs_twakup Data1 Lane The value[7:0] of counter for HS Twakup Twakup for ulpm, Twakup = Tpin_sys_clk*value[9:0]

MIPIPHY_MIPIPHY_REG8E
Address: Operational Base + offset (0x0238)
mipi phy register 8e

Bit	Attr	Reset Value	Description
31:4	RO	0x0	reserved
3:0	RW	0x0	reg_hs_tclk_pre Data1 Lane The value of counter for HS Tclk-pre Tclk-pre for clock lane Tclk-pre = Tpin_txbyteclkhs*value

MIPIPHY_MIPIPHY_REG90
Address: Operational Base + offset (0x0240)
mipi phy register 90

Bit	Attr	Reset Value	Description
31:6	RO	0x0	reserved

Bit	Attr	Reset Value	Description
5:0	RW	0x00	reg_hs_tta_go Data1 Lane The value of counter for HS Tta-go Tta-go for turnaround Tta-go = Ttxclkesc*value

MIPIPHY_MIPIPHY_REG91
Address: Operational Base + offset (0x0244)
mipi phy register 91

Bit	Attr	Reset Value	Description
31:6	RO	0x0	reserved
5:0	RW	0x00	reg_hs_tta_sure Data1 Lane The value of counter for HS Tta-sure Tta-sure for turnaround Tta-sure = Ttxclkesc*value

MIPIPHY_MIPIPHY_REG92
Address: Operational Base + offset (0x0248)
mipi phy register 92

Bit	Attr	Reset Value	Description
31:6	RO	0x0	reserved
5:0	RW	0x00	reg_hs_tta_wait Data1 Lane The value of counter for HS Tta-wait Tta-wait for turnaround Interval from receiving ppi turnaround request to sending esc request. Tta-wait = Ttxclkesc*value

MIPIPHY_MIPIPHY_REGA0
Address: Operational Base + offset (0x0280)
mipi phy register a0

Bit	Attr	Reset Value	Description
31:4	RO	0x0	reserved

Bit	Attr	Reset Value	Description
3:0	RW	0xb	reg_ths_settle Data2 Lane Configure the count time of the THS-SETTLE by protocol. After the count done, D-PHY will begin to receive the high speed data. (Can be configured from 4'h0 to 4'hd) 4'b0000 80-110 MHz 4'b0001 110-150 MHz 4'b0010 150-200 MHz 4'b0011 200-250 MHz 4'b0100 250-300 MHz 4'b0101 300-400 MHz 4'b0110 400-500 MHz 4'b0111 500-600 MHz 4'b1000 600-700 MHz 4'b1001 700-800 MHz 4'b1010 800-1000 MHz 4'b1011 additional adjust 4'b1100 additional adjust 4'b1101 additional adjust 4'b1110 additional adjust

MIPIPHY_MIPIPHY_REGA5
Address: Operational Base + offset (0x0294)
mipi phy register a5

Bit	Attr	Reset Value	Description
31:6	RO	0x0	reserved
5:0	RW	0x05	reg_hs_tlpx Data2 Lane The value of counter for HS Tlpx Time （>=Tlpx） = Tpin_txbyteclkhs * value

MIPIPHY_MIPIPHY_REGA6
Address: Operational Base + offset (0x0298)
mipi phy register a6

Bit	Attr	Reset Value	Description
31:7	RO	0x0	reserved
6:0	RW	0x00	reg_hs_ths_prepare Data2 Lane The value of counter for HS Ths-prepare For clock lane, Ths-prepare(38ns~95ns) For data lane, Ths-prepare(40ns+4UI~85ns+6UI) =Ttxddrclkhs*value

MIPIPHY_MIPIPHY_REGA7
Address: Operational Base + offset (0x029c)
mipi phy register a7

Bit	Attr	Reset Value	Description
31:6	RO	0x0	reserved
5:0	RW	0x00	reg_hs_the_zero Data2 Lane The value of counter for HS Ths-zero For clock lane, Ths-prepare+Ths-zero (>=300ns) For data lane, Ths-prepare+Ths-zero (>= 145 ns + 10*UI) = Tpin_txbyteclkhs*value For clock lane, S_HSTXTHSZERO[5:0] = 6'b100000 For data lane, S_HSTXTHSZERO[5:0] = 6'b001001 Frequency(1/UI) Value(Decimal) 80 -110 MHz 3 110-150 MHz 4 150-200 MHz 4 200-250 MHz 5 250-300 MHz 6 300-400 MHz 7 400-500 MHz 8 500-600 MHz 10 600-700 MHz 11 700-800 MHz 12 800-1000 MHz 15

MIPIPHY_MIPIPHY_REGA8
Address: Operational Base + offset (0x02a0)
mipi phy register a8

Bit	Attr	Reset Value	Description
31:7	RO	0x0	reserved

Bit	Attr	Reset Value	Description
6:0	RW	0x00	reg_hs_ths_trail Data2 Lane The value of counter for HS Ths-trail For clock lane, Ths-trail (>=60ns) For data lane,Ths-trail (>=max(8UI, 60ns+4UI) = Tad_txddrclkhs_i* value For clock lane, S_HXTXTHSTRAIL[6:0] = 7'b0100111 For data lane, S_HXTXTHSTRAIL[6:0] = 7'b0100111 Frequency(1/UI) Value(Decimal) 80 -110 MHz 12 110-150 MHz 13 150-200 MHz 17 200-250 MHz 20 250-300 MHz 24 300-400 MHz 29 400-500 MHz 35 500-600 MHz 41 600-700 MHz 49 700-800 MHz 52 800-1000 MHz 64

MIPIPHY_MIPIPHY_REGA9
Address: Operational Base + offset (0x02a4)
mipi phy register a9

Bit	Attr	Reset Value	Description
31:5	RO	0x0	reserved
4:0	RW	0x00	reg_hs_ths_exit Data2 Lane The value of counter for HS Ths-exit Ths-exit = Tpin_txbyteclkhs*value

MIPIPHY_MIPIPHY_REGAA
Address: Operational Base + offset (0x02a8)
mipi phy register aa

Bit	Attr	Reset Value	Description
31:4	RO	0x0	reserved
3:0	RW	0x0	reg_hs_tclk_post Data2 Lane The value of counter for HS Tclk-post Tclk-post =Tpin_txbyteclkhs*value

MIPIPHY_MIPIPHY_REGAB
Address: Operational Base + offset (0x02ac)
mipi phy register ab

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:0	RW	0x00	reserved Data2 Lane

MIPIPHY_MIPIPHY_REGAC
Address: Operational Base + offset (0x02b0)
mipi phy register ac

Bit	Attr	Reset Value	Description
31:2	RO	0x0	reserved
1:0	RW	0x0	reg_hs_twakup Data2 Lane The value[9:8] of counter for HS Twakup also see REGAD

MIPIPHY_MIPIPHY_REGAD
Address: Operational Base + offset (0x02b4)
mipi phy register ad

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:0	RW	0x00	reg_hs_twakup Data2 Lane The value[7:0] of counter for HS Twakup Twakup for ulpm, Twakup = Tpin_sys_clk*value[9:0]

MIPIPHY_MIPIPHY_REGAE
Address: Operational Base + offset (0x02b8)
mipi phy register ae

Bit	Attr	Reset Value	Description
31:4	RO	0x0	reserved
3:0	RW	0x0	reg_hs_tclk_pre Data2 Lane The value of counter for HS Tclk-pre Tclk-pre for clock lane Tclk-pre = Tpin_txbyteclkhs*value

MIPIPHY_MIPIPHY_REGB0
Address: Operational Base + offset (0x02c0)
mipi phy register b0

Bit	Attr	Reset Value	Description
31:6	RO	0x0	reserved

Bit	Attr	Reset Value	Description
5:0	RW	0x00	reg_hs_tta_go Data2 Lane The value of counter for HS Tta-go Tta-go for turnaround Tta-go = Ttxclkesc*value

MIPIPHY_MIPIPHY_REGB1
Address: Operational Base + offset (0x02c4)
mipi phy register b1

Bit	Attr	Reset Value	Description
31:6	RO	0x0	reserved
5:0	RW	0x00	reg_hs_tta_sure Data2 Lane The value of counter for HS Tta-sure Tta-sure for turnaround Tta-sure = Ttxclkesc*value

MIPIPHY_MIPIPHY_REGB2
Address: Operational Base + offset (0x02c8)
mipi phy register b2

Bit	Attr	Reset Value	Description
31:6	RO	0x0	reserved
5:0	RW	0x00	reg_hs_tta_wait Data2 Lane The value of counter for HS Tta-wait Tta-wait for turnaround Interval from receiving ppi turnaround request to sending esc request. Tta-wait = Ttxclkesc*value

MIPIPHY_MIPIPHY_REGC0
Address: Operational Base + offset (0x0300)
mipi phy register c0

Bit	Attr	Reset Value	Description
31:4	RO	0x0	reserved

Bit	Attr	Reset Value	Description
3:0	RW	0xb	reg_ths_settle Data3 Lane Configure the count time of the THS-SETTLE by protocol. After the count done, D-PHY will begin to receive the high speed data. (Can be configured from 4'h0 to 4'hd) 4'b0000 80-110 MHz 4'b0001 110-150 MHz 4'b0010 150-200 MHz 4'b0011 200-250 MHz 4'b0100 250-300 MHz 4'b0101 300-400 MHz 4'b0110 400-500 MHz 4'b0111 500-600 MHz 4'b1000 600-700 MHz 4'b1001 700-800 MHz 4'b1010 800-1000 MHz 4'b1011 additional adjust 4'b1100 additional adjust 4'b1101 additional adjust 4'b1110 additional adjust

MIPIPHY_MIPIPHY_REGC5
Address: Operational Base + offset (0x0314)
mipi phy register c5

Bit	Attr	Reset Value	Description
31:6	RO	0x0	reserved
5:0	RW	0x05	reg_hs_tlpx Data3 Lane The value of counter for HS Tlpx Time （>=Tlpx） = Tpin_txbyteclkhs * value

MIPIPHY_MIPIPHY_REGC6
Address: Operational Base + offset (0x0318)
mipi phy register c6

Bit	Attr	Reset Value	Description
31:7	RO	0x0	reserved
6:0	RW	0x00	reg_hs_ths_prepare Data3 Lane The value of counter for HS Ths-prepare For clock lane, Ths-prepare(38ns~95ns) For data lane, Ths-prepare(40ns+4UI~85ns+6UI) =Ttxddrclkhs*value

MIPIPHY_MIPIPHY_REGC7
Address: Operational Base + offset (0x031c)
mipi phy register c7

Bit	Attr	Reset Value	Description
31:6	RO	0x0	reserved
5:0	RW	0x00	reg_hs_the_zero Data3 Lane The value of counter for HS Ths-zero For clock lane, Ths-prepare+Ths-zero (>=300ns) For data lane, Ths-prepare+Ths-zero (>= 145 ns + 10*UI) = Tpin_txbyteclkhs*value For clock lane, S_HSTXTHSZERO[5:0] = 6'b100000 For data lane, S_HSTXTHSZERO[5:0] = 6'b001001 Frequency(1/UI) Value(Decimal) 80 -110 MHz 3 110-150 MHz 4 150-200 MHz 4 200-250 MHz 5 250-300 MHz 6 300-400 MHz 7 400-500 MHz 8 500-600 MHz 10 600-700 MHz 11 700-800 MHz 12 800-1000 MHz 15

MIPIPHY_MIPIPHY_REGC8
Address: Operational Base + offset (0x0320)
mipi phy register c8

Bit	Attr	Reset Value	Description
31:7	RO	0x0	reserved

Bit	Attr	Reset Value	Description
6:0	RW	0x00	reg_hs_ths_trail Data3 Lane The value of counter for HS Ths-trail For clock lane, Ths-trail (>=60ns) For data lane,Ths-trail (>=max(8UI, 60ns+4UI) = Tad_txddrclkhs_i* value For clock lane, S_HXTXTHSTRAIL[6:0] = 7'b0100111 For data lane, S_HXTXTHSTRAIL[6:0] = 7'b0100111 Frequency(1/UI) Value(Decimal) 80 -110 MHz 12 110-150 MHz 13 150-200 MHz 17 200-250 MHz 20 250-300 MHz 24 300-400 MHz 29 400-500 MHz 35 500-600 MHz 41 600-700 MHz 49 700-800 MHz 52 800-1000 MHz 64

MIPIPHY_MIPIPHY_REGC9
Address: Operational Base + offset (0x0324)
mipi phy register c9

Bit	Attr	Reset Value	Description
31:5	RO	0x0	reserved
4:0	RW	0x00	reg_hs_ths_exit Data3 Lane The value of counter for HS Ths-exit Ths-exit = Tpin_txbyteclkhs*value

MIPIPHY_MIPIPHY_REGCA
Address: Operational Base + offset (0x0328)
mipi phy register ca

Bit	Attr	Reset Value	Description
31:4	RO	0x0	reserved
3:0	RW	0x0	reg_hs_tclk_post Data3 Lane The value of counter for HS Tclk-post Tclk-post =Tpin_txbyteclkhs*value

MIPIPHY_MIPIPHY_REGCB
Address: Operational Base + offset (0x032c)
mipi phy register cb

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:0	RW	0x00	reserved Data3 Lane

MIPIPHY_MIPIPHY_REGCC
Address: Operational Base + offset (0x0330)
mipi phy register cc

Bit	Attr	Reset Value	Description
31:2	RO	0x0	reserved
1:0	RW	0x0	reg_hs_twakup Data3 Lane The value[9:8] of counter for HS Twakup also see REGCD

MIPIPHY_MIPIPHY_REGCD
Address: Operational Base + offset (0x0334)
mipi phy register cd

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7:0	RW	0x00	reg_hs_twakup Data3 Lane The value[7:0] of counter for HS Twakup Twakup for ulpm, Twakup = Tpin_sys_clk*value[9:0]

MIPIPHY_MIPIPHY_REGCE
Address: Operational Base + offset (0x0338)
mipi phy register ce

Bit	Attr	Reset Value	Description
31:4	RO	0x0	reserved
3:0	RW	0x0	reg_hs_tclk_pre Data3 Lane The value of counter for HS Tclk-pre Tclk-pre for clock lane Tclk-pre = Tpin_txbyteclkhs*value

MIPIPHY_MIPIPHY_REGD0
Address: Operational Base + offset (0x0340)
mipi phy register d0

Bit	Attr	Reset Value	Description
31:6	RO	0x0	reserved

Bit	Attr	Reset Value	Description
5:0	RW	0x00	reg_hs_tta_go Data3 Lane The value of counter for HS Tta-go Tta-go for turnaround Tta-go = Ttxclkesc*value

MIPIPHY_MIPIPHY_REGD1
Address: Operational Base + offset (0x0344)
mipi phy register d1

Bit	Attr	Reset Value	Description
31:6	RO	0x0	reserved
5:0	RW	0x00	reg_hs_tta_sure Data3 Lane The value of counter for HS Tta-sure Tta-sure for turnaround Tta-sure = Ttxclkesc*value

MIPIPHY_MIPIPHY_REGD2
Address: Operational Base + offset (0x0348)
mipi phy register d2

Bit	Attr	Reset Value	Description
31:6	RO	0x0	reserved
5:0	RW	0x00	reg_hs_tta_wait Data3 Lane The value of counter for HS Tta-wait Tta-wait for turnaround Interval from receiving ppi turnaround request to sending esc request. Tta-wait = Ttxclkesc*value

MIPIPHY_MIPIPHY_REGE0
Address: Operational Base + offset (0x0380)
mipi phy register e0

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7	RW	0x0	mipi_mode_en 1: enable mipi mode 0: disable mipi mode
6	RW	0x0	ttl_mode_en 1: enable ttl mode 0: disable ttl mode

Bit	Attr	Reset Value	Description
5	RW	0x0	lvds_mode_en 1: enable lvds mode 0: disable lvds mode
4:3	RO	0x0	reserved
2	RW	0x0	reg_rstn reset the LVDS PHY configuration 1: none 0: reset
1:0	RO	0x0	reserved

MIPIPHY_MIPIPHY_REGEA
Address: Operational Base + offset (0x03a8)
mipi phy register ea

Bit	Attr	Reset Value	Description
31:3	RO	0x0	reserved
2	RW	0x0	lvds_pllpd lvds_pllpd lvds pll power down 1: power down 0: power up
1	RO	0x0	reserved
0	RW	0x0	lvds_bgpd lvds bandgap power down 1: power down 0: power up

16.5 Interface Timing

This section shows a PPI timing relationship at high-speed transmission. While pin_txrequesths is low, the Lane Module ignores the value of pin_txdatahs. To begin the transmission, the protocol drives pin_txdatahs with the first byte of data and asserts pin_txrequesths. This data byte is accepted by the PHY on the first rising edge of pin_txbyteclkhs with pin_txreadyhs also asserted. At this point, the protocol logic drives the next data byte onto pin_txdatahs. After every rising clock cycle with pin_txreadyhs active, the protocol supplies a new valid data byte or ends the transmission. After the last data byte has been transferred to the Lane Module, pin_txrequesths is driven low to cause the Lane Module to stop the transmission and enter Stop state. The minimum number of bytes transmitted could be as small as one.

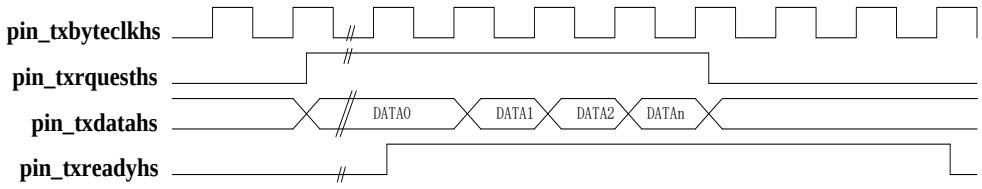


Fig. 16-3 HS-TX PPI Timing

This section shows a PPI timing relationship at low-power data transmission operation. The Protocol directs the Data Lane to enter Low-Power data transmission Escape mode by asserting pin_txrequestesc with pin_txlpdtesc high. The Low-Power transmit data is transferred on the pin_txdataesc lines when pin_txvalidesc and pin_txreadyesc are both active at a rising edge of pin_txclkesc.

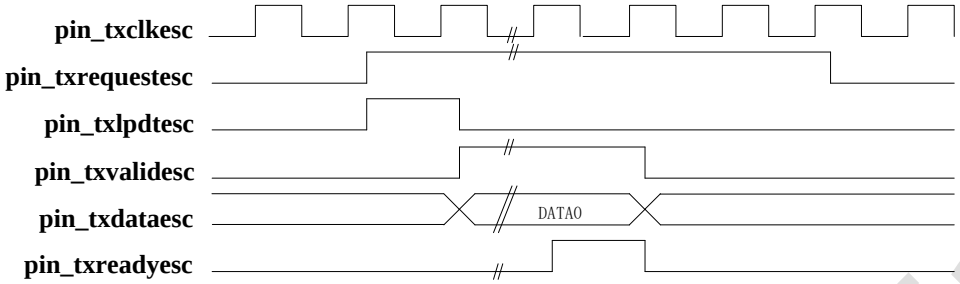


Fig. 16-4 LPDT-TX PPI Timing

This section shows a PPI timing relationship at low-power data reception. The signal pin_rxlpdtesc is asserted when the escape entry command is detected and stays high until the Lane returns to stop state, indicating that the transmission has finished.

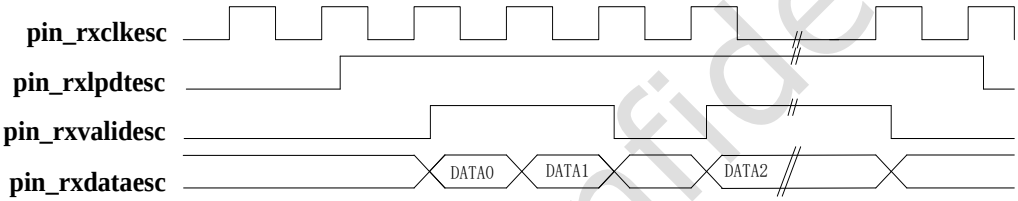


Fig. 16-5 LPDT-RX PPI Timing

16.6 Application Notes

16.6.1 Low power mode

Low Power Mode is a special feature for D-PHY. You can control this function by using proper registers from the Innosilicon D-PHY with few operations. The following is a step by step instruction for low power mode in and out.

Low Power in Steps:
Step1: Send 0x01 to register 0x00. Disable all lanes on analog part.
Step2: Send 0xe3 to register 0x01.Disable PLL and LDO.
Step3: Wait a period before reference clock have been disabled.
Step4: Disable reference clock.

Low Power out Steps:
Step1: Enable reference clock.
Step2: Wait a period after reference clock have been enabled.
Step3: Send 0xe4 to register 0x01. Enable PLL and LDO.
Step4: Send 0x7d to register 0x00. Enable all lanes on analog part.
Step5: Send 0xe0 to register 0x01. Reset analog.
Step6: Wait a period after analog has been reset.
Step7: Send 0x1e to register 0x20. Reset digital.
Step8: Send 0x1f to register 0x20. Reset digital.
Step9: Wait a period before normal transmission.

16.6.2 Programmable PLL IN DSI TX

Frequency Calculating Formula
The PLL output frequency can be calculated using a simple formula:

PLL_Output_Frequency = FREF/PREDIV*FBDIV

PLL_Output_Frequency: It is equal to DDR- Clock-Frequency * 2
FREF :PLL input reference frequency which equals to the frequency of the pin_clkhtref
PREDIV : PLL input reference clock divider which can be configured by the register of reg_prediv
FBDIV :Integer value programmed into feedback divider which can be configured by the register of reg_fbdiv

For example,
FREF =20MHz, PLL output frequency = 800Hz, so set PREDIV=1, FBDIV=40

Additional Programming Considerations
1. The divided reference frequency (FREF/PREDIV) should be less than 40MHz.
2. The all possible settings of feedback divider are 12,13,14,16~511.

16.7 ELECTRICAL SPECIFICATIONS

16.7.1 DC SPECIFICATIONS

Table 16-1 HS Transmitter DC specifications

Parameter	Description	Min	Nom	Max	Unit	Note
VCMTX	HS TX staticCommon-mode voltage	150	200	250	mV	1
|ΔVCMTX(1,0)|	VCMTX mismatch when output is Differential-1 or Differential-0			5	mV	2
|VOD|	HS transmit differential voltage	140	200	270	mV	1
|ΔVOD|	VOD mismatch when output is Differential-1 or Differential-0			10	mV	2
VOHHS	HS output high voltage			360	mV	1
ZOS	Single ended output impedance	40	50	62.5	ohm	
∆ZOS	Single ended output impedance mismatch			10	%	

1. Value when driving into load impedance anywhere in the ZID range.
2. It is recommended the implementer minimize ΔVOD and ΔVCMTX(1,0) in order to minimize radiation and optimize signal integrity.

Table 16-2 HS Transmitter DC specifications

Parameter	Description	Min	Nom	Max	Unit	Note
VIH	Logic 1 input voltage	880			mV	
VIL	Logic 0 input voltage, not in ULPState			550	mV	

VIL-ULPS	Logic 0 input voltage, ULP State			300	mV	
VHYST	Input hysteresis	25			mV	

Table 16-3 LP Transmitter DC Specifications

Parameter	Description	Min	Nom	Max	Unit	Note
VOH	The venin output high level	1.1	1.2	1.3	V	
VOL	The venin output low level	-50		50	mV	
ZOLP	Output impedance of LP transmitter	110			Ω	1

1. Though no maximum value for ZOLP is specified, the LP transmitter output impedance shall ensure theTRLP/TFLP specification is met.

16.7.2 AC specifications

Table 16-4 HS receiver AC specifications

Parameter	Description	Min	Nom	Max	Unit	Note
ΔVCMRX(HF)	Common-mode interference beyond 450 MHz			100	mV	2
ΔVCMRX(LF)	Common-mode interference 50MHz – 450MHz	-50		50	mV	1,4
CCM	Common-mode termination			60	pF	3

1. Excluding 'static' ground shift of 50mV
2. ΔVCMRX(HF)is the peak amplitude of a sine wave superimposed on the receiver inputs.
3. For higher bit rates a 14pF capacitor will be needed to meet the common-mode return loss specification.
4. Voltage difference compared to the DC average common-mode potential.

Table 16-5 LP receiver AC specifications

Parameter	Description	Min	Nom	Max	Unit	Note
eSPIKE	Input pulse rejection			300	V.ps	1，2,3
TMIN-RX	Minimum pulse width response	20			ns	4
VINT	Peak interference amplitude			200	mV	
fINT	Interference frequency	450			MHz	

1. Time-voltage integration of a spike above VIL when being in LP-0 state or below VIH when being in LP-1 state
2. An impulse less than this will not change the receiver state.
3. In addition to the required glitch rejection, implementers shall ensure rejection of known RF-interferers.

4. An input pulse greater than this shall toggle the output.

Table 16-6 LP Transmitter AC specifications

Parameter	Description		Min	Nom	Max	Unit	Note
TRLP/TFLP	15%-85% rise time and fall time				25	ns	1
TREOT	30%-85% rise time and fall time				35	ns	1,5,6
TLP-PULSE-TX	Pulse width of exclusive-OR clock the LP	First LP exclusive-OR clock pulse after Stop state or last pulse before Stop state	40			ns	4
		All other pulses	20				4
TLP-PER-TX	Period of the LP exclusive-OR clock		90			ns	
δV/δtSR	Slew rate @ CLOAD = 0pF				500	mV/ns	1,3,7,8
	Slew rate @ CLOAD = 5pF				300	mV/ns	1,3,7,8
	Slew rate @ CLOAD = 20pF				250	mV/ns	1,3,7,8
	Slew rate @ CLOAD = 70pF				150	mV/ns	1,3,7,8
	Slew rate @ CLOAD = 0 to 70pF(Falling Edge Only)		30			mV/ns	1,2,3
	Slew rate @ CLOAD = 0 to 70pF(Rising Edge Only)		30			mV/ns	1,3,9
	Slew rate @ CLOAD = 0 to 70pF(Rising Edge Only)		30-0.075 * (VO,INST – 700)			mV/ns	1,10,11
CLOAD	Load capacitance		0		70	pF	1

1. CLOAD includes the low-frequency equivalent transmission line capacitance. The capacitance of TX and RX are assumed to always be <10pF. The distributed line capacitance can be up to 50pF for a transmission line with 2ns delay.
2. When the output voltage is between 400 mV and 930 mV.
3. Measured as average across any 50 mV segment of the output signal transition.
4. This parameter value can be lower than TLPX due to differences in rise vs. fall signal slopes and trip levels and mismatches between Dp and Dn LP transmitters. Any LP exclusive-OR pulse observed during HS EoT (transition from HS level to LP-11) is glitch behavior as described in section 8.2.2.
5. The rise-time of TREOT starts from the HS common-level at the moment the differential amplitude drops below 70mV, due to stopping the differential drive.
6. With an additional load capacitance CCM between 0 and 60pF on the termination center

tap at RX side of the Lane
7. This value represents a corner point in a piecewise linear curve.
8. When the output voltage is in the range specified by VPIN (absmax).
9. When the output voltage is between 400 mV and 700 mV.
10. Where VO, INST is the instantaneous output voltage, VDP or VDN, in millivolts.
11. When the output voltage is between 700 mV and 930 mV.

Chapter 17 EBC

17.1 Overview

EBC is the TCON module for Electronic Paper Display (EPD).
System interface:

- AHB slave for register configuration
- AHB master for frame data transfer (DMA)
- Interrupt output

EPD interface:

- E-ink EPD compatible
- Up to 2048x2048 resolution
- Up to 16 level gray scale
- Up to 128 frames every scanning
- LUT updateable (8KB)
- Direct mode and LUT mode
- All-update mode and Diff-update mode
- Single-phase and multi-phase mode
- Support window display
- Source driver interface
- Gate driver interface

17.2 Block Diagram

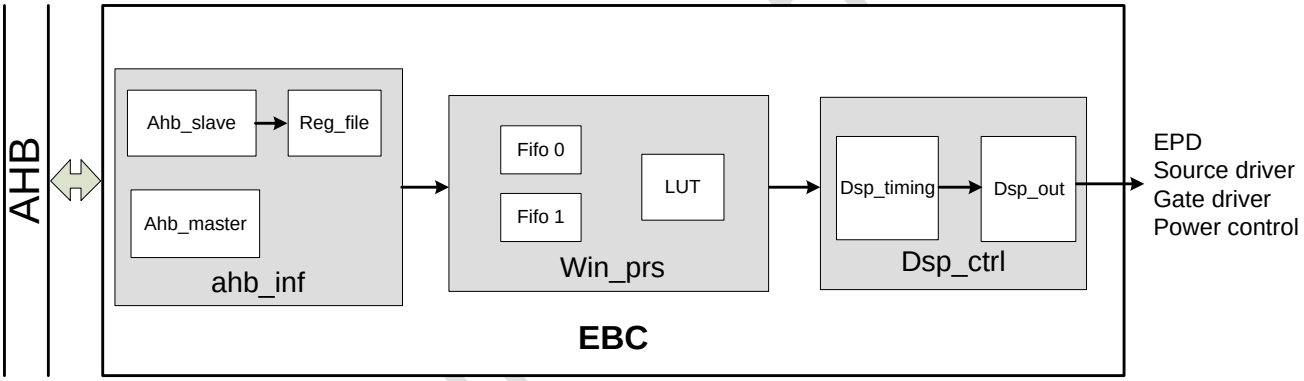


Fig. 17-1 EBC Block Diagram

17.3 Function Description

17.3.1 Data Format

Table 17-1 EBC Input Data Format

DSP_LUT MODE	WIN_FMT [1:0]	Data format	Bit Map
0	xx	S-data(2bpp)	{S3[7:0], S2[7:0], S1[7:0], S0[7:0]}
1	00	Y-data(4bpp)	{Y7[3:0], Y6[3:0], Y5[3:0], Y4[3:0], Y3[3:0], Y2[3:0], Y1[3:0], Y0[3:0]}
1	01	Y-data(8bpp)	{Y3[7:0], Y2[7:0], Y1[7:0], Y0[7:0]}
1	10	RGB888	{8'bx, R[7:0], G[7:0], B[7:0]}
1	11	RGB565	{ R1[4:0], G1[5:0], B1[4:0], R0[4:0], G0[5:0], B0[4:0]}

17.3.2 Waveform generation Format

1. Direct mode

In direct scanning mode, the source data is just read by internal DMA and sent to the display output directly.

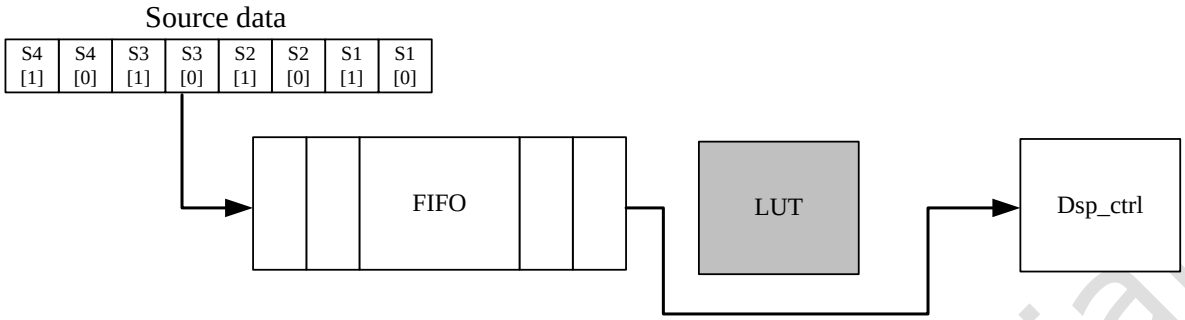


Fig. 17-2 EBC Block Diagram

2. LUT mode

In LUT scanning mode, internal DMA read the original pixel data into the FIFO, then the pixel data is sent to the look-up table to be translated the EPD source data.

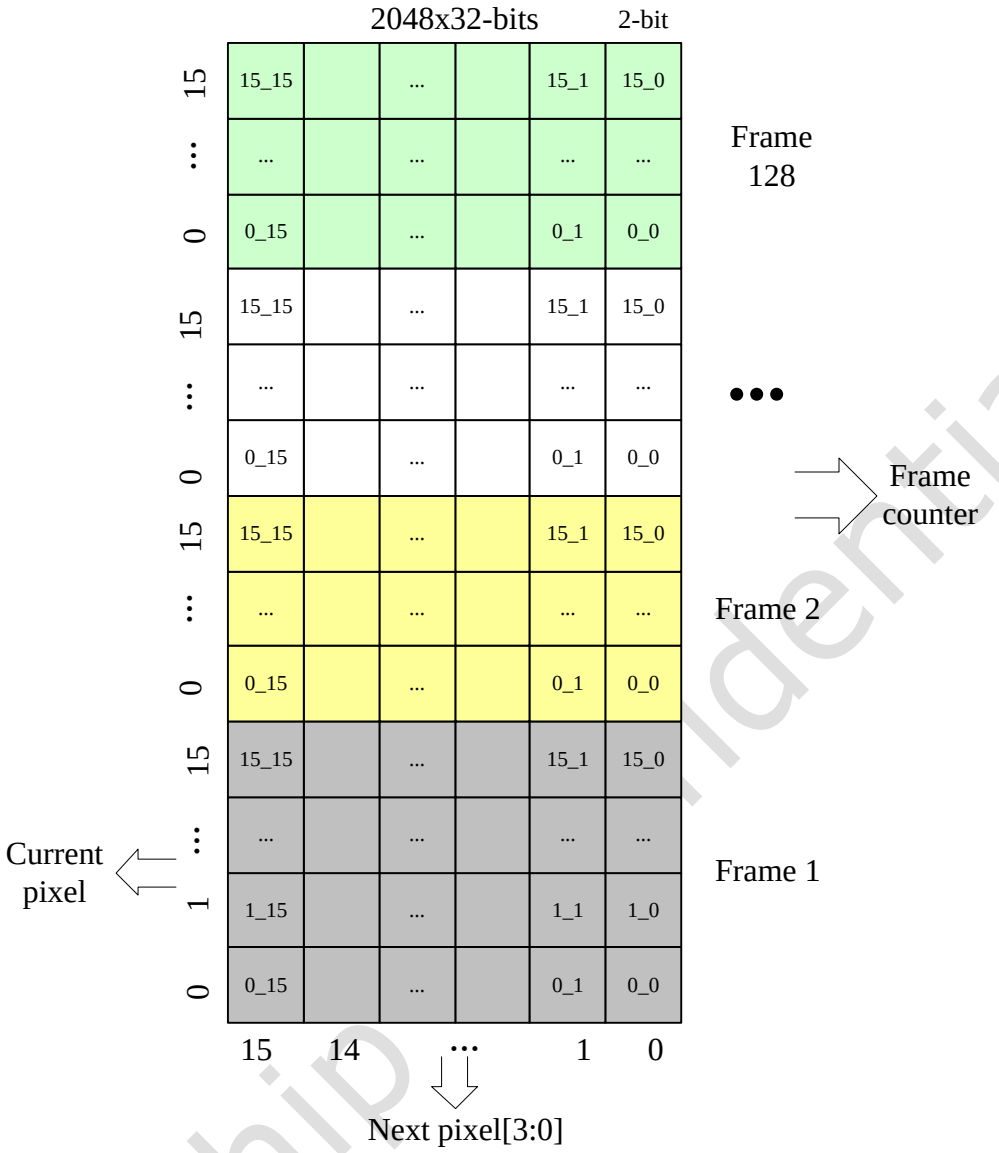


Fig. 17-3 EBC Lut structure

17.3.3 Window display mode

Window display is supported in EBC, dsp_win_width/dsp_win_height and dsp_win_st should be set to define the display window. The source value of other area is background value, which is configurable.

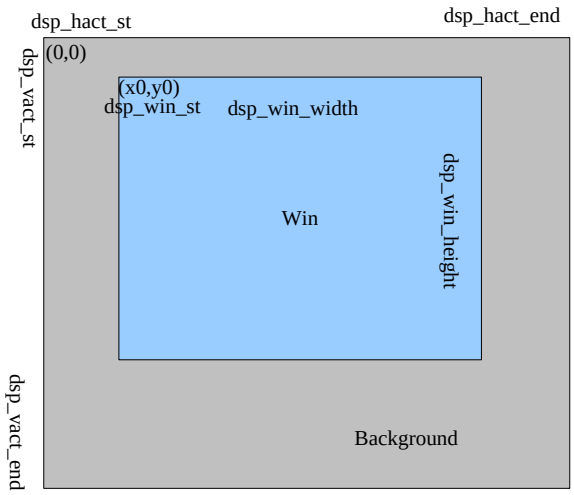


Fig. 17-4 EBC window display

17.3.4 Frame control

1. Single frame mode(always used in direct mode)

In single frame mode (direct mode), every display frame is controlled by the "frame_start" command. The next "frame_start" command should be after the end of the last frame.

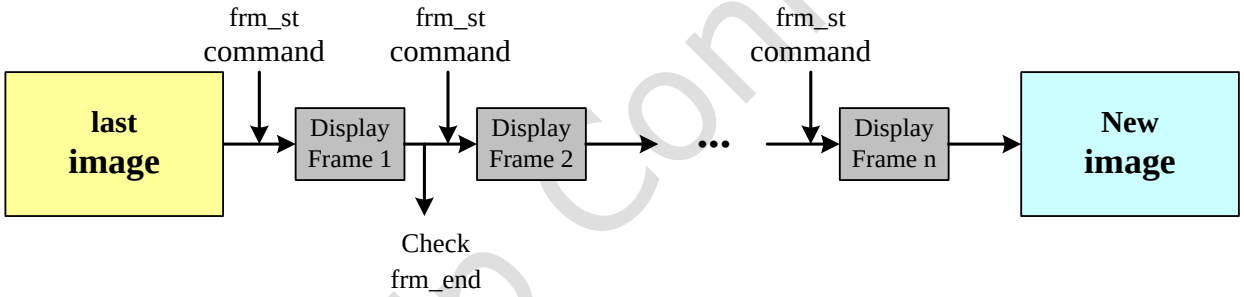


Fig. 17-5 EBC single frame display

2. Multi-frame mode

In multi-frame mode, the number of the display frames should be set before the "frame_start" command. Then the frame scanning is done automatically until the end of the last frame.

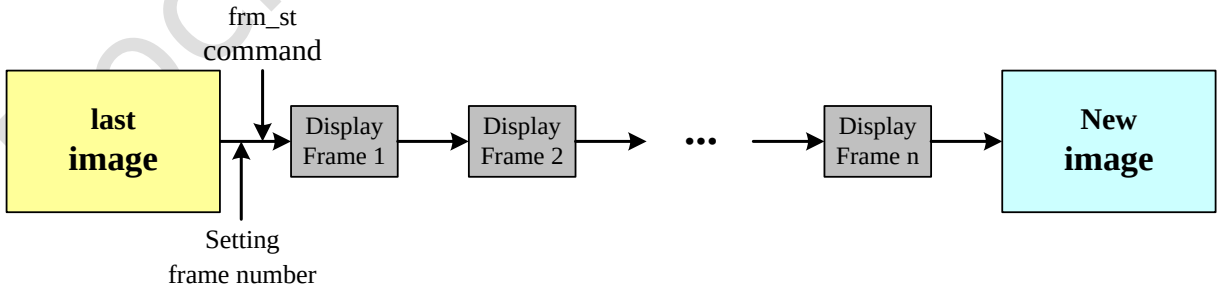


Fig. 17-6 EBC multi-frame display

17.3.5 Image translation phase

1. Single phase mode

If the number of image translationframes is less than 64. Single phase is necessary for

image translation

In this mode, the "vcom_en" is just turn on during the display period.

2. Multi-phase mode

If the number of image translationframes is larger than 64, multi-phase should be done for image translation because the max display frame is 64.

In this mode, the "vcom_en" should not be turn off between the idle period of different display phase. To avoid this, you can set VCOM_MODE = "1" before the first start display start command. To turn off the VCOM_EN, "0" should be write in before the last start display start command.

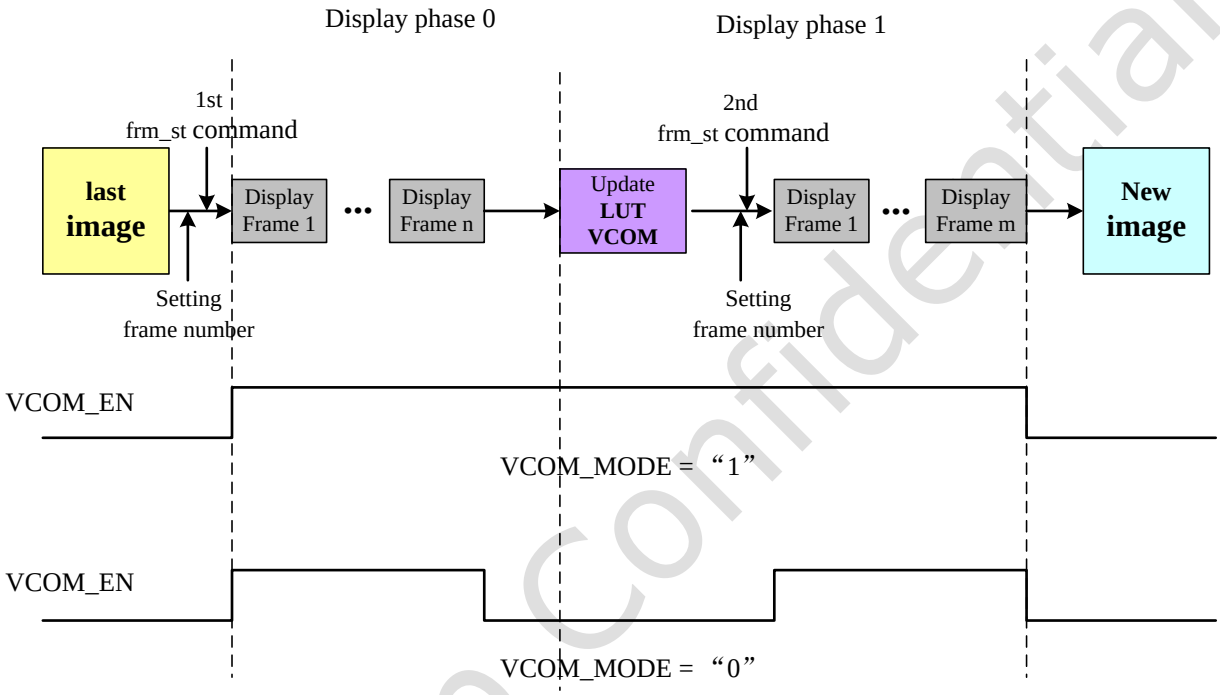


Fig. 17-7 EBC display phase

17.3.6 IO description

Table 17-2 EBC output pins

EINK	Description
SCLK	Source clock
SLE	Source latch data enable
SOE	Source data enable/start pulse
SCE[5:0]	
SDO[7:0]	Source data
SDIR	X scanning direction
GCLK	Gate clock
GSP	Gate start pulse
GOE	Gate output enable
GDIR	Y scanning direction
BORDER[1:0]	COM voltage
VCOM_CTRL	VCOM enable
PWR[2:0]	POWER control

1. IO MUX

EBC share IOs with LCDC and LVDS，following is the IO map.

For EBC output, first switch LVCD's TTL outputs source to EBC outputs by setting GRF_LVDS_CON0[0] = 1'b1. Then switch LCDC's output IOs to EBC outputs by setting GRF_GPIO2B[15:0] = 16'haaaa, GRF_GPIO2C[15:0] = 16'haaaa and GRF_GPIO2D[3:0] = 4'ha.

Table 17-3 EBC IOMUX

IOMUX			
	lcdc0_dclk	ebc_sdclk	
	lcdc0_hsync	ebc_sdle	
	lcdc0_vsync	ebc_sdoe	
	lcdc0_den	ebc_gdclk	
lvds_p0	lcdc0_d0	ebc_sddo0	
lvds_n0	lcdc0_d1	ebc_sddo1	
lvds_p1	lcdc0_d2	ebc_sddo2	
lvds_n1	lcdc0_d3	ebc_sddo3	
lvds_p2	lcdc0_d4	ebc_sddo4	
lvds_n2	lcdc0_d5	ebc_sddo5	
lvds_p3	lcdc0_d6	ebc_sddo6	
lvds_n3	lcdc0_d7	ebc_sddo7	
lvds_clkp	lcdc0_d8	ebc_sdce0	
lvds_clkn	lcdc0_d9	ebc_sdce1	
	lcdc0_d10	ebc_sdce2	
	lcdc0_d11	ebc_sdce3	
	lcdc0_d12	ebc_sdce4	
	lcdc0_d13	ebc_sdce5	
	lcdc0_d14	ebc_vcom	
	lcdc0_d15	ebc_gdoe	
	lcdc0_d16	ebc_gdsp	
	lcdc0_d17	ebc_gdpwr0	
	lcdc0_d18	ebc_gdrl	i2c2_sda
	lcdc0_d19	ebc_sdshr	i2c2_scl
	lcdc0_d20	ebc_border0	urt2_sin
	lcdc0_d21	ebc_border1	urt2_sout
	lcdc0_d22	ebc_gdpwr1	
	lcdc0_d23	ebc_gdpwr2	

2. Source driver interface

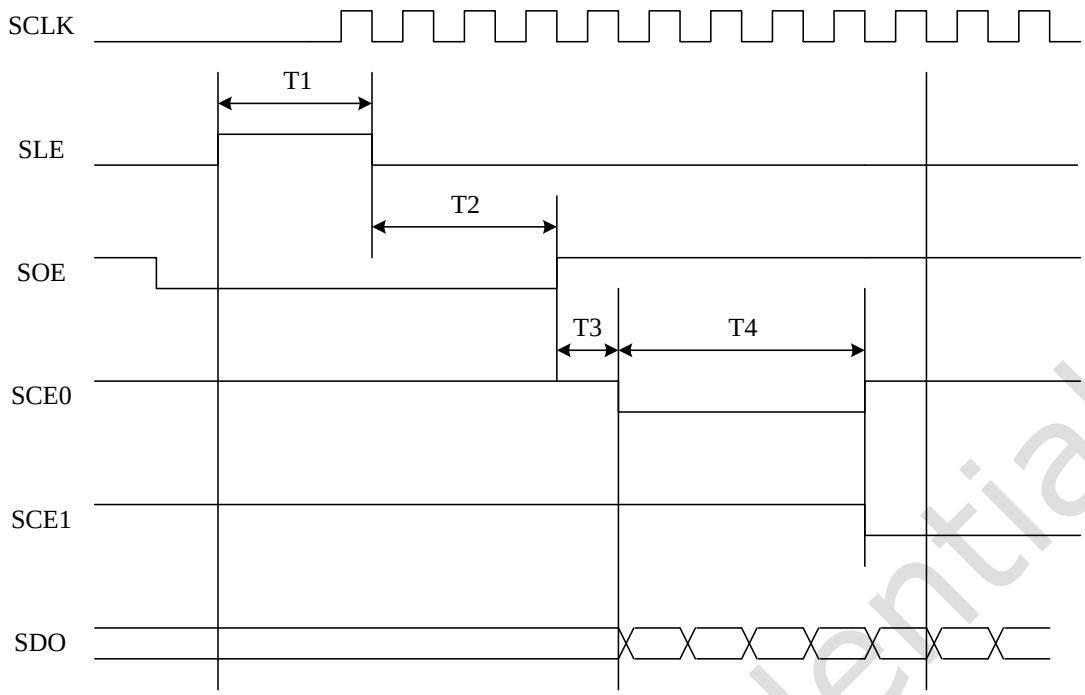


Fig. 17-8 EBC source driver timing 1

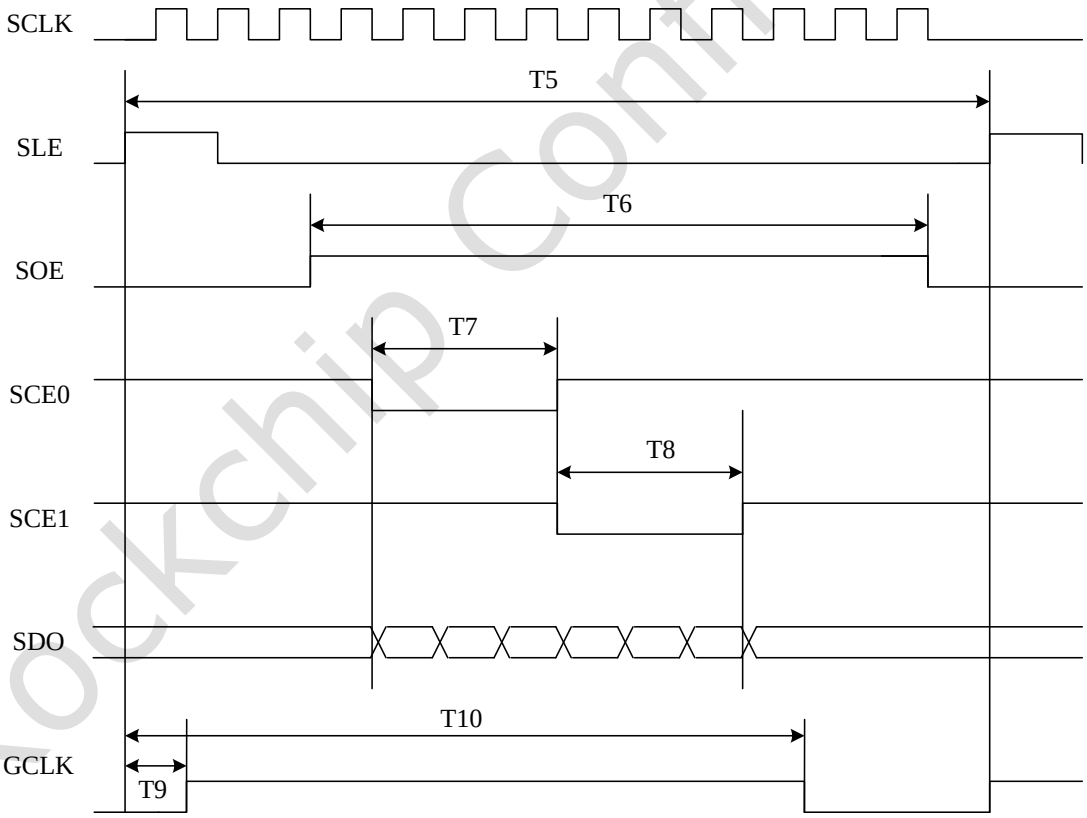


Fig. 17-9 EBC source driver timing 2

Table 17-4 EBC source driver setting

Timing	Value （6-inch 800x600）	Register setting
DCLK	33.25MHz	

Timing	Value （6-inch 800x600）	Register setting
SCLK	4 DCLK(8.3125 MHz)	(DSP_CLK_DIV+1)
T1	10 SCLK	DSP_HS_END
T2	3 SCLK	(DSP_HACT_ST – DSP_HS_END)
T3	1 SCLK	Fixed
T1+T2+T3	14 SCLK	DSP_WIN_XST
T4	100 SCLK	DSP_SCE_WIDTH
T5	315 SCLK	DSP_HTOTAL
T6	300 SCLK	(DSP_HACT_END – DSP_HACT_ST)
T7	100 SCLK	DSP_SCE_WIDTH
T8	100 SCLK	DSP_SCE_WIDTH
T9	0 SCLK	DSP_GCLK_ST
T10	215 SCLK	DSP_GCLK_END

3. Gate driver interface

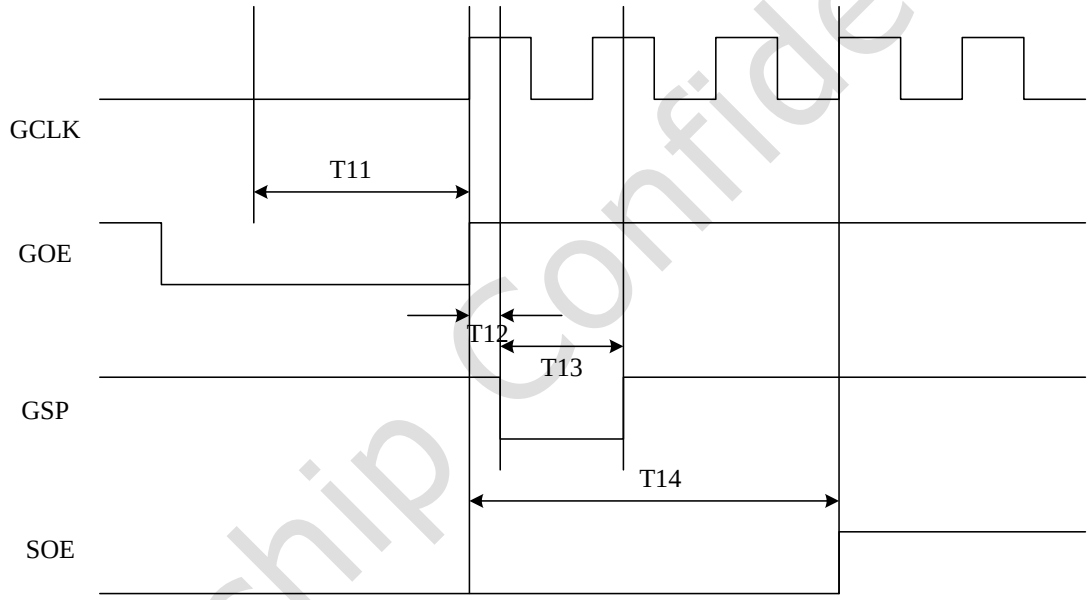


Fig. 17-10 EBC gate driver timing 1

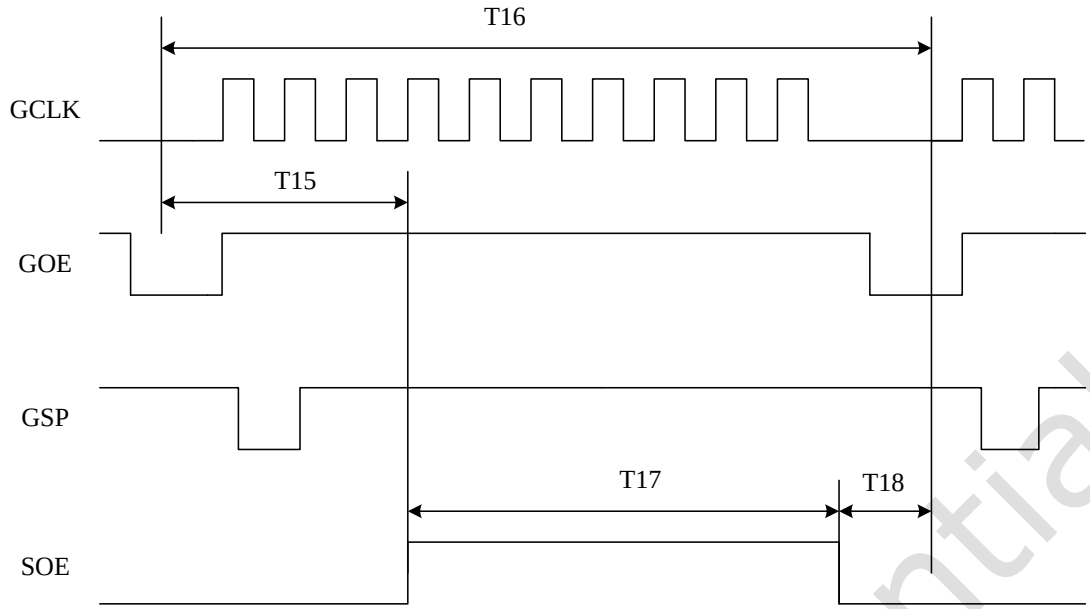


Fig. 17-11 EBC gate driver timing 2

Table 17-5 EBC gate driver setting

Timing	Value（6-inch 800x600）	Register setting
GCLK	315 SCLK	DSP_HTOTAL
T11	4 GCLK	DSP_VS_END
T12	107 SCLK	(DSP_GD_END/2)
T13	1 GCLK	Fixed
T14	4 GCLK	（DSP_VACT_ST - DSP_VS_END）
T15	8 GCLK	DSP_VACT_ST
T16	619 GCLK	DSP_VTOTAL
T17	601 GCLK	（DSP_VACT_END - DSP_VACT_ST）
T18	11 GCLK	（DSP_VTOTAL - DSP_VACT_END）

17.4 Register Description

17.4.1 Register summary

Name	Offset	Size	Reset Value	Description
EBC_DSP_ST	0x0000	W	0x00000000	frame start register
EBC_EPD_CTRL	0x0004	W	0x00006400	EPD control register
EBC_DSP_CTRL	0x0008	W	0x00030000	Display control register
EBC_DSP_HTIMING0	0x000c	W	0x013b000a	Display horizontal timing setting0
EBC_DSP_HTIMING1	0x0010	W	0x0139000d	Display horizontal timing setting1
EBC_DSP_VTIMING0	0x0014	W	0x026b0004	Display vertical timing setting0
EBC_DSP_VTIMING1	0x0018	W	0x02610008	Display vertical timing setting0
EBC_DSP_ACT_INFO	0x001c	W	0x025800c8	Display active width/height

Name	Offset	Size	Reset Value	Description
EBC_WIN_CTRL	0x0020	W	0x00000000	Window control register
EBC_WIN_MST0	0x0024	W	0x00000000	Old window layer memory start address
EBC_WIN_MST1	0x0028	W	0x00000000	New window layer memory start address
EBC_WIN_VIR	0x002c	W	0x00000320	Window layer virtual width
EBC_WIN_ACT	0x0030	W	0x02580320	Window active width/height
EBC_WIN_DSP	0x0034	W	0x02580320	Window display width/height
EBC_WIN_DSP_ST	0x0038	W	0x00000000	Window display start position
EBC_INT_CTRL	0x003c	W	0x00000038	Interrupt control register
EBC_VCOM0	0x0040	W	0x00000000	VCOM0
EBC_VCOM1	0x0044	W	0x00000000	VCOM1
EBC_VCOM2	0x0048	W	0x00000000	VCOM2
EBC_VCOM3	0x004c	W	0x00000000	VCOM3
EBC_CONFIG_DONE	0x0050	W	0x00000000	CONFIG finish register
EBC_LUT_ADDR_MAP	0x1000	W	0x00000000	LUT address map

17.4.2 Detail Register Description

EBC_DSP_ST

Address: Operational Base + offset (0x0000)

frame start register

Bit	Attr	Reset Value	Description
31:9	RO	0x0	reserved
8:2	RW	0x00	frm_total_num Frame total number(n) Frame_num = n+1. Up to 128
1	RO	0x0	reserved
0	RW	0x0	frm_st Frame start bit Writing'1' to trigger one frame display. Read this bit for Hold status.

EBC_EPD_CTRL

Address: Operational Base + offset (0x0004)

EPD control register

Bit	Attr	Reset Value	Description
31:27	RW	0x00	dsp_gd_st DSP_GD_ST GCLK rising edge point(SCLK), which count from the falling edge of hsync
26:16	RW	0x0d7	dsp_gd_end DSP_GD_END GCLK falling edge point(SCLK), which count from the falling edge of hsync
15:8	RW	0x64	dsp_sce_width DSP_SCE_WIDTH Source driver length
7:5	RO	0x0	reserved
4:2	RW	0x0	pwr_en Power enable[3:0]
1	RW	0x0	gate_scan_dir Gate scanning direction 1: button to top 0: top to button
0	RW	0x0	source_scan_dir Source scanning direction 1: right to left 0: left to right

EBC_DSP_CTRL

Address: Operational Base + offset (0x0008)

Display control egister

Bit	Attr	Reset Value	Description
31:30	RW	0x0	dsp_swap Display swap 11: SDO[7:0] = P0,P1,P2,P3 10: SDO[7:0] = P2,P3,P0,P1 01: SDO[7:0] = P1,P0,P3,P2 00: SDO[7:0] = P3,P2,P1,P0
29	RW	0x0	diff_update_mode_en Diff update Mode enable 1: Diff update Mode 0: Normal update Mode
28	RW	0x0	dsp_mode Display Mode 1: LUT mode 0: Direct mode

Bit	Attr	Reset Value	Description
27	RW	0x0	vcom_mode VCOM mode 1: Multi phase scanning mode 0: one phase scanning mode
26	RW	0x0	goe_pol_inv DSP OUTPUT GOE polarity invert 1: invert; 0: default;
25	RW	0x0	gsp_pol_inv DSP OUTPUT GSP polarity invert 1: invert; 0: default;
24	RW	0x0	gclk_pol_inv DSP OUTPUT GCLK polarity invert 1: invert; 0: default;
23	RW	0x0	sce_pol_inv DSP OUTPUT SCE polarity invert 1: invert; 0: default;
22	RW	0x0	soe_pol_inv DSP OUTPUT SOE polarity invert 1: invert; 0: default;
21	RW	0x0	sle_pol_inv DSP OUTPUT SLE polarity invert 1: invert; 0: default;
20	RW	0x0	sclk_pol_inv DSP OUTPUT SCLK polarity invert 1: invert; 0: default;
19:16	RW	0x3	sclk_div SCLK divide rate SCLK =（DSP_CLK_DIV+1）* DCLK
15:0	RW	0x0000	sdo_default_val SDO default value

EBC_DSP_HTIMING0

Address: Operational Base + offset (0x000c)

Display horizontial timing setting0

Bit	Attr	Reset Value	Description
31:27	RO	0x0	reserved
26:16	RW	0x13b	dsp_h_period Panel display scanning horizontal period.
15:8	RO	0x0	reserved
7:0	RW	0x0a	dsp_hsync_width Panel display scanning hsync(SLE) pulse width.

EBC_DSP_HTIMING1

Address: Operational Base + offset (0x0010)

Display horizontial timing setting1

Bit	Attr	Reset Value	Description
31:27	RO	0x0	reserved
26:16	RW	0x139	dsp_hact_end Panel display scanning horizontal active end point
15:8	RO	0x0	reserved
7:0	RW	0x0d	dsp_hact_st Panel display scanning horizontal active start point

EBC_DSP_VTIMING0

Address: Operational Base + offset (0x0014)

Display vertical timing setting0

Bit	Attr	Reset Value	Description
31:27	RO	0x0	reserved
26:16	RW	0x26b	dsp_v_period Panel display scanning vertical period.
15:8	RO	0x0	reserved
7:0	RW	0x04	dsp_vsync_width Panel display scanning vsync(GOE) pulse width.

EBC_DSP_VTIMING1

Address: Operational Base + offset (0x0018)

Display vertical timing setting0

Bit	Attr	Reset Value	Description
31:27	RO	0x0	reserved
26:16	RW	0x261	dsp_vact_end Panel display scanning vertical active end point
15:8	RO	0x0	reserved
7:0	RW	0x08	dsp_vact_st Panel display scanning vertical active start point

EBC_DSP_ACT_INFO

Address: Operational Base + offset (0x001c)

Display active width/height

Bit	Attr	Reset Value	Description
31:27	RO	0x0	reserved
26:16	RW	0x258	dsp_height Display height
15:11	RO	0x0	reserved
10:0	RW	0x0c8	dsp_width Display width

EBC_WIN_CTRL

Address: Operational Base + offset (0x0020)

Window control register

Bit	Attr	Reset Value	Description
31:18	RO	0x0	reserved
17	RW	0x0	win_en Win enable
16:12	RW	0x00	ahb_incr_num Ahb master Incrnum
11:9	RW	0x7	ahb_trans_type Ahb master burst type
8:2	RW	0x70	win_fifo_almost_full_level Win fifo almost full level
1	RO	0x0	reserved

Bit	Attr	Reset Value	Description
0	RW	0x0	win_fmt win_fmt [1:0] win source data format: 11: RGB888(24bpp) 10: RGB565(16bpp) 01: Y-data(8bpp) 00: Y-data(4bpp)

EBC_WIN_MST0

Address: Operational Base + offset (0x0024)

Old window layer memory start address

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	win_mst0 Start address of current image data in memory

EBC_WIN_MST1

Address: Operational Base + offset (0x0028)

New window layer memory start address

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	win_mst1 Start address of next image data in memory

EBC_WIN_VIR

Address: Operational Base + offset (0x002c)

Window layer virtual width

Bit	Attr	Reset Value	Description
31:16	RO	0x0	reserved
15:0	RW	0x0320	win_vir_width Win virtual width

EBC_WIN_ACT

Address: Operational Base + offset (0x0030)

Window active width/height

Bit	Attr	Reset Value	Description
31:27	RO	0x0	reserved
26:16	RW	0x258	win_act_height Win active height
15:11	RO	0x0	reserved
10:0	RW	0x320	win_act_width Win active width

EBC_WIN_DSP

Address: Operational Base + offset (0x0034)

Window display width/height

Bit	Attr	Reset Value	Description
31:27	RO	0x0	reserved
26:16	RW	0x258	win_dsp_height Win display height
15:11	RO	0x0	reserved
10:0	RW	0x320	win_dsp_width Win display width

EBC_WIN_DSP_ST

Address: Operational Base + offset (0x0038)

Window display start position

Bit	Attr	Reset Value	Description
31:27	RO	0x0	reserved
26:16	RW	0x008	win_dsp_y_st Win display y start point
15:11	RO	0x0	reserved
10:0	RW	0x00d	win_dsp_x_st Win display x start point

EBC_INT_CTRL

Address: Operational Base + offset (0x003c)

Interrupt control register

Bit	Attr	Reset Value	Description
31:16	RO	0x0	reserved
15:9	RW	0x00	frm_flag_num Frame number of the scanning flag interrupt The display frame number when the flag interrupt occur, the range is (0~ DSP_FRM_TOTAL-1).
8	W1C	0x0	frm_flag_int_clr Frame flag Interrupt clear After be set to 1, this bit will clear by itself 1 cycle later.
7	W1C	0x0	dsp_end_int_clr Display end interrupt clear After be set to 1, this bit will clear by itself 1 cycle later.
6	W1C	0x0	frm_end_int_clr Frame end interrupt clear After be set to 1, this bit will clear by itself 1 cycle later.
5	RW	0x1	frm_flag_int_msk Frame flag Interrupt Mask 0: unmask; 1: mask;
4	RW	0x1	dsp_end_int_msk Display end interrupt mask 0: unmask; 1: mask;
3	RW	0x1	frm_end_int_msk Frame end interrupt mask 0: unmask; 1: mask;
2	RO	0x0	frm_flag_int_sta Frame flag Interrupt status
1	RO	0x0	dsp_end_int_sta Display end interrupt status
0	RO	0x0	frm_end_int_sta Frame end interrupt status

EBC_VCOM0

Address: Operational Base + offset (0x0040)

VCOM0

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	vcom0 VCOM [1:0] values in 0-16 frame. The VCOM [1:0] indicate the COM voltage in one frame. The VCOM can be different in different frame. So we can set the VCOM [1:0] sequence using display frames before start the display.

EBC_VCOM1

Address: Operational Base + offset (0x0044)

VCOM1

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	vcom1 VCOM [1:0] values in 16-32 frame. The VCOM [1:0] indicate the COM voltage in one frame. The VCOM can be different in different frame. So we can set the VCOM [1:0] sequence using display frames before start the display.

EBC_VCOM2

Address: Operational Base + offset (0x0048)

VCOM2

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	vcom2 VCOM [1:0] values in 32-48 frame. The VCOM [1:0] indicate the COM voltage in one frame. The VCOM can be different in different frame. So we can set the VCOM [1:0] sequence using display frames before start the display.

EBC_VCOM3

Address: Operational Base + offset (0x004c)

VCOM3

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	vcom3 VCOM [1:0] values in 48-64 frame. The VCOM [1:0] indicate the COM voltage in one frame. The VCOM can be different in different frame. So we can set the VCOM [1:0] sequence using display frames before start the display.

EBC_CONFIG_DONE

Address: Operational Base + offset (0x0050)

CONFIG finish register

Bit	Attr	Reset Value	Description
31:1	RO	0x0	reserved
0	W1C	0x0	config_done config finish In the first setting of the register, the new value was saved into the mirror register. When all the register config finish, writing this register to enable the copyright of the mirror register to real register. Then register would be updated at the start of every frame.

EBC_LUT_ADDR_MAP

Address: Operational Base + offset (0x1000)

LUT address map

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	lut_addr_map LUT memory address map (8k Byte)

Chapter 18 DSI Controller

18.1 Overview

The Display Serial Interface (DSI) is part of a group of communication protocols defined by the MIPI Alliance. The MIPI Controller is a digital core that implements all protocol functions defined in the MIPI DSI Specification. The MIPI Controller provides an interface between the system and the MIPI D-PHY, allowing the communication with a DSI-compliant display. The MIPI Controller supports one to four lanes for data transmission with MIPI D-PHY.
The MIPI Controller supports the following features:

- Compliant with MIPI Alliance standards
- Support the DPI interface color coding mappings into 24-bit Interface
  - 16 bits per pixel, configurations 1,2,and 3
  - 18 bits per pixel, configurations 1 and 2
  - 24 bits per pixel
- Programmable polarity of all DPI interface signals
- Extended resolutions beyond the DPI standard maximum resolution of 800x480 pixels:
  - Up to 2047 vertical active lines
  - Up to 63 vertical back porch lines
  - Up to 63 vertical front porch lines
  - Maximum resolution is limited by available DSI Physical link bandwidth which depends on the number of lanes and maximum speed per lane
- All commands defined in MIPI Alliance Specification for Display Command Set (DCS)
- Interface with MIPI D-PHY following PHY Protocol Interface (PPI), as defined in MIPI Alliance Specification for D-PHY
- Up to four D-PHY Data Lanes
- Bidirectional communication and escape mode support through data lane 0
- Transmission of all generic commands
- ECC and Checksum capabilities
- End of Transmission Packet (EOTp)
- Ultra Low-Power mode
- Fault recovery schemes

18.2 Block Diagram

The following diagram shows the MIPI Controller architecture.

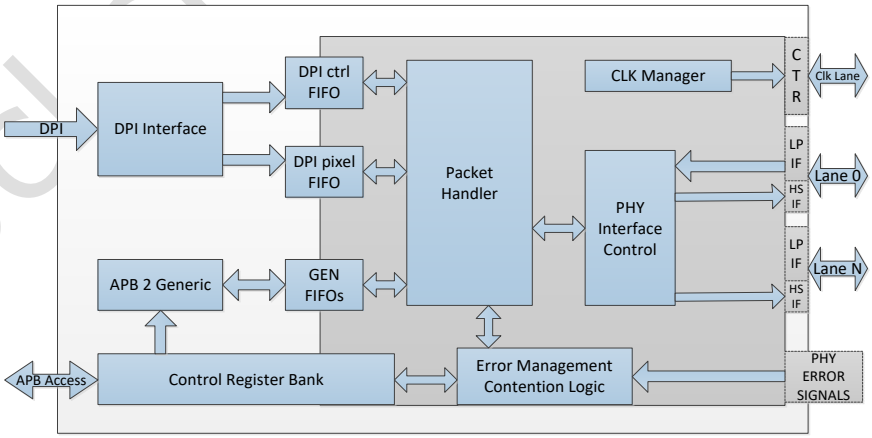


Fig. 18-1 MIPI Controller architecture

The DPI interface captures the data and control signals and conveys them to a FIFO for video control signals and another one for pixel data. This data is then used to build Video packets, hen in Video mode.
The Register Bank is accessible through a standard AMBA-APB slave interface, providing access to the MIPI Controller registers for configuration and control. There is also a fully programmable interrupt generator to inform the system about certain events.
The PHY Interface Control is responsible for managing the D-PHY PPI interface. It

acknowledges the current operation and enables low-power transmission/reception or a high-speed transmission. It also performs data splitting between available D-PHY lanes for high-speed transmission.
The Packet Handler schedules the activities inside the link. It performs several functions based on the interfaces that are currently DPI and the video transmission mode that is used (burst mode or non-burst mode with sync pulse or sync events). It builds long or short packet generating correspondent ECC and CRC codes. This block also performs the following functions: Packet reception, Validation of packet header by checking the ECC, Header correction and notification for single-bit errors, Termination of reception, Multiple header error notification.
The APB-to-Generic block bridges the APB operations into FIFOs holding the Generic commands. The block interfaces with the following FIFOS: Command FIFO, Write payload FIFO, Read payload FIFO.
The Error Management notifies and monitors the error conditions on the DSI link. It controls the timers used to determine if a timeout condition occurred, performing an internal soft reset and triggering an interruption notification.

18.3 Function Description

18.3.1 DPI interface function

TheDPI interface follows the MIPI DPI specification with pixel data bus width up to 24 bits. It is used to transmit the information in Video mode in which the transfers from the host processor to the peripheral take the form of a real-time pixel stream. This interface allows sending ShutDown (SD) and ColorMode (CM) commands, which are triggered directly by writing to the register of CFG_MISC_CON[2:1]. To transfer additional commands(for example, to initialize the display), use another interface such as APB Slave Generic Interface to complement the DPI interface.
The DPI interface captures the data and control signals and conveys them to the FIFO interfaces that transmit them to the DSI link. Two different streams of data are presented at the interface; video control signals and pixel data. Depending on the interface color coding, the pixel data is disposed differently throughout the dpipixdata bus. The following table shows the Interface pixel color coding.

Table 18-1 Color table

Signal Line	16-bit			18-bit		24-bit
	Config1	Config2	Config3	Config1	Config2	
dpipixdata23	Not used	Not used	Not used	Not used	Not used	R7
dpipixdata22	Not used	Not used	Not used	Not used	Not used	R6
dpipixdata21	Not used	Not used	R4	Not used	R5	R5
dpipixdata20	Not used	R4	R3	Not used	R4	R4
dpipixdata19	Not used	R3	R2	Not used	R3	R3
dpipixdata18	Not used	R2	R1	Not used	R2	R2
dpipixdata17	Not used	R1	R0	R5	R1	R1
dpipixdata16	Not used	R0	Not used	R4	R0	R0
dpipixdata15	R4	Not used	Not used	R3	Not used	G7
dpipixdata14	R3	Not used	Not used	R2	Not used	G6
dpipixdata13	R2	G5	G5	R1	G5	G5
dpipixdata12	R1	G4	G4	R0	G4	G4

dpipixdata11	R0	G3	G3	G5	G3	G3
dpipixdata10	G5	G2	G2	G4	G2	G2
dpipixdata9	G4	G1	G1	G3	G1	G1
dpipixdata8	G3	G0	G0	G2	G0	G0
dpipixdata7	G2	Not used	Not used	G1	Not used	B7
dpipixdata6	G1	Not used	Not used	G0	Not used	B6
dpipixdata5	G0	Not used	B4	B5	B5	B5
dpipixdata4	B4	B4	B3	B4	B4	B4
dpipixdata3	B3	B3	B2	B3	B3	B3
dpipixdata2	B2	B2	B1	B2	B2	B2
dpipixdata1	B1	B1	B0	B1	B1	B1
dpipixdata0	B0	B0	Not used	B0	B0	B0

The DPI interface can be configured to increase flexibility and promote correct usage of this interface for several systems. These configuration options are as follows: Polarity control: All the control signals are programmable to change the polarity depending on system requirements.

After the MIPI Controller reset, DPI waits for the first VSYNC active transition to start signal sampling, including pixel data, and preventing image transmission in the middle of a frame.

If interface pixel color coding is 18 bits and the 18-bit loosely packed stream is disabled, the number of lines programmed in the pixels per lines configuration is a multiple of four. This means that in this mode, the two LSBs in the configuration are always inferred as zero. The specification states that in this mode, the pixel line size should be a multiple of four.

18.3.2 APB Slave Generic Interface

The APB Slave interface allows the transmission of generic information in Command mode, and follows the proprietary register interface. Commands sent through this interface are not constrained to comply with the DCS specification, and can include generic commands described in the DSI specification as manufacturer-specific.

The MIPI Controller supports the transmission or write and read command mode packets as described in the DSI specification. These packets are built using the APB register access. The GEN_PLD_DATA register has two distinct functions based on the operation. Writing to this register sends the data as payload when sending a Command mode packet. Reading this register returns the payload of a read back operation. The GEN_HDR register contains the Command mode packet header type and header data. Writing to this register triggers the transmission of the packet implying that for a long Command mode packet, the packet's payload needs to be written in advance in the GEN_PLD_DATA register.

The valid packets available to be transmitted through the Generic interface are as follows:

Generic Write Short Packet 0 Parameters
Generic Write Short Packet 1 Parameters
Generic Write Short Packet 2 Parameter
Generic Write Short Packet 0 Parameter
Generic Write Short Packet 1 Parameters
Generic Write Short Packet 2 Parameter
Maximum Read Packet Configuration
Generic Long Write Packet
DCS Write Short Packet 0 Parameter
DCS Write Short Packet 1 Parameter
DCS Write Short Packet 0 Parameter
DCS Write Long Packet

A set of bits in the CMD_PKT_STATUS register report the status of the FIFOs associated

with APB interface support.
Generic interface packets are always transported using one of the DSI transmission modes; Video mode or Command mode. If neither of these mode are selected, the packets are not transmitted through the link and the released FIFOs eventually get overflowed.
The transfer of packets through the APB bus is based on the following conditions:
The APB protocol defines that the write and read procedure takes two clock cycles each to be executed. This means that the maximum input data rate through the APB interfaces is always half the speed of the APB clock.
The data input bus has a maximum width of 32 bits. This allows for a relation to be defined between the input APB clock frequency and maximum bi rate achievable by the APB interface.
The DSI link bit rate when using solely APB is equal to (APB clock frequency) *16 Mbps.
The bandwidth is dependent on the APB clock frequency; the available bandwidth increases with the clock frequency.
To drive the APB interface to achieve high bandwidth Command mode traffic transported by the DSI link, the MIPI Controller should operate in the Command mode only and the APB interface should be the only data source that is currently in use. Thus, the APB interface has the entire bandwidth of the DSI link and does not share it with any another input interface source.
The memory write commands require maximum throughout from the APB interface, because they contain the most amount of data conveyed by the DSI link. While writing the packet information, first write the payload of a given packet into the payload FIFO using the GEN_PLD_DATA register. When the payload data is for the command parameters, place the first byte to be transmitted in the least significant byte position of the APB data bus.
After writing the payload, write the packet header into the command FIFO. For more information and it should follow the pixel to byte conversion organization referred in the Annexure A of the DCS specification. The follow figures show how the pixel data should be organized in the APB data write bus. The memory write commands are conveyed in DCS long packets. DCS long packets are encapsulated in a DSI packet. The DSI included in the diagrams. In the follow figures, the Write Memory Command can be replaced by the DCS command Write Memory Start and Write Memory Continue.

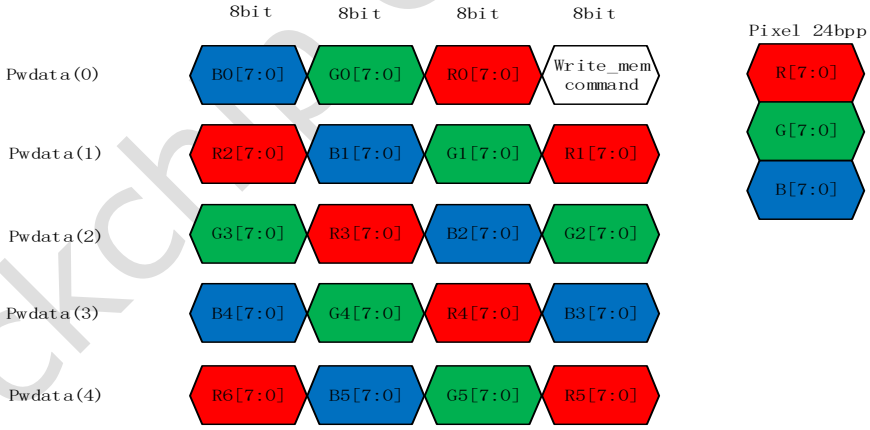


Fig. 18-2 24 bpp APB Pixel to Byte Organization

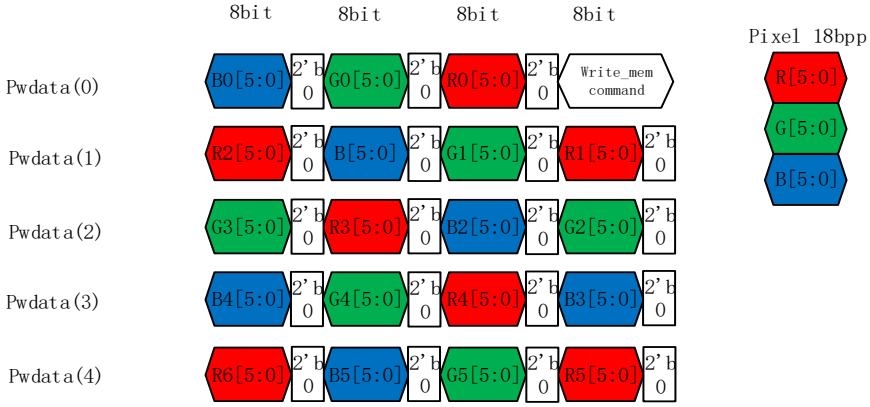


Fig. 18-3 18 bpp APB Pixel to Byte Organization

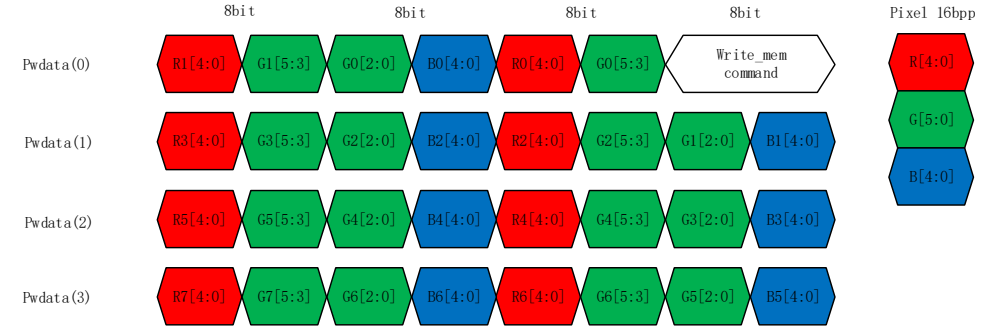


Fig. 18-4 16 bpp APB Pixel to Byte Organization

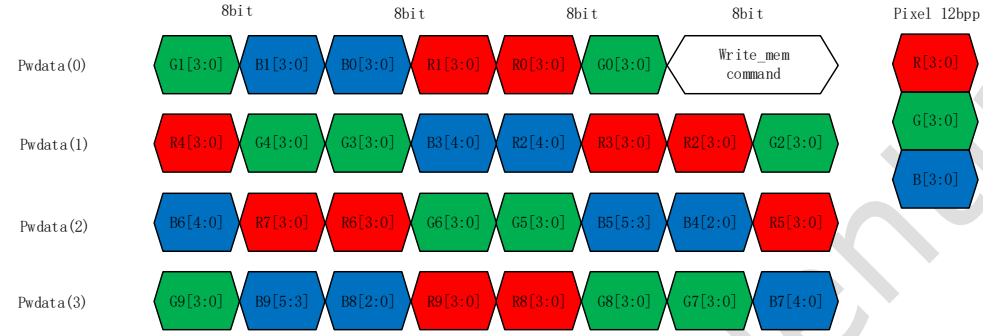


Fig. 18-5 12 bpp APB Pixel to Byte Organization

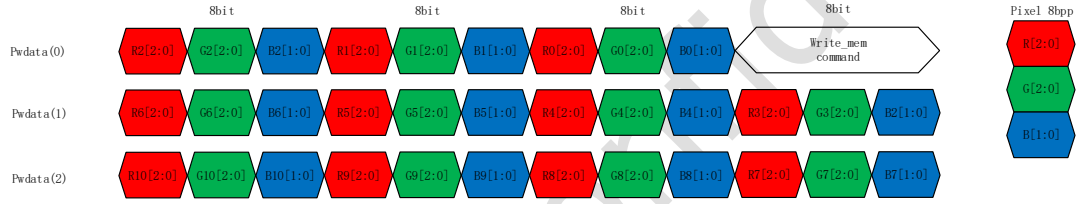


Fig. 18-6 8 bpp APB Pixel to Byte Organization

18.3.3 Transmission of Commands in Video Mode

The MIPI Controller supports the transmission of commands, both in high-speed and low-power, while in Video mode. The DSI controller uses Blanking or Low-Power (BLLP) periods to transmit commands inserted through the APB Generic interface. Those periods correspond to the shaded areas of the following figure.

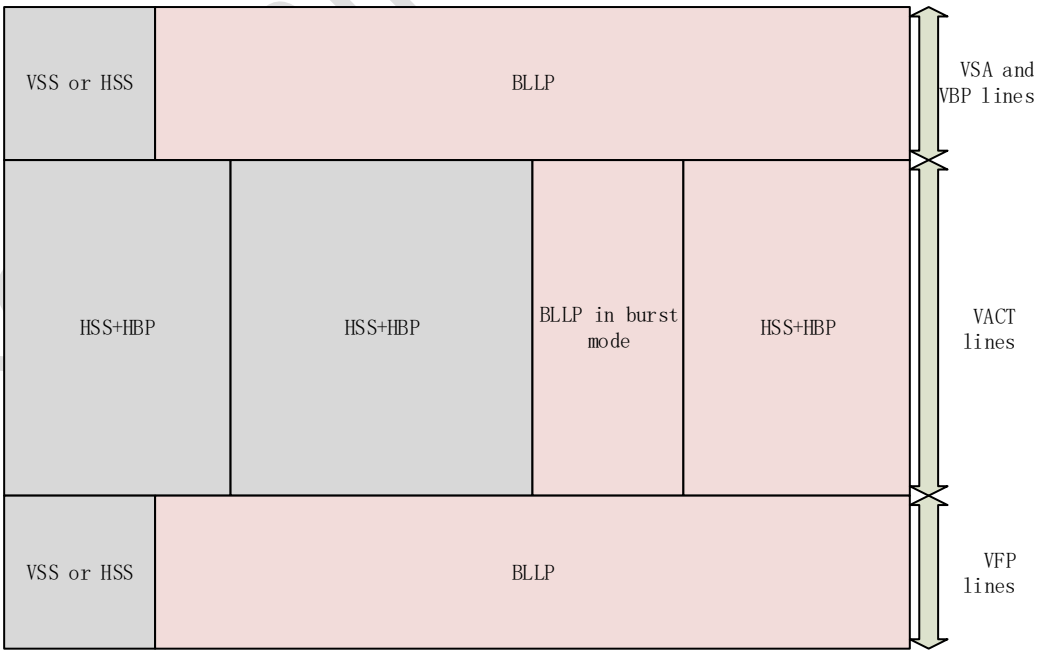


Fig. 18-7 Command Transmission Periods within the Image Area

Commands are transmitted in the blanking periods after the following packet/states:

- Vertical Sync Start (VSS) packets, if the Video Sync pulses are not enabled

- Horizontal Sync End (HSE) packets, in the VSA, VBP, and VFP regions
- Horizontal Sync Start (HSS) packets, if the Video Sync pulses are not enabled in the VSA, VBP, and VFP regions
- Horizontal Active (HACT) state

Only one command is transmitted per line, even in the case of the last line of a frame but one command is possible for each line.
The MIPI Controller avoids sending commands in the last line because it is possible that the last line is shorter than the other ones. For instance, the line time (tL) could be half a cycle longer than the tL on the DPI interface, that is, each line in the frame taking half a cycle from time for the last line. This results in the last line being (1/2 cycle) * (number of lines - 1) shorter than tL.
The dpicolorm and dpishutdn input signals are also able to trigger the sending of command packets. The commands are DSI data types Color Mode On, Color Mode Off, Shut Down Peripheral, and Turn on Peripheral. These commands are not sent in the VACT region. If the lpcmden bit of the VID_MODE_CFG register is 1, these commands are sent in LP mode. In LP mode, the ouvact_lpcmd_time field of the LP_CMD_TIM register is used to determine if these commands can be transmitted. It is assumed that outvact_lpcmd_time is greater than or equal to 4 bytes (number of bytes in a short packet), because the mipi_dsi_host does not transmit these commands on the last line.
If the frame_BTA_ack field is set in the VID_MODE_CFG register, a BTA is generated by mipi_dsi_host after the last line of a frame. This may coincide with a write command or a read command. In either case, the edpihalt signal is held asserted until an acknowledge has been received (control of the DSI bus is returned to the host).
If the lpcmden bit of the VID_MODE_CFG register is set to 1, the commands are sent in low-power in Video mode. In this case, it is necessary to calculate the time available, in bytes, to transmit a command in LP mode for Horizontal Front Porch (HFP), Vertical Sync Active (VSA), Vertical Back Porch (VBP), and Vertical Front Porch(VFP) regions.
The outvact_lpcmd_time field of the LP_CMD_TIM register indicates the time available (in bytes) to transmit a command in LP mode, based on the escape clock, on a line during the VSA, VBP, and the VFP
Outvact_lpcmd_time=(tL-(Time to transmit HSS and HSE frames +tHSA+Time to enter and leave LP mode+Time to send the D-PHY LPDT command)) / escape clock period / 8 / 2
Where,
tL=Line time
tHSA=Time to send a short packet (for sync events) or time of the HAS pulse (for sync pulses)
In the above equation, division by eight is done to convert the time available to bytes and division by two is done because one bit is transmitted once in every two escape clock cycles.
The outvact_lpcmd_time filed can be compared directly with the size of the command to be transmitted to determine if there is enough time to transmit the command. The maximum size of a command that can be transmitted in LP mode is limited to 255 bytes by this field. This register must be programmed to a value greater than or equal to 4 bytes for the transmission of the DCTRL commands such as shutdown and colorm in LP mode.
Consider an example with 12.6 µs per line and assume an escape clock frequency of 15 MHz. In this case, 189 escape clock cycles are available to enter and exit LP mode and transmit command. The following are assumed:
Sync pulses are not being transmitted
Two lane byte clock ticks are required to transmit a short packet
phy_lp2hs_time=16
phy_lp2p_time=20
In this example, a 11-byte command can be transmitted as follows:
outvact_lpcmd_time=(12.6µs-(2*10 ns)-(16*10 ns)-(20*10 ns)-(8*66 ns)) / 66 ns / 8 / 2 = 11 bytes
The invact_lpcmd_time field of the LP_CMD_TIM register indicates the time available (in bytes) to transmite a command in LP mode (based on the escape clock) in the Vertical Active (VACT) region. This time is calculated as follows:

Invact_lpcmd_time= ((tHFP-Time to enter and leave low-power mode + Blanking period before the HFP when in Burst mode- Time to send the D-PHY LPDT command) / escape clock period) / 8
Where,
tHFP=line time-tHSA-tHBP-tHACT
tHACT=vid_pkt_size*bits_per_pixel*lane_byte_clock_period / num_lanes
The invact_lpcmd_time field can be compared directly with the size of the command to be transmitted to determine if there is time to transmit the command.
Consider an example where the refresh rate is 60 Hz. The number of lines is 1320 (typical). The tL in this case is 12.6µs. With a lane byte clock of 100 MHz, 1260 clock ticks are available to transmit a single frame. If 800 ticks are used for pixel data then 460 ticks (4.6µs) are available for Horizontal Sync Start (HSS), HFP, and HBP. Assuming that 2.3µs is available for HFP and the escape clock is 15MHz, only 34 LP clock ticks are available to enter LP, transmit a command, and return from LP mode. Approximately 12 escape clock ticks are required to enter and leave LP mode. Therefore, only 1 byte could be transmitted in this period.
A short packet (for example, generic short write) requires a minimum of 4 bytes. Therefore, in this example, commands are not sent in the VACT region. If Burst mode is enabled, more time is available to transmit commands in the VACT region. The following are assumed:
The controller is not in Burst mode
phy_lp2hs_time=16
phy_lp2hs_time=16
In this examplem invact_lpcmd_time is calculated as follows:
Invact_lpcmd_time = (2.3µs –(16*10 ns)-(20*10 ns)-(8*66 ns)) /66 ns / 8 = 2 bytes
The outvact_lpcmd_time and invact_lpcmd_time fields allow a simple comparison to determine if a command can be transmitted in any of the BLLP periods.
Figure 5-21 illustrates the meaning of invact_lpcmd_time and outvact_lpcmd_time, matching them with the shaded areas and the VACT region.

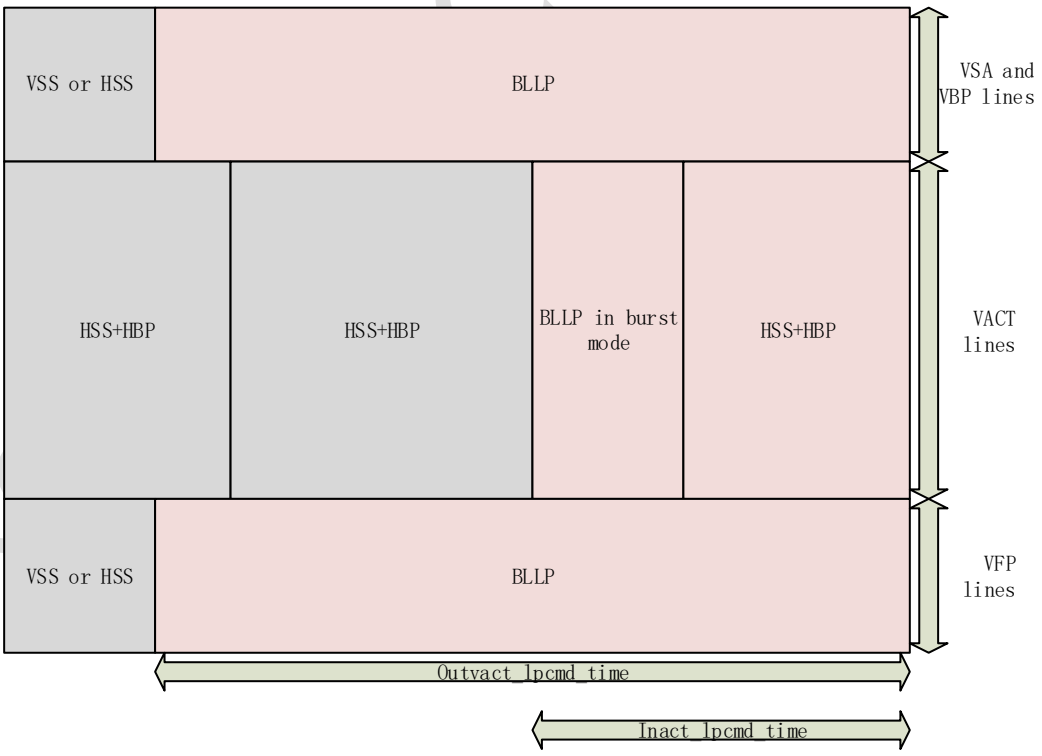


Fig. 18-8 Location of outvact_lpcmd_time and invact_lpcmd_time in the Image Area

If the lpcmden bit of the VID_MODE_CFG register is 0, the commands are sent in high_speed in Video Mode. In this case, the mipi_dsi_host automatically determines the area where each command can be sent and no programming or calculation is required.
On read command Transmission, the max_rd_time field of the PHY_TMR_CFG register

configures the maximum amount of time required to perform a read command in lane byte clock cycles.
The maximum time required to perform a read command in Lane byte clock cycles (max_rd_time) =Time to transmit the read command in LP mode + Time to enter and leave LP mode + Time to return the read data packet from the peripheral device.
The time to return the read data packet from the peripheral depends on the number of bytes read and the escape clock frequency of the peripheral; not the escape clock of the host. The max_rd_time field is used in both HS and LP mode to determine if there is time to complete a read command in a BLLP period.
In high-speed mode (lpcmden=0), max_rd_time is calculated as follows:
max_rd_time = phy_hs2lp_time + Time to return the read data packet from the peripheral device + phy_hs2hs_time
In low-power mode (lpcmden = 1), max_rd_time is calculated as follows:
max_rd_time = phy_hs2lp_time +LPDT command time + Read command time in LP mode + Time to return the data read from the peripheral device + phy_lp2hs_time
Where,
LPDT command time = (8*Host escape clock period) / Lane byte clock period
Read command time in LP mode = (32 * host escape clock period) / lane byte clock period
It is recommended to keep the maximum number of bytes read from the peripheral to a minimum to have sufficient time available to issue the read commands on a line. Ensure that max_rd_time* Lane byte clock period is less than outvact_lpcmd_time *8*Escape clock period of the host.
Otherwise, the read commands are serviced on the last line of a frame and the edpihalt signal may be asserted. If it is necessary to read a large number of parameters (>16), increase the max_rd_time while the read command is being executed. When the read has completed, decrease the max_rd_time to a lower value.

18.4 Register Description

This section describes the control/status registers of the design.

18.4.1 Registers Summary

Name	Offset	Size	Reset Value	Description
MIPIC_VERSION	0x0000	W	0x3132302a	Version of the mipi controller
MIPIC_PWR_UP	0x0004	W	0x00000000	Core power-up
MIPIC_CLKMGR_CFG	0x0008	W	0x00000000	Configuration of the internal clock dividers
MIPIC_DPI_VCID	0x000c	W	0x00000000	The DPI interface configuration.
MIPIC_DPI_COLOR_CODING	0x0010	W	0x00000000	
MIPIC_DPI_CFG_POL	0x0014	W	0x00000000	
MIPIC_LP_CMD_TIM	0x0018	W	0x00000000	Low-power Command Timing Configuration Register.
MIPIC_PCKHDL_CFG	0x002c	W	0x00000000	Packet handler configuration
MIPIC_GEN_VCID	0x0030	W	0x00000000	
MIPIC_MODE_CFG	0x0034	W	0x00000000	
MIPIC_VID_MODE_CFG	0x0038	W	0x00000000	Video mode configuration.
MIPIC_VID_PKT_SIZE	0x003c	W	0x00000000	
MIPIC_VID_NUM_CHUNKS	0x0040	W	0x00000000	
MIPIC_VID_NULL_SIZE	0x0044	W	0x00000000	
MIPIC_VID_HSA_TIME	0x0048	W	0x00000000	Line timing configuration.

Name	Offset	Size	Reset Value	Description
MIPIC_VID_HBP_TIME	0x004c	W	0x00000000	
MIPIC_VID_HLINE_TIME	0x0050	W	0x00000000	
MIPIC_VID_VSA_LINES	0x0054	W	0x00000000	Vertical timing configuration.
MIPIC_VID_VBP_LINES	0x0058	W	0x00000000	
MIPIC_VID_VFP_LINES	0x005c	W	0x00000000	
MIPIC_VID_VACTIVE_LINES	0x0060	W	0x00000000	
MIPIC_EDPI_CMD_SIZE	0x0064	W	0x00000000	
MIPIC_CMD_MODE_CFG	0x0068	W	0x00000000	Command mode configuration
MIPIC_GEN_HDR	0x006c	W	0x00000000	Generic packet header configuration.
MIPIC_GEN_PLD_DATA	0x0070	W	0x00000000	Generic payload data in and out.
MIPIC_CMD_PKT_STATUS	0x0074	W	0x00000000	Command packet status
MIPIC_TO_CNT_CFG	0x0078	W	0x00000000	Timeout timers configuration
MIPIC_HS_RD_TO_CNT	0x007c	W	0x00000000	
MIPIC_LP_RD_TO_CNT	0x0080	W	0x00000000	
MIPIC_HS_WR_TO_CNT	0x0084	W	0x00000000	
MIPIC_LP_WR_TO_CNT	0x0088	W	0x00000000	
MIPIC_BTA_TO_CNT	0x008c	W	0x00000000	
MIPIC_LPCLK_CTRL	0x0094	W	0x00000000	
MIPIC_PHY_TMR_LPCLK_CFG	0x0098	W	0x00000000	
MIPIC_PHY_TMR_CFG	0x009c	W	0x00000000	D-PHY timing configuration
MIPIC_PHY_RSTZ	0x00a0	W	0x00000000	D-PHY reset control
MIPIC_PHY_IF_CFG	0x00a4	W	0x00000000	D-PHY interface configuration
MIPIC_PHY_ULPS_CTRL	0x00a8	W	0x00000000	D-PHY PPI interface control
MIPIC_PHY_TX_TRIGGERS	0x00ac	W	0x00000000	
MIPIC_PHY_STATUS	0x00b0	W	0x00000000	D-PHY PPI status interface
MIPIC_RESERVED3	0x00b4	W	0x00000000	Reserved
MIPIC_RESERVED4	0x00b8	W	0x00000000	Reserved
MIPIC_ERROR_ST0	0x00bc	W	0x00000000	Interrupt status register 0
MIPIC_ERROR_ST1	0x00c0	W	0x00000000	Interrupt status register 1
MIPIC_MSK0	0x00c4	W	0x00000000	Masks the interrupt generation triggerd by the ERROR_ST0 reg
MIPIC_MSK1	0x00c8	W	0x00000000	Masks the interrupt generation triggerd by the ERROR_ST1 reg

*Notes:Size:**B**- Byte (8 bits) access, **HW**- Half WORD (16 bits) access, **W**-WORD (32 bits) access*

18.4.2 Detail Register Description

MIPIC_VERSION
Address: Operational Base + offset (0x0000)
Version of the mipi controller

Bit	Attr	Reset Value	Description
31:0	RO	0x3132302a	version indicates the version of the mipi_controller

MIPIC_PWR_UP
Address: Operational Base + offset (0x0004)
Core power-up

Bit	Attr	Reset Value	Description
31:1	RO	0x0	reserved
0	RW	0x0	shutdownz This bit indicates the core power-up or the reset 0-Reset 1-Power-up

MIPIC_CLKMGR_CFG
Address: Operational Base + offset (0x0008)
Configuration of the internal clock dividers

Bit	Attr	Reset Value	Description
31:16	RO	0x0	reserved
15:8	RW	0x00	TO_CLK_DIVISION This field indicates the division factor for the Time Out clock used as the timing unit in the configuration of HS to LP and LP to HS transition error.
7:0	RW	0x00	TX_ESC_CLK_DIVISION Field0000 Abstract This field indicates the division factor for the TX_Escape clock source (lanebyteclk).The value 0 and 1 stop the TX_ESC clock generation

MIPIC_DPI_VCID
Address: Operational Base + offset (0x000c)
The DPI interface configuration.

Bit	Attr	Reset Value	Description
31:2	RO	0x0	reserved
1:0	RW	0x0	dpi_vid This field configures the DPI virtual channel id that is indexed to the Video mode packets.

MIPIC_DPI_COLOR_CODING
Address: Operational Base + offset (0x0010)

Bit	Attr	Reset Value	Description
31:9	RO	0x0	reserved
8	RW	0x0	en18_loosely When set to 1,this bit enables 18 loosely packed pixel stream.
7:4	RO	0x0	reserved

Bit	Attr	Reset Value	Description
3:0	RW	0x0	dpi_color_coding This field configures the DPI color coding as follows: 000:16bit configuration 1 001:16bit configuration 2 010:16bit configuration 3 011:18bit configuration 1 100:18bit configuration 2 101,110,and 111:24bits

MIPIC_DPI_CFG_POL
Address: Operational Base + offset (0x0014)

Bit	Attr	Reset Value	Description
31:5	RO	0x0	reserved
4	RW	0x0	colorm_active_low When set to 1,this bit configures the color mode pin as active low
3	RW	0x0	shutd_active_low When set to 1,this bit configures the shut down pin as active low
2	RW	0x0	hsync_active_low When set to 1,this bit configures the horizontal synchronism pin as active low.
1	RW	0x0	vsync_active_low When set to 1,this bit configures the vertical synchronism pin as active low
0	RW	0x0	dataen_active_low When set to 1,this bit configures the data enable pin as active low

MIPIC_LP_CMD_TIM
Address: Operational Base + offset (0x0018)
Low-power Command Timing Configuration Register.

Bit	Attr	Reset Value	Description
31:24	RO	0x0	reserved
23:16	RW	0x00	outvact_lpcmd_time outside VACT region command time. This field configures the time available to transmit a command in low-power mode. The time value is expressed in a number of bytes format. The number of bytes represents the maximum size of a packet that can fit in a line during the VSA, VBP, and VFP region. This field must be configured with a value greater than or equal to four bytes to allow the transmission of the DCTRL commands such as shutdown and colorm in low-power mode.
15:8	RO	0x0	reserved
7:0	RW	0x00	invact_lpcmd_time Inside VACT region command time. This field configures the time available to transmit a command in low-power mode. The time value is expressed in a number of bytes format. The number of bytes represents the maximum size of the packet that can fit a line during the VACT region.

MIPIC_PCKHDL_CFG
Address: Operational Base + offset (0x002c)
Packet handler configuration

Bit	Attr	Reset Value	Description
31:5	RO	0x0	reserved
4	RW	0x0	en_CRC_rx When set to 1,this bit enables the CRC reception and error reporting
3	RW	0x0	en_ECC_rx When set to 1,this bit enables the ECC reception, error correction, and reporting
2	RW	0x0	en_BTA When set to 1,this bit enables the Bus Turn-Around(BTA) request.
1	RW	0x0	en_EOTp_rx When set to 1,this bit enables the EOTp reception
0	RW	0x0	en_EOTp_tx When set to 1,this bit enables the EOTp transmission

MIPIC_GEN_VCID
Address: Operational Base + offset (0x0030)
Packet handler configuration

Bit	Attr	Reset Value	Description
31:2	RO	0x0	reserved
1:0	RW	0x0	gen_vid_rx This field indicates the Generic interface read-back virtual channel identification

MIPIC_MODE_CFG
Address: Operational Base + offset (0x0034)

Bit	Attr	Reset Value	Description
31:1	RO	0x0	reserved
0	RW	0x0	en_video_mode When set to 1,this bit enables the DPI Video mode transmission.

MIPIC_VID_MODE_CFG
Address: Operational Base + offset (0x0038)
Video mode configuration.

Bit	Attr	Reset Value	Description
31:16	RO	0x0	reserved
15	RW	0x0	lpcmden When set to 1,this bit enables the command transmission only in low-power mode
14	RW	0x0	frame_BTA_ack When set to 1,this bit enables the request for an acknowledge response at the end of a frame
13	RW	0x0	en_lp_hfp When set to 1,this bit enables the return to low-power inside the HFP period when timing allows.

Bit	Attr	Reset Value	Description
12	RW	0x0	en_lp_hbp When set to 1,this bit enables the return to low-power inside the HBP period when timing allows.
11	RW	0x0	en_lp_vact When set to 1,this bit enables the return to low-power inside the VACT period when timing allows.
10	RW	0x0	en_lp_vfp When set to 1,this bit enables the return to low-power inside the VFP period when timing allows.
9	RW	0x0	en_lp_vbp When set to 1,this bit enables the return to low-power inside the VBP period when timing allows.
8	RW	0x0	en_lp_vsa When set to 1,this bit enables the return to low-power inside the VSA period when timing allows.
7:2	RO	0x0	reserved
1:0	RW	0x0	vid_mode_type This field indicates the video mode transmission type as follows: 00:Non-burst with sync pulses 01:Non-burst with sync events 10 and 11:Burst with sync pulses

MIPIC_VID_PKT_SIZE
Address: Operational Base + offset (0x003c)

Bit	Attr	Reset Value	Description
31:14	RO	0x0	reserved
13:0	RW	0x0000	vid_pkt_size This field configures the number of pixels on a single video packet. if you use the 18-bit mode and do not enable loosely packed stream, this value must be a multiple of 4.

MIPIC_VID_NUM_CHUNKS
Address: Operational Base + offset (0x0040)

Bit	Attr	Reset Value	Description
31:13	RO	0x0	reserved
12:0	RW	0x0000	num_chunks This field configures the number of chunks to be transmitted during a line period(a chunk is a video packet or a null packet)

MIPIC_VID_NULL_SIZE
Address: Operational Base + offset (0x0044)

Bit	Attr	Reset Value	Description
31:13	RO	0x0	reserved

Bit	Attr	Reset Value	Description
12:0	RW	0x0000	null_pkt_size This field configures the number of bytes in a null packet

MIPIC_VID_HSA_TIME

Address: Operational Base + offset (0x0048)

Line timing configuration.

Bit	Attr	Reset Value	Description
31:12	RO	0x0	reserved
11:0	RW	0x000	hsa_time This field configures the Horizontal Synchronism Active period in lane byte clock cycles.

MIPIC_VID_HBP_TIME

Address: Operational Base + offset (0x004c)

Bit	Attr	Reset Value	Description
31:12	RO	0x0	reserved
11:0	RW	0x000	hbp_time This field configures the Horizontal Back Porch period in lane byte clock cycles

MIPIC_VID_HLINE_TIME

Address: Operational Base + offset (0x0050)

Bit	Attr	Reset Value	Description
31:15	RO	0x0	reserved
14:0	RW	0x0000	hline_time This field configures the size of the total lines counted in lane byte cycles.

MIPIC_VID_VSA_LINES

Address: Operational Base + offset (0x0054)

Vertical timing configuration.

Bit	Attr	Reset Value	Description
31:10	RO	0x0	reserved
9:0	RW	0x000	vsa_lines This field configures the Vertical Synchronism Active period measured in number of horizontal lines.

MIPIC_VID_VBP_LINES
Address: Operational Base + offset (0x0058)

Bit	Attr	Reset Value	Description
31:10	RO	0x0	reserved
9:0	RW	0x000	vbp_lines This field configures the Vertical Back Porch period measured in horizontal lines.

MIPIC_VID_VFP_LINES
Address: Operational Base + offset (0x005c)

Bit	Attr	Reset Value	Description
31:10	RO	0x0	reserved
9:0	RW	0x000	vfp_lines This field configures the Vertical Front Porch period measured in horizontal lines.

MIPIC_VID_VACTIVE_LINES
Address: Operational Base + offset (0x0060)

Bit	Attr	Reset Value	Description
31:14	RO	0x0	reserved
13:0	RW	0x0000	v_active_line This field configures the Vertical Active period measured in horizontal lines.

MIPIC_EDPI_CMD_SIZE
Address: Operational Base + offset (0x0064)

Bit	Attr	Reset Value	Description
31:16	RO	0x0	reserved
15:0	RW	0x0000	edpi_allowed_cmd_size This field configures the maximum allowed size for an eDPI write memory command, measured in pixels. Automatic partitioning of data obtained from eDPI is permanently enabled.

MIPIC_CMD_MODE_CFG
Address: Operational Base + offset (0x0068)

Command mode configuration

Bit	Attr	Reset Value	Description
31:25	RO	0x0	reserved
24	RW	0x0	max_rd_pkt_size This bit configures the maximum read packet size command transmission type: 0:High-speed 1:Low-power
23:20	RO	0x0	reserved
19	RW	0x0	dcs_lw_tx This bit configures the DCS long write packet command transmission type: 0:High-speed 1:Low-power
18	RW	0x0	dcs_sr_0p_tx This bit configures the DCS short read packet with zero parameter command transmission type: 0:High-speed 1:Low-power
17	RW	0x0	dcs_sw_1p_tx This bit configures the DCS short write packet with one parameter command transmission type: 0:High-speed 1:Low-power
16	RW	0x0	dcs_sw_0p_tx This bit configures the DCS short write packet with zero parameter command transmission type: 0:High-speed 1:Low-power
15	RO	0x0	reserved
14	RW	0x0	gen_lw_tx This bit configures the Generic long write packet command 0:High-speed 1:Low-power
13	RW	0x0	gen_sr_2p_tx This bit configures the Generic short read packet with two parameter command transmission type: 0:High-speed 1:Low-power
12	RW	0x0	gen_sr_1p_tx This bit configures the Generic short read packet with one parameter command transmission type: 0:High-speed 1:Low-power

Bit	Attr	Reset Value	Description
11	RW	0x0	gen_sr_0p_tx This bit configures the Generic short read packet with zero parameter command transmission type: 0:High-speed 1:Low-power
10	RW	0x0	gen_sw_2p_tx This bit configures the Generic short write packet with two parameter command transmission type: 0:High-speed 1:Low-power
9	RW	0x0	gen_sw_1p_tx This bit configures the Generic short write packet with one parameter command transmission type: 0:High-speed 1:Low-power
8	RW	0x0	gen_sw_0p_tx This bit configures the Generic short write packet with zero parameter command transmission type: 0:High-speed 1:Low-power
7:2	R0	0x0	reserved
1	RW	0x0	ack_rqst_en When
0	RW	0x0	tear_fx_en When set to 1,this bit enables the tearing effect acknowledge request

MIPIC_GEN_HDR

Address: Operational Base + offset (0x006c)

Generic packet header configuration.

Bit	Attr	Reset Value	Description
31:24	RO	0x0	reserved
23:16	RW	0x00	gen_WC_MSbyte This field configures the most significant byte of the header packet's Word count for long packets or data 1 for short packets.
15:8	RW	0x00	gen_WC_LSbyte This field configures the least significant byte of the header packet's Word count for long packets or data 0 for short packets.
7:6	RW	0x0	gen_VC This field configures the virtual channel id of the header packet.
5:0	RW	0x00	gen_DT This field configures the packet data type of the header packet

MIPIC_GEN_PLD_DATA

Address: Operational Base + offset (0x0070)

Generic payload data in and out.

Bit	Attr	Reset Value	Description
31:24	RW	0x00	gen_pld_b4 This field indicates byte 4 of the packet payload.
23:16	RW	0x00	gen_pld_b3 This field indicates byte 3 of the packet payload.
15:8	RW	0x00	gen_pld_b2 This field indicates byte 2 of the packet payload.
7:0	RW	0x00	gen_pld_b1 This field indicates byte 1 of the packet payload.

MIPIC_CMD_PKT_STATUS

Address: Operational Base + offset (0x0074)

Command packet status

Bit	Attr	Reset Value	Description
31:8	RO	0x0	reserved
7	RW	0x0	reserved reserved
6	RW	0x0	gen_rd_cmd_busy This bit is set when a read command is issued and cleared when the entire response is stored in the FIFO
5	RW	0x0	gen_pld_r_full This bit indicates the full status of the generic read payload FIFO Vaule after reset:0x0
4	RO	0x0	gen_pld_r_empty This bit indicates the empty status of the generic read payload FIFO Vaule after reset:0x1
3	RO	0x0	gen_pld_w_full This bit indicates the full status of the generic write payload FIFO Vaule after reset:0x0
2	RO	0x0	gen_pld_w_empty This bit indicates the empty status of the generic write payload FIFO Vaule after reset:0x1
1	RO	0x0	gen_cmd_full This bit indicates the full status of the generic command FIFO Vaule after reset:0x0
0	RO	0x0	gen_cmd_empty This bit indicates the empty status of the generic command FIFO Vaule after reset:0x1

MIPIC_TO_CNT_CFG

Address: Operational Base + offset (0x0078)
Timeout timers configuration

Bit	Attr	Reset Value	Description
31:16	RW	0x0000	hstx_to_cnt This field configures the timeout counter that triggers a high-speed transmission timeout contention detection(measured in TO_CLK_DIVISION cycles)
15:0	RW	0x0000	lprx_to_cnt This field configures the timeout counter that triggers a low-power reception timeout contention detection(measured in TO_CLK_DIVISION cycles)

MIPIC_HS_RD_TO_CNT
Address: Operational Base + offset (0x007c)

Bit	Attr	Reset Value	Description
31:1	RO	0x0	reserved
0	RW	0x0	hs_rd_to_cnt This field sets a period for which the MIPI Controller keeps the link still, after sending a high-speed read operation. This period is measured in cycles of lanebyteclk. The counting starts when the D-PHY enters the Stop state and causes no interrupts.

MIPIC_LP_RD_TO_CNT
Address: Operational Base + offset (0x0080)

Bit	Attr	Reset Value	Description
31:16	RO	0x0	reserved
15:0	RW	0x0000	lp_rd_to_cnt This field sets a period for which MIPI Controller keeps the link still, after sending a low-power read operation. This period is measured in cycles of lanebyteclk. The counting starts when the D-PHY enters the Stop state and causes no interrupts.

MIPIC_HS_WR_TO_CNT
Address: Operational Base + offset (0x0084)

Bit	Attr	Reset Value	Description
31:25	RO	0x0	reserved

Bit	Attr	Reset Value	Description
24	RW	0x0	presp_to_mode When set to 1, this bit ensures that the peripheral response timeout caused by hs_wr_to_cnt is used only once per eDPI frame, when both the following conditions are met: dpivsync_edpiwms has risen and fallen Packets originated from eDPI have been transmitted and its FIFO is empty again.
23:16	RO	0x0	reserved
15:0	RW	0x0000	hs_wr_to_cnt This field sets a period for which the MIPI Controller keeps the link inactive after sending a high-speed write operation. This period is measured in cycles of lanebyteclk. The counting starts when the D-PHY enters the Stop state and causes no interrupts.

MIPIC_LP_WR_TO_CNT
Address: Operational Base + offset (0x0088)

Bit	Attr	Reset Value	Description
31:16	RO	0x0	reserved
15:0	RW	0x0	lp_wr_to_cnt This field sets a period for which the DSI Controller keeps the link still, after sending a low-power write operation. This period is measured in cycles of lanebyteclk. The counting starts when the D-PHY enters the Stop state and causes no interrupts.

MIPIC_BTA_TO_CNT
Address: Operational Base + offset (0x008c)

Bit	Attr	Reset Value	Description
31:16	RO	0x0	reserved
15:0	RW	0x0000	bta_to_cnt This field sets a period for which the DSI Controller keeps the link still, after completing a Bus Turn-Around. This period is measured in cycles of lanebyteclk. The counting starts when the D-PHY enters the Stop state and causes no interrupts.

MIPIC_LPCLK_CTRL
Address: Operational Base + offset (0x0094)

Bit	Attr	Reset Value	Description
31:2	RO	0x0	reserved

Bit	Attr	Reset Value	Description
1	RW	0x0	auto_clklane_ctrl This bit enables the automatic mechanism to stop providing clock in the clock lane when time allows.
0	RW	0x0	phy_txrequestclkhs This bit controls the D-PHY PPI tx requestclkhs signal

MIPIC_PHY_TMR_LPCLK_CFG

Address: Operational Base + offset (0x0098)

Bit	Attr	Reset Value	Description
31:26	RO	0x0	reserved
25:16	RW	0x000	phy_hs2lp_time This field configures the maximum time that the PHY takes to go from high-speed to low-power transmission measured in lane byte clock cycles.(clock lane)
15:10	RO	0x0	reserved
9:0	RW	0x000	phy_lp2hs_time This field configures the maximum time that the PHY takes to go from low-power to high-speed transmission measured in lane byte clock cycles.(clock lane)

MIPIC_PHY_TMR_CFG

Address: Operational Base + offset (0x009c)

D-PHY timing configuration

Bit	Attr	Reset Value	Description
31:24	RW	0x00	phy_hs2lp_time This field configures the maximum time that the PHY takes to go from high-speed to low-power transmission measured in lane byte clock cycles.
23:16	RW	0x00	phy_lp2hs_time This field configures the maximum time that the PHY takes to go from low-power to high-speed transmission measured in lane byte clock cycles.
15	RW	0x0	reserved reserved for future use
14:0	RW	0x0000	max_rd_time This field configures the maximum time required to perform a read command in lane byte clock cycles. This register can only by modified when read commands are not in progress.

MIPIC_PHY_RSTZ

Address: Operational Base + offset (0x00a0)

D-PHY reset control

Bit	Attr	Reset Value	Description
31:3	RO	0x0	reserved
2	RW	0x0	phy_enableclk When set to 1,this bit enables the D-PHY Clock Lane Module
1	RW	0x0	reserved1
0	RW	0x0	reserved

MIPIC_PHY_IF_CFG

Address: Operational Base + offset (0x00a4)

D-PHY interface configuration

Bit	Attr	Reset Value	Description
31:16	RO	0x0	reserved
15:8	RW	0x00	phy_stop_wait_time This field configures the minimum wait period to request a high-speed transmission after the Stop state is accounted in clock lane cycles.
7:2	RO	0x0	reserved
1:0	RW	0x0	n_lanes This field configures the number of active data lanes: 00:One data lane(lane 0) 01:Two data lane(lanes 0 and 1) 10:Three data lanes(lanes 0,1,and 2) 11:Four data lanes(lanes 0,1,2,and 3)

MIPIC_PHY_ULPS_CTRL

Address: Operational Base + offset (0x00a8)

D-PHY PPI interface control

Bit	Attr	Reset Value	Description
31:4	RO	0x0	reserved
3	RW	0x0	phy_txexitulpslan ULPS mode Exit on all active data lanes
2	RW	0x0	phy_txrequlpslan ULPS mode Request on all active data lanes
1	RW	0x0	phy_txexitulpsclk ULPS mode Exit on clock lane
0	RW	0x0	phy_txrequlpsclk ULPS mode Request on clock lane

MIPIC_PHY_TX_TRIGGERS

Address: Operational Base + offset (0x00ac)

Bit	Attr	Reset Value	Description
31:4	RO	0x0	reserved

Bit	Attr	Reset Value	Description
3:0	RW	0x0	phy_tx_triggers This field controls the trigger transmissions.

MIPIC_PHY_STATUS

Address: Operational Base + offset (0x00b0)

D-PHY PPI status interface

Bit	Attr	Reset Value	Description
31:13	RO	0x0	reserved
12	RO	0x0	ulpsactivenot3lane This bit indicates the status of ulpsactivenot3lane D-PHY signal
11	RO	0x0	phystopstate3lane This bit indicates the status of phystopstate3lane D-PHY signal
10	RO	0x0	ulpsactivenot2lane This bit indicates the status of ulpsactivenot2lane D-PHY signal
9	RO	0x0	phystopstate2lane This bit indicates the status of phystopstate2lane D-PHY signal
8	RO	0x0	ulpsactivenot1lane This bit indicates the status of ulpsactivenot1lane D-PHY signal
7	RO	0x0	phystopstate1lane This bit indicates the status of phystopstate1lane D-PHY signal
6	RW	0x0	rxulpsesc0lane This bit indicates the status of rxulpsesc0lane D-PHY signal
5	RO	0x0	ulpsactivenot0lane This bit indicates the status of ulpsactivenot0lane D-PHY signal
4	RO	0x0	phystopstate0lane This bit indicates the status of phystopstate0lane D-PHY signal
3	RO	0x0	phyulpsactivenotclk This bit indicates the status of phyulpsactivenotclk D-PHY signal
2	RO	0x0	phystopstateclklane This bit indicates the status of phystopstateclklane D-PHY signal
1	RO	0x0	phydirection This bit indicates the status of phydirection D-PHY signal
0	RO	0x0	phylock This bit indicates the status of phylock D-PHY signal

MIPIC_RESERVED3

Address: Operational Base + offset (0x00b4)

Reserved

Bit	Attr	Reset Value	Description
31:1	RO	0x0	reserved
0	RW	0x0	Reserved

MIPIC_RESERVED4

Address: Operational Base + offset (0x00b8)

Reserved

Bit	Attr	Reset Value	Description
31:1	RO	0x0	reserved
0	RW	0x0	Reserved

MIPIC_ERROR_ST0

Address: Operational Base + offset (0x00bc)

Interrupt status register 0

Bit	Attr	Reset Value	Description
31:21	RO	0x0	reserved
20	RO	0x0	dphy_errors_4 This bit indicates LP1 contention error ErrContentionLP1 from Lane 0
19	RO	0x0	dphy_errors_3 This bit indicates LP0 contention error ErrContentionLP0 from Lane 0
18	RO	0x0	dphy_errors_2 This bit indicates control error ErrControl from Lane 0
17	RO	0x0	dphy_errors_1 This bit indicates ErrSyncEsc low-power data transmission synchronization error from Lane 0
16	RO	0x0	dphy_errors_0 This bit indicates ErrEsc escape entry error from Lane 0
15	RO	0x0	ack_with_err_15 This bit retrieves the DSI protocol violation from the Display Acknowledge error report
14	RO	0x0	ack_with_err_14 This bit retrieves the reserved(specific to device) from the Display Acknowledge error report
13	RO	0x0	ack_with_err_13 This bit retrieves the invalid transmission length from the Display Acknowledge error report
12	RO	0x0	ack_with_err_12 This bit retrieves the DSI VC ID Invalid from the Display Acknowledge error report
11	RO	0x0	ack_with_err_11 This bit retrieves the not recognized DSI data type from the Display Acknowledge error report
10	RO	0x0	ack_with_err_10 This bit retrieves the checksum error(long packet only) from the Display Acknowledge error report

Bit	Attr	Reset Value	Description
9	RO	0x0	ack_with_err_9 This bit retrieves the ECC error, multi-bit(detected and corrected) from the Display Acknowledge error report
8	RO	0x0	ack_with_err_8 This bit retrieves the ECC error, single-bit(detected and corrected) from the Display Acknowledge error report
7	RO	0x0	ack_with_err_7 This bit retrieves the reserved(specific to device) error from the Display Acknowledge error report
6	RO	0x0	ack_with_err_6 This bit retrieves the False Control error from the Display Acknowledge error report
5	RO	0x0	ack_with_err_5 This bit retrieves the HS Receive Timeout error from the Display Acknowledge error report
4	RO	0x0	ack_with_err_4 This bit retrieves the LP Transmit Sync error from the Display Acknowledge error report
3	RO	0x0	ack_with_err_3 This bit retrieves the Escape Mode Entry command error from the Display Acknowledge error report
2	RO	0x0	ack_with_err_2 This bit retrieves the EoT sync error from the Display Acknowledge error report
1	RO	0x0	ack_with_err_1 This bit retrieves the SoT Sync error from the Display Acknowledge error report
0	RO	0x0	ack_with_err_0 This bit retrieves the SoT error from the Display Acknowledge error report

MIPIC_ERROR_ST1

Address: Operational Base + offset (0x00c0)

Interrupt status register 1

Bit	Attr	Reset Value	Description
31:13	RO	0x0	reserved
12	RO	0x0	gen_pld_recv_err This bit indicates that during a generic interface packet read back, the payload FIFO becomes full and the received data is corrupted.
11	RO	0x0	gen_pld_rd_err This bit indicates that during a DCS read data,the payload FIFO becomes empty and the data sent to the interface is corrupted

Bit	Attr	Reset Value	Description
10	RO	0x0	gen_pld_send_err This bit indicates that during a Generic interface packet build, the payload FIFO becomes empty and corrupt data is sent.
9	RO	0x0	gen_pld_wr_err This bit indicates that the system tried to write a payload data through the Generic interface and the FIFO is full. Therefore, the command is not written.
8	RO	0x0	gen_cmd_wr_err This bit indicates that the system tried to write a command through the Generic interface and the FIFO is full. Therefore, the command is not written.
7	RO	0x0	dpi_pld_wr_err This bit indicates that during a DPI pixel line storage, the payload FIFO becomes full and the data stored is corrupted.
6	RO	0x0	eopt_err This bit indicates that the EOTp packet is not received at the end of the incoming peripheral transmission.
5	RO	0x0	pkt_size_err This bit indicates that the packet size error is detected during the packet reception.
4	RO	0x0	crc_err This bit indicates that the CRC error is detected in a received packet.
3	RO	0x0	ecc_multi_err This bit indicates that the ECC multiple error is detected and corrected in a received packet.
2	RO	0x0	ecc_single_err This bit indicates that the ECC single error is detected and corrected in a received packet.
1	RO	0x0	to_lp_rx This bit indicates that the low-power reception timeout counter reached the end and contention detection is detected.
0	RO	0x0	to_hs_tx This bit indicates that the high-speed transmission timeout counter reached the end and contention detection is detected.

MIPIC_MSK0

Address: Operational Base + offset (0x00c4)

Masks the interrupt generation triggerd by the ERROR_ST0 reg

Bit	Attr	Reset Value	Description
31:21	RO	0x0	reserved
20	RW	0x0	dphy_errors_4 This bit indicates LP1 contention error ErrContentionLP1 from Lane 0

Bit	Attr	Reset Value	Description
19	RW	0x0	dphy_errors_3 This bit indicates LP0 contention error ErrContentionLP0 from Lane 0
18	RW	0x0	dphy_errors_2 This bit indicates control error ErrControl from Lane 0
17	RW	0x0	dphy_errors_1 This bit indicates ErrSyncEsc low-power data transmission synchronization error from Lane 0
16	RW	0x0	dphy_errors_0 This bit indicates ErrEsc escape entry error from Lane 0
15	RW	0x0	ack_with_err_15 This bit retrieves the DSI protocol violation from the Display Acknowledge error report
14	RW	0x0	ack_with_err_14 This bit retrieves the reserved(specific to device) from the Display Acknowledge error report
13	RW	0x0	ack_with_err_13 This bit retrieves the invalid transmission length from the Display Acknowledge error report
12	RW	0x0	ack_with_err_12 This bit retrieves the DSI VC ID Invalid from the Display Acknowledge error report
11	RW	0x0	ack_with_err_11 This bit retrieves the not recognized DSI data type from the Display Acknowledge error report
10	RW	0x0	ack_with_err_10 This bit retrieves the checksum error(long packet only) from the Display Acknowledge error report
9	RW	0x0	ack_with_err_9 This bit retrieves the ECC error,multi-bit(detected and corrected) from the Display Acknowledge error report
8	RW	0x0	ack_with_err_8 This bit retrieves the ECC error,single-bit(detected and corrected) from the Display Acknowledge error report
7	RW	0x0	ack_with_err_7 This bit retrieves the reserved(specific to device) error from the Display Acknowledge error report
6	RW	0x0	ack_with_err_6 This bit retrieves the False Control error from the Display Acknowledge error report
5	RW	0x0	ack_with_err_5 This bit retrieves the HS Receive Timeout error from the Display Acknowledge error report

Bit	Attr	Reset Value	Description
4	RW	0x0	ack_with_err_4 This bit retrieves the LP Transmit Sync error error from the Display Acknowledge error report
3	RW	0x0	ack_with_err_3 This bit retrieves the Escape Mode Entry command error from the Display Acknowledge error report
2	RW	0x0	ack_with_err_2 This bit retrieves the EoT sync error from the Display Acknowledge error report
1	RW	0x0	ack_with_err_1 This bit retrieves the SoT Sync error from the Display Acknowledge error report
0	RW	0x0	ack_with_err_0 This bit retrieves the SoT error from the Display Acknowledge error report

MIPIC_MSK1

Address: Operational Base + offset (0x00c8)

Masks the interrupt generation triggerd by the ERROR_ST1 reg

Bit	Attr	Reset Value	Description
31:13	RO	0x0	reserved
12	RO	0x0	gen_pld_recv_err This bit indicates that during a generic interface packet read back, the payload FIFO becomes full and the received data is corrupted.
11	RO	0x0	gen_pld_rd_err This bit indicates that during a DCS read data, the payload FIFO becomes empty and the data sent to the interface is corrupted
10	RO	0x0	gen_pld_send_err This bit indicates that during a Generic interface packet build, the payload FIFO becomes empty and corrupt data is sent.
9	RO	0x0	gen_pld_wr_err This bit indicates that the system tried to write a payload data through the Generic interface and the FIFO is full. Therefore, the command is not written.
8	RO	0x0	gen_cmd_wr_err This bit indicates that the system tried to write a command through the Generic interface and the FIFO is full. Therefore, the command is not written.
7	RO	0x0	dpi_pld_wr_err This bit indicates that during a DPI pixel line storage, the payload FIFO becomes full and the data stored is corrupted.

Bit	Attr	Reset Value	Description
6	RO	0x0	eopt_err This bit indicates that the EOTp packet is not received at the end of the incoming peripheral transmission.
5	RO	0x0	pkt_size_err This bit indicates that the packet size error is detected during the packet reception.
4	RO	0x0	crc_err This bit indicates that the CRC error is detected in a received packet.
3	RO	0x0	ecc_multi_err This bit indicates that the ECC multiple error is detected and corrected in a received packet.
2	RO	0x0	ecc_single_err This bit indicates that the ECC single error is detected and corrected in a received packet.
1	RO	0x0	to_lp_rx This bit indicates that the low-power reception timeout counter reached the end and contention detection is detected.
0	RO	0x0	to_hs_tx This bit indicates that the high-speed transmission timeout counter reached the end and contention detection is detected.

18.5 Application Notes

Low Power Mode is a special feature for D-PHY. You can control this function by using proper registers from the Innosilicon D-PHY with few operations. The following is a step by step instruction for low power mode in and out.

Perform the following steps to configure the DPI packet transmission:
Step1:Global configuration:
Configure n_lanes (PHY_IF_CFG-[1:0]) to define the number of lanes in which the controller has to perform high-speed transmissions.

Step2:Configure the DPI Interface to define how the DPI interface interacts with the controller.
Configure dpi_vid (DPI_CFG-[1:0]): This field configures the virtual channel that the packet generated by the DPI interface is indexed to.
Configure dpi_color_coding (DPI_CFG-[4:2]): This field configures the bits per pixels that the interface transmits and also the variant configuration of each bpp. If you select 18 bpp, and the Enable_18_loosely_packed is not active, the number or pixels per line should be a multiple of four.
Configure dataen_active_low (DPI_CFG-[5]): This bit configures the polarity of the dpidataen signal and enables if it is active low.
Configure vsync_active_low(DPI_CFG-[6]): This bit configures the polarity of the dpivsync signal and enables if it is active low.
Configure vsync_active_low(DPI_CFG-[7]): This bit configures the polarity of the dpivsync signal and enables if it is active low.
Configure vsync_active_low(DPI_CFG-[8]): This bit configures the polarity of the dpishutdn signal and enables if it is active low.
Configure vsync_active_low(DPI_CFG-[9]): This bit configures the polarity of the dpicolorm signal and enables if it is active low.

Configure en 18_loosely(DPI_CFG-[10]): This bit configures if the pixel packing is done loosely or packed when dpi_color_coding is 18 bpp. This bit enables loosely packing.

Step3: Select the Video Transmission Mode to define how the processor requires the video line to be transported through the DSI link.

Configure low-power transitions (VID_MODE_CFG-[8:3]): This defines the video line to be transported through the DSI link.

Configure low-power transitions (VID_MODE_CFG-[8:3]): This defines the video periods which are permitted to go to low-power if there is available time to do so.

Configure frame_BTA_ack (VID_MODE_CFG-[11]): This specifies if the controller should request the peripheral acknowledge message at the end of frames.

Burst mode: In this mode, the entire active pixel line is buffered into a FIFO and transmitted in a single packed with no interruptions. This transmission mode requires that the DPI Pixel FIFO has the capacity to store a full line of active pixel data inside it. This mode is optimally used if the difference between pixel required bandwidth and DSI link bandwidth is very different. This enables the mipi_dsi_host to quickly dispatch the entire active video line in a single burst of data and then return to low-power mode.
Configure the register fiedl vid_mode_type (VID_MODE_CFG-[10]), num_chunks (VID_PKT_CFG-[20:11]), and null_pkt_size (VID_PKT_CFG-[30:21]) are automatically ignored by the mipi_dsi_host.
Non-Burst mode: In this mode, the processor uses the partitioning properties of the mipi_dsi_host to divide the video line transmission into several DSI packets. This is done to match th pixel required bandwidth with the DSI link bandwidth. With this mode, the controller configuration dose not require a full line of pixel data to be stored inside the DPI Pixel FIFO. It requires only the content of one video packet.
Configure the vid_mode_type field (VID_MODE_CFG-[2:1]) with 2'b0x.
Configure the vid_mode_type field (VID_MODE_CFG-[2:1]) with 2'b00x to enable the transmission of sync pulses.
Configure the vid_mode_type field (VID_MODE_CFG-[2:1]) with 2'b01to enable the transmission of sync events.
Configure the vid_mode_type field (VID_MODE_CFG-[10:0]) with the number of pixels to be transmitted in a single packet.
Configure the en_multi_pkt field (VID_MODE_CFG-[9]) to enable the division of the active video transmission into more than one packet.
Configure the num_chunks field (VID_MODE_CFG-[20:11]) with the number of video chunks that the active video transmission is divided into.
Configure the en_null_pkt field (VID_MODE_CFG-[10]) to enable the insertion of null packets between video packets.
The field is effective only when en_multi_pkt field is activated, otherwise the controller ignores it and does not sent the null packets.
Configure the null_pkt_size field (VID_MODE_CFG-[30:21]) with the actual size of the inserted null packet.

Step4:Define the DPI Horizontal timing configuration as follows:
Configure the hline_time field (TMR_LINE_CFG-[31:18]) with the time taken by a DPI video line accounted in Clock Lane bytes clock cycles (for a clock lane at 500 MHz the Lane byte clock period is 8 ns). When the DPI clock and Clock Lane clock are not multiples, the hline_time is a result of a round of a number. If the mipi_dsi_host is configured to go to low-power, it is possible that the error included in a line is incremented with the next one. At the end of several lines, the mipi_dsi_host can have a number of errors that can cause a malfunction of the video transmission.
Configure the hsa_time field (TMR_LINE_CFG-[8:0]) with the time taken by a DPI Horizontal Sync Active period accounted in Clock Lane byte clock cycles (normally a period of 8ns).
Configure the hbp_time field (TMR_LINE_CFG-[17:9]) with the time taken by a DPI Horizontal Sync Active period accounted in Clock Lane byte clock cycles (normally a period of 8ns). Special attention should be given to the calculation of this parameter.

Step5:Define the Vertical line configuration:
Configure the vsa_lines field (VTIMING_CFG-[3:0]) with the number of lines existing in the DPI Vertical Sync Active period.
Configure the vbp_lines field (VTIMING_CFG-[9:4]) with the number of lines existing in the DPI Vertical Back Porch period.
Configure the vfp_lines field (VTIMING_CFG-[15:10]) with the number of lines existing in the DPI Vertical Front Porch period.
Configure the v_active_lines field (VTIMING_CFG-[26:16]) with the number of lines existing in the DPI Vertical Active period.

Chapter 19 Crypto

19.1 Overview

Crypto is a hardware accelerator of encrypting or decrypting. It supports the most commonly used algorithm: DES/3DES, AES, SHA1, SHA256, MD5 and RSA.
The Crypto supports following features:

- Support AES 128/192/256 bits key mode, ECB/CBC/CTR chain mode, Slave/FIFO mode
- Support DES/3DES (ECB and CBC chain mode) , 3DES (EDE/ EEE key mode), Slave/FIFO mode
- Support SHA1/SHA256/MD5 (with hardware padding) HASH function, FIFO mode only
- Support 160 bit Pseudo Random Number Generator (PRNG)
- Support PKA 512/1024/2048 bit Exp Modulator
- Support up to 150M clock frequency

19.2 Block Diagram

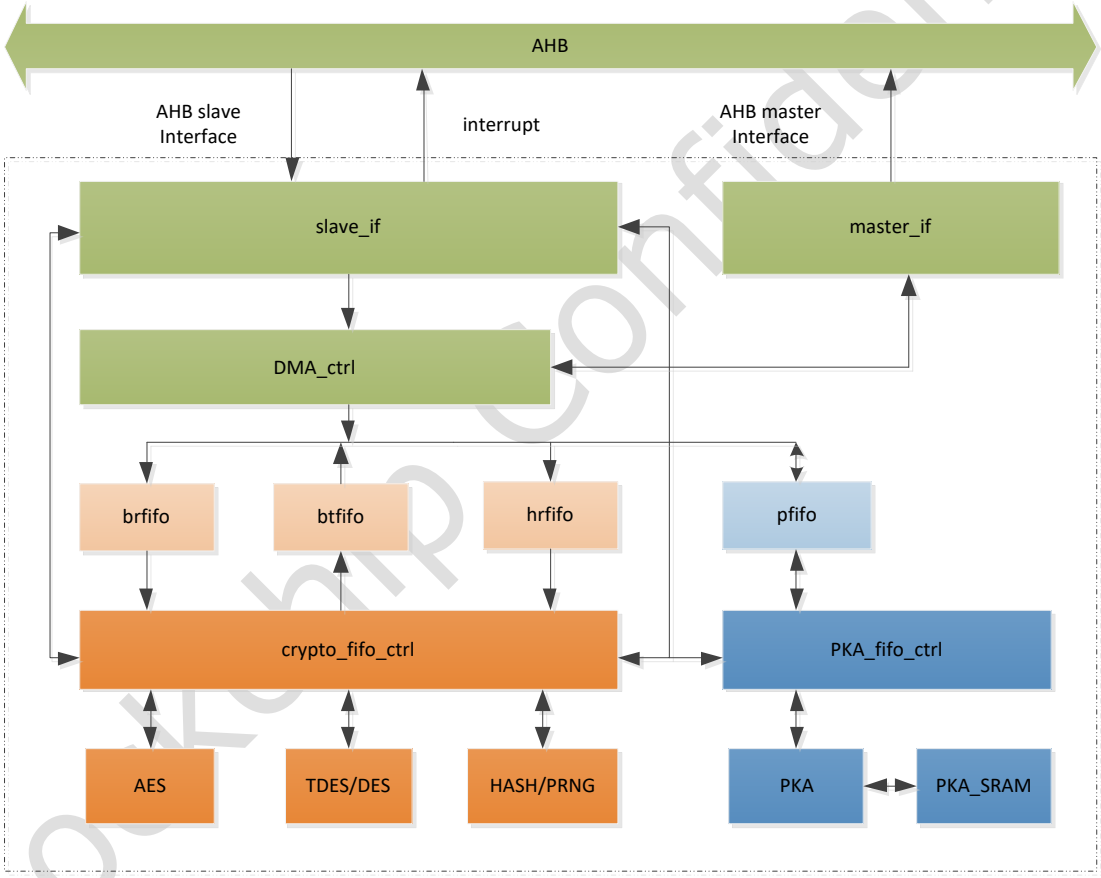


Fig. 19-1 Crypto Architecture

Figure above shows the architecture of Crypto.

19.3 Register description

19.3.1 Register Summary

Name	Offset	Size	Reset Value	Description
Crypto_INTSTS	0x0000	W	0x00000000	Interrupt Status Register
Crypto_INTENA	0x0004	W	0x00000000	Interrupt Set Register
Crypto_CTRL	0x0008	W	0x00000000	Control Register

Name	Offset	Size	Reset Value	Description
Crypto_BRDMAS	0x000c	W	0x00000000	Block Receiving DMA Start Address Register
Crypto_BTDMAS	0x0010	W	0x00000000	Block Transmiting DMA Start Address Register
Crypto_BRDMAL	0x0014	W	0x00000000	Block Receiving DMA Length Register
Crypto_HRDMAS	0x0018	W	0x00000000	Hash Receiving DMA Start Address Register
Crypto_HRDMAL	0x001c	W	0x00000000	Hash Receiving DMA Length Register
Crypto_AES_CTRL	0x0020	W	0x00000000	AES Control Register
Crypto_AES_STS	0x0024	W	0x00000000	Status Register
Crypto_AES_DIN_0	0x0028	W	0x00000000	AES Input Data 0 Register
Crypto_AES_DIN_1	0x002c	W	0x00000000	AES Input Data 1 Register
Crypto_AES_DIN_2	0x0030	W	0x00000000	AES Input Data 2 Register
Crypto_AES_DIN_3	0x0034	W	0x00000000	AES Input Data 3 Register
Crypto_AES_DOUT_0	0x0038	W	0x00000000	AES Output Data 0 Register
Crypto_AES_DOUT_1	0x003c	W	0x00000000	AES Output Data 1 Register
Crypto_AES_DOUT_2	0x0040	W	0x00000000	AES Output Data 2 Register
Crypto_AES_DOUT_3	0x0044	W	0x00000000	AES Output Data 3 Register
Crypto_AES_IV_0	0x0048	W	0x00000000	AES IV data 0 Register
Crypto_AES_IV_1	0x004c	W	0x00000000	AES IV data 1 Register
Crypto_AES_IV_2	0x0050	W	0x00000000	AES IV data 2 Register
Crypto_AES_IV_3	0x0054	W	0x00000000	AES IV data 3 Register
Crypto_AES_KEY_0	0x0058	W	0x00000000	AES Key data 0 Register
Crypto_AES_KEY_1	0x005c	W	0x00000000	AES Key data 1 Register
Crypto_AES_KEY_2	0x0060	W	0x00000000	AES Key data 2 Register
Crypto_AES_KEY_3	0x0064	W	0x00000000	AES Key data 3 Register
Crypto_AES_KEY_4	0x0068	W	0x00000000	AES Key data 4 Register
Crypto_AES_KEY_5	0x006c	W	0x00000000	AES Key data 5 Register
Crypto_AES_KEY_6	0x0070	W	0x00000000	AES Key data 6 Register
Crypto_AES_KEY_7	0x0074	W	0x00000000	AES Key data 7 Register
Crypto_AES_CNT_0	0x0078	W	0x00000000	AES Input Counter 0 Register
Crypto_AES_CNT_1	0x007c	W	0x00000000	AES Input Counter 1 Register
Crypto_AES_CNT_2	0x0080	W	0x00000000	AES Input Counter 2 Register
Crypto_AES_CNT_3	0x0084	W	0x00000000	AES Input Counter 3 Register
Crypto_TDES_CTRL	0x0088	W	0x00000000	TDES Control Register
Crypto_TDES_STS	0x008c	W	0x00000000	Status Register
Crypto_TDES_DIN_0	0x0090	W	0x00000000	TDES Input Data 0 Register
Crypto_TDES_DIN_1	0x0094	W	0x00000000	TDES Input Data 1 Register
Crypto_TDES_DOUT_0	0x0098	W	0x00000000	TDES Output Data 0 Register
Crypto_TDES_DOUT_1	0x009c	W	0x00000000	TDES Output Data 1 Register
Crypto_TDES_IV_0	0x00a0	W	0x00000000	TDES IV data 0 Register

Name	Offset	Size	Reset Value	Description
Crypto_TDES_IV_1	0x00a4	W	0x00000000	TDES IV data 1 Register
Crypto_TDES_KEY1_0	0x00a8	W	0x00000000	TDES Key1 data 1 Register
Crypto_TDES_KEY1_1	0x00ac	W	0x00000000	TDES Key1 data 1 Register
Crypto_TDES_KEY2_0	0x00b0	W	0x00000000	TDES Key2 data 0 Register
Crypto_TDES_KEY2_1	0x00b4	W	0x00000000	TDES Key2 data 1 Register
Crypto_TDES_KEY3_0	0x00b8	W	0x00000000	TDES Key3 data 0 Register
Crypto_TDES_KEY3_1	0x00bc	W	0x00000000	TDES Key3 data 1 Register
Crypto_HASH_CTRL	0x00c0	W	0x00000000	Hash Control Register
Crypto_HASH_STS	0x00c4	W	0x00000000	Hash Status Register
Crypto_HASH_MSG_LEN	0x00c8	W	0x00000000	Hash Message Len
Crypto_HASH_DOUT_0	0x00cc	W	0x00000000	Hash Result Register 0
Crypto_HASH_DOUT_1	0x00d0	W	0x00000000	Hash Result Register 1
Crypto_HASH_DOUT_2	0x00d4	W	0x00000000	Hash Result Register 2
Crypto_HASH_DOUT_3	0x00d8	W	0x00000000	Hash Result Register 3
Crypto_HASH_DOUT_4	0x00dc	W	0x00000000	Hash Result Register 4
Crypto_HASH_DOUT_5	0x00e0	W	0x00000000	Hash Result Register 4
Crypto_HASH_DOUT_6	0x00e4	W	0x00000000	Hash Result Register 6
Crypto_HASH_DOUT_7	0x00e8	W	0x00000000	Hash Result Register 7
Crypto_HASH_SEED_0	0x00ec	W	0x00000000	PRNG Seed/HMAC Key Register 0
Crypto_HASH_SEED_1	0x00f0	W	0x00000000	PRNG Seed/HMAC Key Register 1
Crypto_HASH_SEED_2	0x00f4	W	0x00000000	PRNG Seed/HMAC Key Register 2
Crypto_HASH_SEED_3	0x00f8	W	0x00000000	PRNG Seed/HMAC Key Register 3
Crypto_HASH_SEED_4	0x00fc	W	0x00000000	PRNG Seed/HMAC Key Register 4
Crypto_PKA_CTRL	0x0100	W	0x00000000	PKA Control Register
Crypto_PKA_STS	0x0104	W	0x00000000	PKA Status Register
Crypto_PKA_M	0x0400	W	0x00000000	
Crypto_PKA_C	0x0500	W	0x00000000	
Crypto_PKA_N	0x0600	W	0x00000000	
Crypto_PKA_E	0x0700	W	0x00000000	

*Notes:Size:**B**- Byte (8 bits) access, **HW**- Half WORD (16 bits) access, **W**-WORD (32 bits) access*

19.3.2 Detail Register Description

Crypto_INTSTS

Address: Operational Base + offset (0x0000)
Interrupt Status Register

Bit	Attr	Reset Value	Description
31:6	RO	0x0	reserved
5	RW	0x0	PKA_DONE_INT PKA Done Interrupt
4	W1C	0x0	HASH_DONE_INT Hash Done Interrupt
3	W1C	0x0	HRDMA_ERR_INT Specifies the interrupt of hash receiving DMA Error

Bit	Attr	Reset Value	Description
2	W1C	0x0	HRDMA_DONE_INT Specifies the interrupt of hash receiving DMA DONE
1	W1C	0x0	BCDMA_ERR_INT Specifies the interrupt of block cipher Error
0	W1C	0x0	BCDMA_DONE_INT Specifies the interrupt of block cipher DONE

Crypto_INTENA
Address: Operational Base + offset (0x0004)
Interrupt Set Register

Bit	Attr	Reset Value	Description
31:6	RO	0x0	reserved
5	RW	0x0	PKA_DONE_ENA Set the interrupt Enable of PKA done 1'b1: enable 1'b0: disable
4	RW	0x0	HASH_DONE_ENA Set the interrupt Enable of hash done 1'b1: enable 1'b0: disable
3	RW	0x0	HRDMA_ERR_ENA Set the interrupt Enable of hash receiving DMA Error 1'b1: enable 1'b0: disable
2	RW	0x0	HRDMA_DONE_ENA Set the interrupt Enable of hash receiving DMA DONE 1'b1: enable 1'b0: disable
1	RW	0x0	BCDMA_ERR_ENA Set the interrupt Enable of block cipher DMA Error 1'b1: enable 1'b0: disable
0	RW	0x0	BCDMA_DONE_ENA Set the interrupt Enable of block cipher DMA DONE 1'b1: enable 1'b0: disable

Crypto_CTRL
Address: Operational Base + offset (0x0008)
Control Register

Bit	Attr	Reset Value	Description
31:17	RO	0x0	reserved

Bit	Attr	Reset Value	Description
16	RW	0x0	HR_ADDR_MODE Hash Receive DMA Address Mode 1'b1: fix 1'b0: increment
15	RW	0x0	BT_ADDR_MODE Block Transmit DMA Address Mode 1'b1: fix 1'b0: increment
14	RW	0x0	BR_ADDR_MODE Block Receive DMA Address Mode 1'b1: fix 1'b0: increment
13	RW	0x0	Byteswap_HRFIFO If this bit is high, then the data read from the bus is byte-swapped in a word boundary. If this bit is low (default), then the data is handed over to the FIFO without byte-swap. For little endian bus, this bit should be 1'b1.
12	RW	0x0	Byteswap_BTFIFO If this bit is high, then the data read from the bus is byte-swapped in a word boundary. If this bit is low (default), then the data is handed over to the FIFO without byte-swap. For little endian bus, this bit should be 1'b1.
11	RW	0x0	Byteswap_BRFIFO If this bit is high, then the data read from the bus is byte-swapped in a word boundary. If this bit is low (default), then the data is handed over to the FIFO without byte-swap. For little endian bus, this bit should be 1'b1.
10	RW	0x0	DESSEL Specifies the Destination block cipher of FIFO. 1'b0:AES; 1'b1:DES.
9:8	RW	0x0	HASHINSEL Specifies the following 2'b00:Data from independent source; 2'b01:Data from block cipher input; 2'b10:Data from block cipher output; 2'b11:Reserved.
7	R/WSC	0x0	PKA_FLUSH Software write 1 to start Flush Process.The process will clear BRFIFO, BTFIFO, and state machine. Then Software should write 0 to end FLUSH Process
6	RW	0x0	HASH_FLUSH Software write 1 to start Flush Process. The process will clear BRFIFO, BTFIFO, and state machine. Then Software should write 0 to end FLUSH Process

Bit	Attr	Reset Value	Description
5	RW	0x0	BLOCK_FLUSH Software write 1 to start Flush Process. The process will clear BRFIFO, BTFIFO, and state machine. Then Software should write 0 to end FLUSH Process. It must last for at least 20 cycles to clean registers and FSM
4	R/W SC	0x0	PKA_START Starts/initializes PKA Software write 1 to start. When finishes, the core will clear it.
3	R/W SC	0x0	HASH_START Starts/initializes HASH/PRNG/HMAC Software write 1 to start. When finishes, the core will clear it.
2	R/W SC	0x0	BLOCK_START Starts/initializes Block Cipher Software write 1 to start. When finishes, the core will clear it.
1	R/W SC	0x0	TDES_START Starts/initializes TDES Software write 1 to start. When finishes, the core will clear it.
0	R/W SC	0x0	AES_START Starts/initializes AES Software write 1 to start. When finishes, the core will clear it.

Crypto_BRDMAS
Address: Operational Base + offset (0x000c)
Block Receiving DMA Start Address Register

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	STARTADDR Specifies the Start Address of DMA The address should be aligned by 32-bit.

Crypto_BTDMAS
Address: Operational Base + offset (0x0010)
Block Transmiting DMA Start Address Register

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	STARTADDR Specifies the Start Address of DMA The address needs to be aligned by 32-bit.

Crypto_BRDMAL
Address: Operational Base + offset (0x0014)
Block Receiving DMA Length Register

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	LENGTH Specifies the Block length of DMA. The length unit is WORD.

Crypto_HRDMAS
Address: Operational Base + offset (0x0018)
Hash Receiving DMA Start Address Register

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	STARTADDR Specifies the Start Address of DMA The address needs to be aligned by 32-bit.

Crypto_HRDMAL
Address: Operational Base + offset (0x001c)
Hash Receiving DMA Length Register

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	LENGTH Specifies the Block length of DMA. The length unit is BYTE.

Crypto_AES_CTRL
Address: Operational Base + offset (0x0020)
AES Control Register

Bit	Attr	Reset Value	Description
31:12	RO	0x0	reserved
11	RW	0x0	AES_ByteSwap_CNT Change the Big-endian and Little-endian by swapping the byte oder. 0 = Disables Counter data byte swap 1 = Enables Counter data byte swap
10	RW	0x0	AES_ByteSwap_Key Change the Big-endian and Little-endian by swapping the byte oder. 0 = Disables Key byte swap 1 = Enables Key byte swap
9	RW	0x0	AES_ByteSwap_IV Change the Big-endian and Little-endian by swapping the byte oder. 0 = Disables Initial value byte swap 1 = Enables Initial value byte swap
8	RW	0x0	AES_ByteSwap_DO Change the Big-endian and Little-endian by swapping the byte oder. 0 = Disables Output data byte swap 1 = Enables Output data byte swap
7	RW	0x0	AES_ByteSwap_DI Change the Big-endian and Little-endian by swapping the byte oder. 0 = Disables Input data byte swap 1 = Enables Input data byte swap

Bit	Attr	Reset Value	Description
6	RW	0x0	AES_KeyChange Specifies the AES key change mode selection signal. When the bit is asserted, it will not do key-expansion function to calculate new sub-key. So it is a faster way, when several times of calculation use the same key. But if the keys are different, asserting this bit will have the wrong result. 0 = Key is not changed 1 = Key is changed
5:4	RW	0x0	AES_ChainMode Specifies AES chain mode selection 00 = ECB mode 01 = CBC mode 10 = CTR mode
3:2	RW	0x0	AES_KeySize Specifies the AES key size selection signal 00 : 128-bit key 01 : 192-bit key 10 : 256-bit key
1	RW	0x0	AES_FifoMode Specify AES Fifo Mode 1'b0: Slave mode 1'b1: fifo mode
0	RW	0x0	AES_Enc Specifies the Encryption/ Decryption mode selection signal 0 : Encryption 1 : Decryption

Crypto_AES_STS
Address: Operational Base + offset (0x0024)
Status Register

Bit	Attr	Reset Value	Description
31:1	RO	0x0	reserved
0	RW	0x0	AES_DONE When AES finish, it will be HIGH, And it will not be LOW until it restart . 1: done 0: not done

Crypto_AES_DIN_0
Address: Operational Base + offset (0x0028)
AES Input Data 0 Register

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	AES_DIN_0 Specifies AES Input data [127:96].

Crypto_AES_DIN_1
Address: Operational Base + offset (0x002c)
AES Input Data 1 Register

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	AES_DIN_1 Specifies AES Input data [95:64].

Crypto_AES_DIN_2
Address: Operational Base + offset (0x0030)
AES Input Data 2 Register

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	AES_DIN_2 Specifies AES Input data [63:32]

Crypto_AES_DIN_3
Address: Operational Base + offset (0x0034)
AES Input Data 3 Register

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	AES_DIN_3 Specifies AES Input data [31:0]

Crypto_AES_DOUT_0
Address: Operational Base + offset (0x0038)
AES Output Data 0 Register

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	AES_DOUT_0 Specifies AES Output data [127:96].

Crypto_AES_DOUT_1
Address: Operational Base + offset (0x003c)
AES Output Data 1 Register

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	AES_DOUT_1 Specifies the Output data [95:64].

Crypto_AES_DOUT_2
Address: Operational Base + offset (0x0040)
AES Output Data 2 Register

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	AES_DOUT_2 Specifies AES Output data [63:32].

Crypto_AES_DOUT_3
Address: Operational Base + offset (0x0044)
AES Output Data 3 Register

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	AES_DOUT_3 Specifies AES Output data [31:0].

Crypto_AES_IV_0
Address: Operational Base + offset (0x0048)
AES IV data 0 Register

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	AES_IV_0 Specifies AES Initialization vector [127:96]

Crypto_AES_IV_1
Address: Operational Base + offset (0x004c)
AES IV data 1 Register

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	AES_IV_1 Specifies AES Initialization vector [95:64]

Crypto_AES_IV_2
Address: Operational Base + offset (0x0050)
AES IV data 2 Register

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	AES_IV_2 Specifies AES Initialization vector [63:32]

Crypto_AES_IV_3
Address: Operational Base + offset (0x0054)
AES IV data 3 Register

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	AES_IV_3 Specifies AES Initialization vector [31:0]

Crypto_AES_KEY_0
Address: Operational Base + offset (0x0058)
AES Key data 0 Register

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	AES_KEY_0 Specifies AES key data [255:224]

Crypto_AES_KEY_1
Address: Operational Base + offset (0x005c)
AES Key data 1 Register

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	AES_KEY_1 Specifies AES key data [223:192]

Crypto_AES_KEY_2
Address: Operational Base + offset (0x0060)
AES Key data 2 Register

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	AES_KEY_2 Specifies AES key data [191:160]

Crypto_AES_KEY_3
Address: Operational Base + offset (0x0064)
AES Key data 3 Register

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	AES_KEY_3 Specifies AES key data [159:128]

Crypto_AES_KEY_4
Address: Operational Base + offset (0x0068)
AES Key data 4 Register

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	AES_KEY_4 Specifies AES key data [127:96]

Crypto_AES_KEY_5
Address: Operational Base + offset (0x006c)
AES Key data 5 Register

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	AES_KEY_5 Specifies the key data [95:64]

Crypto_AES_KEY_6
Address: Operational Base + offset (0x0070)
AES Key data 6 Register

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	AES_KEY_6 Specifies AES key data [63:32]

Crypto_AES_KEY_7
Address: Operational Base + offset (0x0074)
AES Key data 7 Register

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	AES_KEY_7 Specifies the key data [31:0]

Crypto_AES_CNT_0
Address: Operational Base + offset (0x0078)
AES Input Counter 0 Register

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	AES_CNT_0 Specifies AES Input Counter [127:96].

Crypto_AES_CNT_1
Address: Operational Base + offset (0x007c)
AES Input Counter 1 Register

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	AES_CNT_1 Specifies AES Input Counter [95:64].

Crypto_AES_CNT_2
Address: Operational Base + offset (0x0080)
AES Input Counter 2 Register

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	AES_CNT_2 Specifies AES Input Counter[63:32]

Crypto_AES_CNT_3
Address: Operational Base + offset (0x0084)
AES Input Counter 3 Register

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	AES_CNT_3 Specifies AES Input Counter [31:0]

Crypto_TDES_CTRL
Address: Operational Base + offset (0x0088)
TDES Control Register

Bit	Attr	Reset Value	Description
31:9	RO	0x0	reserved
8	RW	0x0	TDES_ByteSwap_Key 0 = Disables Key byte swap 1 = Enables Key byte swap
7	RW	0x0	TDES_ByteSwap_IV 0 = Disables Initial value byte swap 1 = Enables Initial value byte swap
6	RW	0x0	TDES_ByteSwap_DO 0 = Disables Output data byte swap 1 = Enables Output data byte swap
5	RW	0x0	TDES_ByteSwap_DI 0 = Disables Input data byte swap 1 = Enables Input data byte swap
4	RW	0x0	TDES_ChainMode Specifies TDES chain mode selection 0 : ECB mode 1 : CBC mode

Bit	Attr	Reset Value	Description
3	RW	0x0	TDES_EEE Specifies the TDES key mode selection 1'b0 : EDE 1'b1 : EEE
2	RW	0x0	TDES_Select Specify DES or TDES cipher 1'b0 : DES 1'b1 : TDES
1	RW	0x0	TDES_FifoMode Specify TDES Fifo Mode 1'b0: Slave mode 1'b1: Fifo mode
0	RW	0x0	TDES_Enc Specifies the Encryption/ Decryption mode selection signal 0 : Encryption 1 : Decryption

Crypto_TDES_STS
Address: Operational Base + offset (0x008c)
Status Register

Bit	Attr	Reset Value	Description
31:1	RO	0x0	reserved
0	RW	0x0	TDES_DONE When DES/TDES finishes, it will be HIGH, And it will not be LOW until it restart . 1: done 0: not done

Crypto_TDES_DIN_0
Address: Operational Base + offset (0x0090)
TDES Input Data 0 Register

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	TDES_DIN_0 Specifies TDES Input data [63:32].

Crypto_TDES_DIN_1
Address: Operational Base + offset (0x0094)
TDES Input Data 1 Register

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	TDES_DIN_1 Specifies TDES Input data [31:0].

Crypto_TDES_DOUT_0
Address: Operational Base + offset (0x0098)
TDES Output Data 0 Register

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	TDES_DOUT_0 Specifies TDES Output data [63:32].

Crypto_TDES_DOUT_1
Address: Operational Base + offset (0x009c)
TDES Output Data 1 Register

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	TDES_DOUT_1 Specifies TDES Output data [31:0].

Crypto_TDES_IV_0
Address: Operational Base + offset (0x00a0)
TDES IV data 0 Register

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	TDES_IV_0 Specifies TDES Initialization vector [63:32]

Crypto_TDES_IV_1
Address: Operational Base + offset (0x00a4)
TDES IV data 1 Register

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	TDES_IV_1 Specifies TDES Initialization vector [31:0]

Crypto_TDES_KEY1_0
Address: Operational Base + offset (0x00a8)
TDES Key1 data 1 Register

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	TDES_KEY1_0 Specifies TDES key1 data [63:32]

Crypto_TDES_KEY1_1
Address: Operational Base + offset (0x00ac)
TDES Key1 data 1 Register

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	TDES_KEY1_1 Specifies TDES key1 data [31:0]

Crypto_TDES_KEY2_0
Address: Operational Base + offset (0x00b0)
TDES Key2 data 0 Register

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	TDES_KEY2_0 Specifies TDES key2 data [63:32]

Crypto_TDES_KEY2_1
Address: Operational Base + offset (0x00b4)
TDES Key2 data 1 Register

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	TDES_KEY_1 Specifies TDES key data [31:0]

Crypto_TDES_KEY3_0
Address: Operational Base + offset (0x00b8)
TDES Key3 data 0 Register

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	TDES_KEY3_0 Specifies TDES key3 data [63:32]

Crypto_TDES_KEY3_1
Address: Operational Base + offset (0x00bc)
TDES Key3 data 1 Register

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	AES_KEY3_1 Specifies TDES key3 data [31:0]

Crypto_HASH_CTRL
Address: Operational Base + offset (0x00c0)
Hash Control Register

Bit	Attr	Reset Value	Description
31:4	RO	0x0	reserved
3	RW	0x0	HASH_SWAP_DO Specifies the Byte swap of data output (hash result) 0 = Does not swap (default) 1 = Swap
2	RW	0x0	HASH_SWAP_DI Specifies the Byte swap of data input. 0 = Does not swap (default) 1 = Swap
1:0	RW	0x0	Engine_Selection 2'b00: SHA1_HASH 2'b01: MD5_HASH 2'b10: SHA256_HASH 2'b11: PRNG

Crypto_HASH_STS
Address: Operational Base + offset (0x00c4)
Hash Status Register

Bit	Attr	Reset Value	Description
31:1	RO	0x0	reserved

Bit	Attr	Reset Value	Description
0	RW	0x0	HASH_DONE Hash Done Signal When HASH finishes, it will be HIGH, And it will not be LOW until it restart 1'b1 : done 1'b0 : not done

Crypto_HASH_MSG_LEN
Address: Operational Base + offset (0x00c8)
Hash Message Len

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	Msg_size Hash total byte.

Crypto_HASH_DOUT_0
Address: Operational Base + offset (0x00cc)
Hash Result Register 0

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	HASH_RESULT_0 Specifies the HASH Result [159:128]

Crypto_HASH_DOUT_1
Address: Operational Base + offset (0x00d0)
Hash Result Register 1

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	HASH_RESULT_1 Specifies the HASH Result [127:96]

Crypto_HASH_DOUT_2
Address: Operational Base + offset (0x00d4)
Hash Result Register 2

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	HASH_RESULT_2 Specifies the HASH Result [95:64]

Crypto_HASH_DOUT_3
Address: Operational Base + offset (0x00d8)
Hash Result Register 3

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	HASH_RESULT_3 Specifies the HASH Result [63:32]

Crypto_HASH_DOUT_4
Address: Operational Base + offset (0x00dc)
Hash Result Register 4

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	HASH_RESULT_4 Specifies the HASH Result [31:0]

Crypto_HASH_DOUT_5
Address: Operational Base + offset (0x00e0)
Hash Result Register 4

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	HASH_RESULT_5 Specifies the HASH Result [31:0]

Crypto_HASH_DOUT_6
Address: Operational Base + offset (0x00e4)
Hash Result Register 6

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	HASH_RESULT_6 Specifies the HASH Result [31:0]

Crypto_HASH_DOUT_7
Address: Operational Base + offset (0x00e8)
Hash Result Register 7

Bit	Attr	Reset Value	Description
31:0	RO	0x00000000	HASH_RESULT_7 Specifies the HASH Result [31:0]

Crypto_HASH_SEED_0
Address: Operational Base + offset (0x00ec)
PRNG Seed/HMAC Key Register 0

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	HASH_SEED_0 Specifies PRNG Seed/HMAC Key buffer [159:128]

Crypto_HASH_SEED_1
Address: Operational Base + offset (0x00f0)
PRNG Seed/HMAC Key Register 1

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	HASH_SEED_1 Specifies PRNG Seed/HMAC Key buffer [127:96]

Crypto_HASH_SEED_2
Address: Operational Base + offset (0x00f4)
PRNG Seed/HMAC Key Register 2

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	HASH_SEED_2 Specifies PRNG Seed/HMAC Key buffer [95:64]

Crypto_HASH_SEED_3
Address: Operational Base + offset (0x00f8)
PRNG Seed/HMAC Key Register 3

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	HASH_SEED_3 Specifies PRNG Seed/HMAC Key buffer [63:32]

Crypto_HASH_SEED_4
Address: Operational Base + offset (0x00fc)
PRNG Seed/HMAC Key Register 4

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	HASH_SEED_4 Specifies PRNG Seed/HMAC Key buffer [31:0]

Crypto_PKA_CTRL
Address: Operational Base + offset (0x0100)
PKA Control Register

Bit	Attr	Reset Value	Description
31:2	RO	0x0	reserved
1:0	RW	0x0	block_size PKA Size It specifies the bits of N in PKA calculation. 2'b00: 512 bit 2'b01: 1024 bit 2'b10: 2048 bit

Crypto_PKA_STS
Address: Operational Base + offset (0x0104)
PKA Status Register

Bit	Attr	Reset Value	Description
31:1	RO	0x0	reserved
0	RW	0x0	PKA_DONE PKA done

Crypto_PKA_M
Address: Operational Base + offset (0x0400)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	m PKA input or output data. PKA result = (M ^ E) mod N. When it finishes, the result data is in M position. Start from PKA_M base address, and may contain 512/1024/2048 bits data.

Crypto_PKA_C

Address: Operational Base + offset (0x0500)

Bit	Attr	Reset Value	Description
31:1	RO	0x0	reserved
0	RW	0x0	c PKA pre-calculate data, C = 2 ^(2n+2) mod N

Crypto_PKA_N
Address: Operational Base + offset (0x0600)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	n PKA modular

Crypto_PKA_E
Address: Operational Base + offset (0x0700)

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	e PKA exponent.

19.4 Application Note

19.4.1 Reset a port

CRU. crypto_srstn_req is used to do a soft reset to crypto . Please refer to "Chapter CRU" for more details.

19.4.2 Overall Performance

Use CRU.crypto_div_con to select crypto frequency: $F_{crypto} = F_{aclk} / (div + 1)$.
Make sure F_{crypto} do not exceed 150M.
The performance of crypto FIFO mode is list below.

Table 19-1 Crypto Performance Description

algorithm	cycle	block size	frequency	throughput rate
DES	17	64 bit	100M	<=376 M bps
TDES	51	64 bit	100M	<=125 M bps
AES	11/13/15	128 bit	100M	<=1160/984/853Mbps
SHA-1	81	512 bit	100M	<=632 Mbps
MD5	65	512 bit	100M	<=787 Mbps

19.4.3 Usage

1. Symmetric algorithm

DES/3DES, AES are symmetric algorithms. There are two ways of using these algorithms: Slave mode and FIFO mode.
In Slave mode, you can calculate 1 block size of data by starting the engine. Take AES-128 for example, you should

- Program Input 128 bit Data to AES_DIN_0~AES_DIN_3
- Program Input 128 bit Key to AES_KEY_0~AES_KEY_3
- Program control mode to AES_CTRL to run in different mode
- Program CTRL.AES_START to run
- wait AES_STS.DONE High

- Read AES_DOUT_0 ~ AES_DOUT_3 to get result.

In FIFO mode,

- Program the source address to BRDMAS, the destination address to BTDMAS, program the length in word unit to BRDMAL;
- Program Input 128 bit Key to AES_KEY_0~AES_KEY_3;
- Program control mode to AES_CTRL to run in different mode;
- Program INTENA to enable interrupt;
- Program CTRL.BLOCK_START to start;
- wait interrupt asserted;
- Program INTSTS to clear interrupt status;
- Read the destination address which BTDMA points to.

FIFO mode get much higher throughput rate.

2. HASH

HASH is used to get digest of data. Only support FIFO mode.
There are three source: (1) hr_fifo; (2) br_fifo; (3) bt_fifo.
Take hr_fifo for example
Program CTRL.HASH_FLUSH 1'b1 to clear, wait several cycle (>= 10 cycles) , and Program CTRL.HASH_FLUSH 1'b0
Program data source address to HRDMAS, program 1 time data length in word unit to HRDMAL, program total length in byte unit to HASH_MSG_LEN
Program HASH_CTRL to choose algorithm, for example SHA-256
Program INTENA to enable interrupt;
Program CTRL.HASH_START 1'b1 to start;
Wait interrupt asserted; Only if HRDMAL length meets can this interrupt be asserted
If you have another section of data to hash, then go to (2), HASH_MSG_LEN need not to be programmed;
else go to (8)
wait HASH_STS.done asserted. Only if Hash_MSG_LEN meet can this bit status register asserted.
Read HASH_DOUT_0 – HASH_DOUT_7 to get result.

3. Asymmetric Algorithm

Support 512/1024/2048 bit RSA calculation. It provide the big number calculation. Result = ME mod N
Program CTRL.PKA_FLUSH 1'b1 to flush RSA module;
Wait CTRL.PKA_FLUSH to be LOW. It is self-cleared;
Program input_data(M) to PKA_M; Program pre_caculated C to PKA_C; Program Key(N) to PKA_N; Program Key(E) to PKA_E. C = 2 (2n+2) mod N. n is the required bit of N. For example 2048 bit N, n = 2048;
Program PKA_CTRL to select RSA size: 512/1024/2048
Program INTENA to enable interrupt;
Program CTRL.PKA_START to start;
Wait interrupt asserted.
Read PKA_M to get results.

Chapter 20 Global Positioning System (GPS)

20.1 Overview

The GPS is a high-performance baseband device which has an AHB master interface and an AHB slave interface.
CPU can access GPS registers through the AHB slave interface.
The GPS has a 32-channel DMA inside, which can read/write data to system memory through the AHB master interface.
The GPS supports following features:

- Single chip, integrate GPS baseband with CPU
- 32 DMA channels for AHB master access
- Complete l1-band, C/A, and NMEA-0183 compatibility
- Support reference frequency 16.368MHz
- High sensitivity for indoor fixes
- Low power consumption
- Low cost with smaller size
- Multi modes support both standalone GPS and A_GPS

20.2 Block Diagram

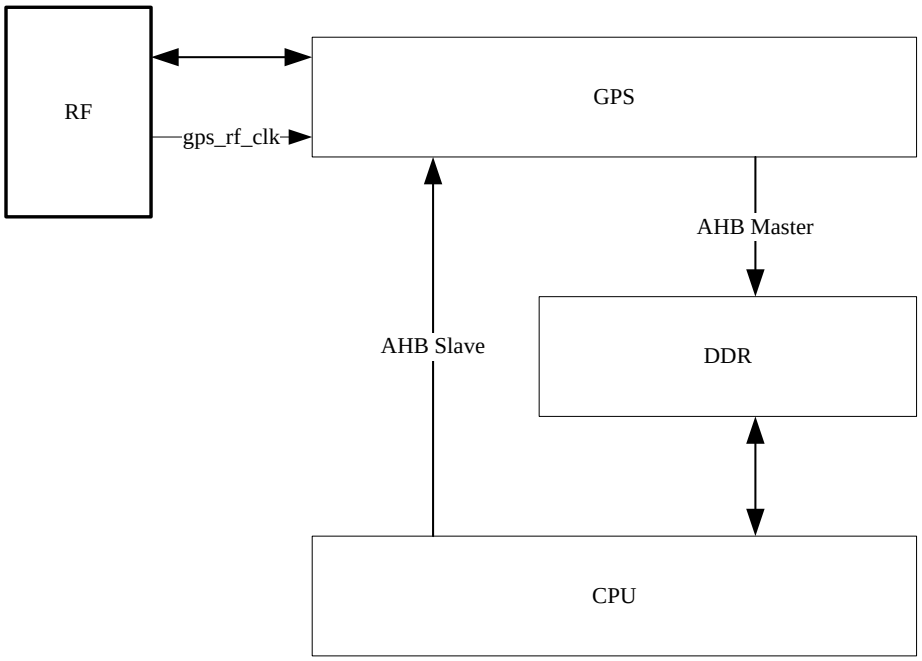


Fig. 20-1 GPS Block Diagram

the GPS controller should be connected to RF chip through GPIO. The RF chip has 3 1-bit signals output to GPS controller inside chip. They are GPS_RF_CLK, GPS_SIG, GPS_MAG. The CPU can start RF chip through GPIO, after GPS configuring completion.

20.3 Register Description

This section describes the control/status registers of the design. The base address of GPS is 0x10400000.

20.3.1 Base band Registers Summary

Name	Offset	Size	Reset Value	Description
BB_CTRL	0x400	W	0x0	base band control register
BB_START_ADR	0x404	W	0x0	correlator start address register
BB_DS_PARAMETER	0x408	W	0x0	down-sample control register
GPS_INT_ENA	0x40c	W	0x0	interrupt enable register
GPS_INT_STATUS	0x410	W	0x0	interrupt status register
BB_CHN_COR_STATUS	0x414	W	0x0	channel correlator status register

Name	Offset	Size	Reset Value	Description
BB_CHN_COR_VALID	0x418	W	0x0	channel correlator valid register
BB_RF_TIMER_VAL	0x41c	W	0x0	RF clock timer value register
BB_RF_WT_ADDR	0x420	W	0x1fffffff	RF FIFO write address register
CH_MISCx	x<<5+0	W	N/A	control information
CH_CAR_NCOx	x<<5+4	W	N/A	carrier NCO
CH_CODE_NCOx	x<<5+8	W	N/A	code NCO
CH_DMA_ADDRx	x<<5+12	W	N/A	dma address
CH_CAR_FREQx	x<<5+16	W	N/A	carrier frequency
CH_CODE_FREQx	x<<5+20	W	N/A	carrier frequency

*Notes:Size:**B**- Byte (8 bits) access, **HW**- Half WORD (16 bits) access, **W**-WORD (32 bits) access, x is for the channel number from 0-31*

20.3.2 ACC operation register summary

Name	Offset	Size	Reset Value	Description
ACC_CTRL	0x8000	W	0x0	acc control register
DMA_CH0_START_ADDR	0x8004	W	0xffffffff	channel 0 dma start address
DMA_CH1_START_ADDR	0x800c	W	0xaaaaaaaa	channel 1 dma start address
DMA_CH2_START_ADDR	0x8014	W	0x55555555	channel 2 dma start address
FFT_VLD_NUM_START_POINT	0x801c	W	0x0	FFT_START_POINT, FFT_VALID_NUM
FFT_COR_NUM_LOOP_NUM	0x8020	W	0x0	FFT_LOOP_NUM, TOT_EPD_COR_NUM, EPD_START_COR_NUM
EPD_MAXCORID	0x8024	W	0x0	EPD_MAXCORID, MAX_FREQINDEX
EPD_MAX_DATA_MAN	0x8028	W	0x0	the maximum energy mantissa
EPD_MAX_DATA_EXP	0x802c	W	0x0	the exponent of the maximum energy
EPD_MAX_RIGHT_MAN	0x8030	W	0x0	the mantissa of the right data to the max energy
EPD_MAX_RIGHT_EXP	0x8034	W	0x0	the exponent of the right data to the max energy
EPD_MAX_LEFT_MAN	0x8038	W	0x0	the mantissa of the left data to the max energy
EPD_MAX_LEFT_EXP	0x803c	W	0x0	the exponent of the left data to the max energy
EPD_L0_DATA_MAN	0x8040	W	0x0	the mantissa of the max energy of the cor0 data
EPD_L0_DATA_EXP	0x8044	W	0x0	the exponent of the max energy of the cor0 data
EPD_L1_DATA_MAN	0x8048	W	0x0	the mantissa of the max energy of the cor1 data
EPD_L1_DATA_EXP	0x804c	W	0x0	the exponent of the max energy of the cor1 data
EPD_L2_DATA_MAN	0x8050	W	0x0	the mantissa of the max energy of the cor2 data
EPD_L2_DATA_EXP	0x8054	W	0x0	the exponent of the max energy of the cor2 data
EPD_L3_DATA_MAN	0x8058	W	0x0	the mantissa of the max energy of the cor3 data
EPD_L3_DATA_EXP	0x805c	W	0x0	the exponent of the max energy of the cor3 data
EPD_L4_DATA_MAN	0x8060	W	0x0	the mantissa of the max energy of the cor4 data

Name	Offset	Size	Reset Value	Description
EPD_L4_DATA_EXP	0x8064	W	0x0	the exponent of the max energy of the cor4 data
EPD_L5_DATA_MAN	0x8068	W	0x0	the mantissa of the max energy of the cor5 data
EPD_L5_DATA_EXP	0x806c	W	0x0	the exponent of the max energy of the cor5 data
EPD_L6_DATA_MAN	0x8070	W	0x0	the mantissa of the max energy of the cor6 data
EPD_L6_DATA_EXP	0x8074	W	0x0	the exponent of the max energy of the cor6 data
EPD_L7_DATA_MAN	0x8078	W	0x0	the mantissa of the max energy of the cor7 data
EPD_L7_DATA_EXP	0x807c	W	0x0	the exponent of the max energy of the cor7 data
EPD_L8_DATA_MAN	0x8080	W	0x0	the mantissa of the max energy of the cor8 data
EPD_L8_DATA_EXP	0x8084	W	0x0	the exponent of the max energy of the cor8 data
EPD_L9_DATA_MAN	0x8088	W	0x0	the mantissa of the max energy of the cor9 data
EPD_L9_DATA_EXP	0x808c	W	0x0	the exponent of the max energy of the cor9 data

*Notes:Size:**B**- Byte (8 bits) access, **HW**- Half WORD (16 bits) access, **W**-WORD (32 bits) access*

20.4 Interface Description

The IOMUX configuration of GPS is shown below:

Table 20-1 GPS IOMUX Setting

Module Pin	IO	Pad Name	IOMUX Setting
gps_rfclk	I	IO_LCD0d22_EBCgdpwr1_GPSclk_GMACcol_GPIO2d0	GPIO2D_IOMUX[14:12]=3'b011
gps_mag	I	IO_LCD0d21_EBCborder1_GPSmag_GMACtxd3_GPIO2c7	GPIO2C_IOMUX2[14:12]=3'b011
gps_sig	I	IO_LCD0d20_EBCborder0_GPSsign_GMACtxd2_GPIO2c6	GPIO2C_IOMUX2[10:8]=3'b011

Notes: 1. I=input, O=output, I/O=input/output, bidirectional

20.5 Application Notes

GPS has 2 interrupt output signals: gps_irq and gps_timer_irq. The gps_irq interrupt has a system interrupt ID 44, and gps_timer_irq has a system interrupt ID 45.

