福昕PDF编辑器

• 永久　• 轻巧　• 自由

升级会员　　批量购买

永久使用

无限制使用次数

极速轻巧

超低资源占用，告别卡顿慢

自由编辑

享受Word一样的编辑自由

扫一扫，关注公众号

http://editor.foxitsoftware.cn

Chapter 18 Video Input Processor (VIP)

18.1 Overview

The Video Input Processor, receives the data from Camera or CCIR656 encoder, and transfers the data into system main memory by AXI bus.

The Video Input Processor supports following features:

Support YCbCr422 input
Support Raw8bit input
Support CCIR656(PAL/NTSC) input
Support JPEG input
Support YCbCr422/420 output
Support UYVY/VYUY/YUYV/YVYU configurable
Support up to 8192x8192 resolution source
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

18.2 Block Diagram

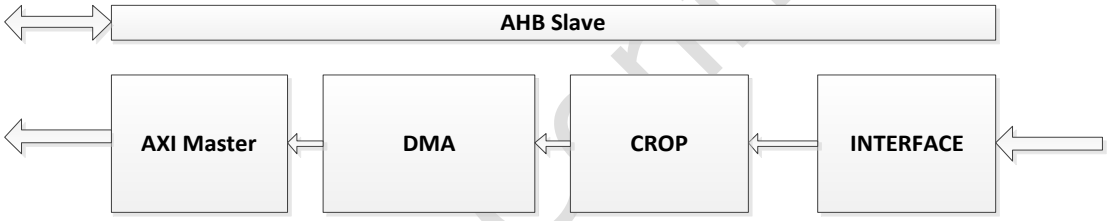

Fig. 18-1 VIP block diagram

The VIP 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

18.3 Function description

This chapter is used to illustrate the operational behavior of how VIP works. If YUV422 or ccir656 signal is received from external devices, VIP 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.

18.3.1 Support Vsync high active or low active

Vsync Low active as below

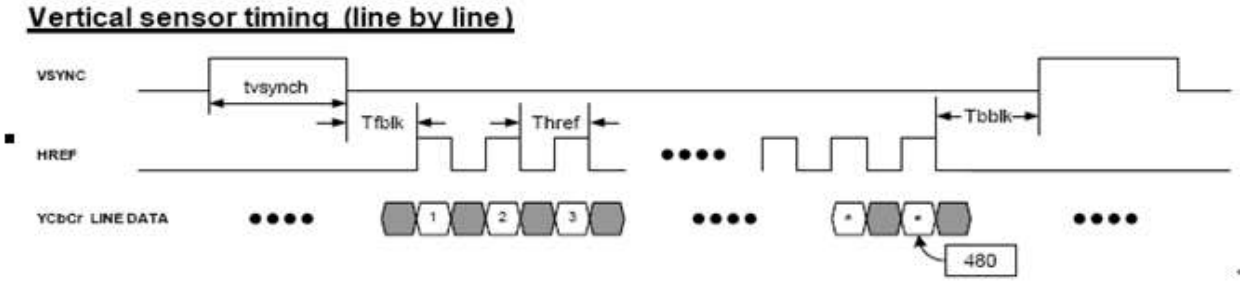

Fig. 18-2 Timing diagram for VIP when vsync low active

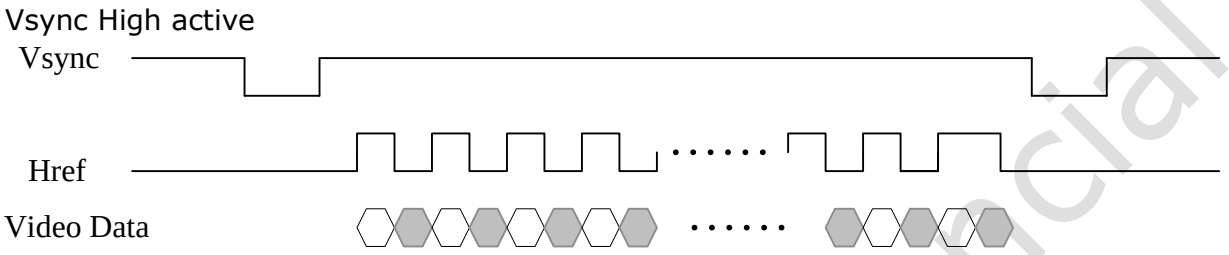

Fig. 18-3 Timing diagram for VIP when vsync high active

18.3.2 Support href high active or low active

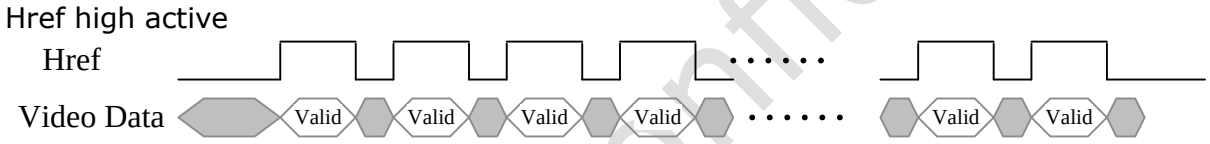

Fig. 18-4 Timing diagram for VIP when href high active

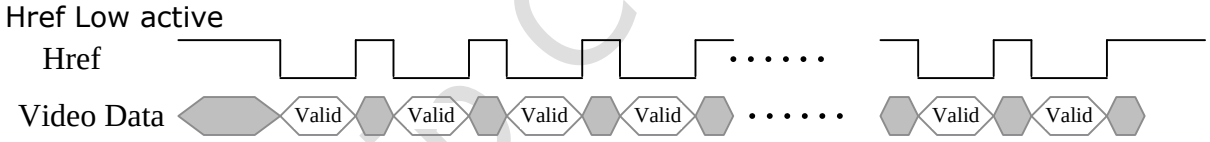

Fig. 18-5 Timing diagram for VIP when href low active

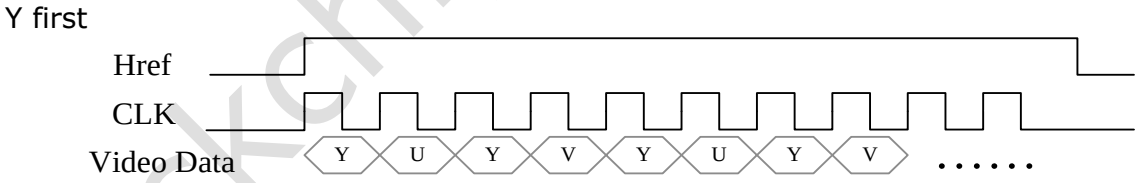

Fig. 18-6 Timing diagram for VIP when Y data first

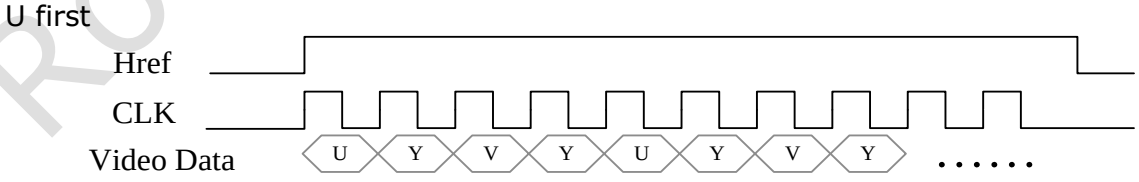

Fig. 18-7 Timing diagram for VIP when U data first

18.3.3 Support CCIR656 (NTSC and PAL)

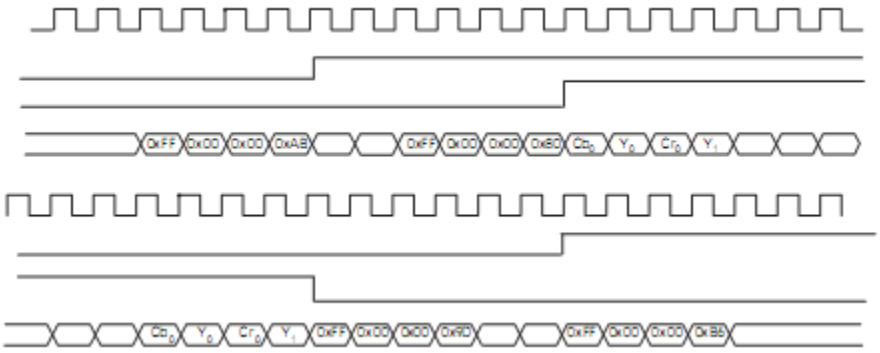

Fig. 18-8 CCIR656 timing

18.3.4 Support Raw data (8-bit) or JPEG

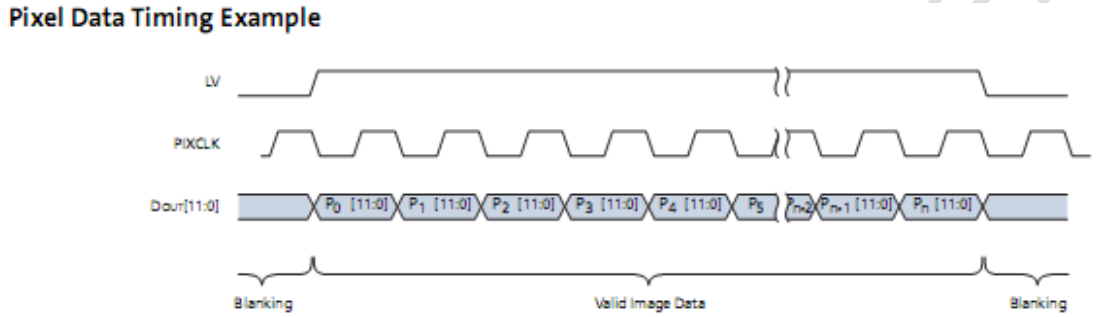

Fig. 18-9 Raw Data or JPEG Timing

VIP module can work in three 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, VIP will automatic stop. After capturing, the image Y, UV data will be stored at main memory location defined by VIP_FRM0_ADDR_Y, FRM0_ADDR_UV separately.

Ping-Pong mode

After one frame(F1) captured, VIP will start to capture the next frame(F2) automatically, and host must assign new address pointer of frame1 and clear the frame1 status, thus VIP 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 VIP 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 halfword 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.

18.4 Register description

18.4.1 Register Summary

Name	Offset	Size	Reset Value	Description
VIP_CTRL	0x00000	W	0x00007000	VIP control

Name	Offset	Size	Reset Value	Description
VIP_INTEN	0x00004	W	0x00000000	VIP interrupt status
VIP_INTSTAT	0x00008	W	0x00000000	VIP interrupt status
VIP_FOR	0x0000c	W	0x00000000	VIP format
VIP_FRM0_ADDR_Y	0x00014	W	0x00000000	VIP frame0 y address
VIP_FRM0_ADDR_UV	0x00018	W	0x00000000	VIP frame0 uv address
VIP_FRM1_ADDR_Y	0x0001c	W	0x00000000	VIP frame1 y address
VIP_FRM1_ADDR_UV	0x00020	W	0x00000000	VIP frame1 uv address
VIP_VIR_LINE_WIDTH	0x00024	W	0x00000000	VIP virtual line width
VIP_SET_SIZE	0x00028	W	0x01e002d0	VIP frame set size
VIP_CROP	0x00044	W	0x00000000	VIP crop start point
VIP_SCL_CTRL	0x00048	W	0x00000000	VIP scale control
VIP_FIFO_ENTRY	0x00054	W	0x00000000	VIP FIFO entry
VIP_FRAME_STATUS	0x00060	W	0x00000000	VIP frame status
VIP_CUR_DST	0x00064	W	0x00000000	VIP current destination address
VIP_LAST_LINE	0x00068	W	0x00000000	VIP last frame line number
VIP_LAST_PIX	0x0006c	W	0x00000000	VIP last line pixel number

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

18.4.2 Detail Register Description

VIP_CTRL
Address: Operational Base + offset (0x00000)
VIP 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

VIP_INTEN
Address: Operational Base + offset (0x00004)
VIP interrupt status

Bit	Attr	Reset Value	Description
31:10	RO	0x0	reserved

Bit	Attr	Reset Value	Description
9	RW	0x0	PST_INF_FRAME_END_EN frame end after interface FIFO interrupt enable 0 : disable 1 : enable
8	RW	0x0	PRE_INF_FRAME_END_EN frame end before interface FIFO interrupt enable 0 : disable 1 : enable
7	RO	0x0	reserved
6	W1C	0x0	BUS_ERR_EN axi master or ahb slave response error interrupt enable 0 : disable 1 : enable
5	RW	0x0	DFIFO_OF_EN DMA FIFO overflow interrupt enable 0 : disable 1 : enable
4	RW	0x0	IFIFO_OF_EN interface FIFO overflow interrupt enable 0 : disable 1 : enable
3	W1C	0x0	PIX_ERR_EN the pixel number of last line not equal to the set height interrupt enable 0 : disable 1 : enable
2	W1C	0x0	LINE_ERR_EN the line number of last frame not equal to the set height interrupt enable 0 : disable 1 : enable
1	W1C	0x0	LINE_END_EN line end interrupt enable 0 : disable 1 : enable
0	W1C	0x0	DMA_FRAME_END_EN dma frame end interrupt enable 0 : disable 1 : enable

VIP_INTSTAT
Address: Operational Base + offset (0x00008)

VIP interrupt status

Bit	Attr	Reset Value	Description
31:10	RO	0x0	reserved
9	RW	0x0	PST_INF_FRAME_END frame end after interface FIFO interrupt 0 : no interrupt 1 : interrupt
8	RW	0x0	PRE_INF_FRAME_END frame end before interface FIFO interrupt 0 : no interrupt 1 : interrupt
7	RO	0x0	reserved
6	W1C	0x0	BUS_ERR axi master or ahb slave response error interrupt 0 : no interrupt 1 : interrupt
5	RW	0x0	DFIFO_OF DMA FIFO overflow interrupt 0 : no interrupt 1 : interrupt
4	RW	0x0	IFIFO_OF interface FIFO overflow interrupt 0 : no interrupt 1 : interrupt
3	W1C	0x0	PIX_ERR the pixel number of last line not equal to the set height interrupt 0 : no interrupt 1 : interrupt
2	W1C	0x0	LINE_ERR the line number of last frame not equal to the set height interrupt 0 : no interrupt 1 : interrupt
1	W1C	0x0	LINE_END line end interrupt 0 : no interrupt 1 : interrupt
0	W1C	0x0	DMA_FRAME_END dma frame end interrupt 0 : no interrupt 1 : interrupt

VIP_FOR
Address: Operational Base + offset (0x0000c)

VIP 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 0 : UV in the even line 1 : 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	RO	0x0	reserved
14:13	RW	0x0	MIPI_MODE mipi mode 00 : 32 bit bypass mode; 01 : rgb mode; 10 : yuv mode; 11 : reserve;
12:11	RW	0x0	RAW_WIDTH raw data width 00 : 8bit raw data; 01 : 10bit raw data; 10 : 12bit raw data; 11 : reserve;
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 0 : UV in the even line 1 : UV in the odd line Note: The first line is even line(line 0).

Bit	Attr	Reset Value	Description
7	RW	0x0	INPUT_420 input 420 or 422 0 : 422 1 : 420
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 : low active 1 : high active

VIP_FRM0_ADDR_Y
Address: Operational Base + offset (0x00014)
VIP frame0 y address

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	FRM0_ADDR_Y frame0 y address

VIP_FRM0_ADDR_UV
Address: Operational Base + offset (0x00018)
VIP frame0 uv address

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	FRM0_ADDR_UV frame0 uv address

VIP_FRM1_ADDR_Y
Address: Operational Base + offset (0x0001c)
VIP frame1 y address

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	FRM1_ADDR_Y frame1 y address

VIP_FRM1_ADDR_UV
Address: Operational Base + offset (0x00020)
VIP frame1 uv address

Bit	Attr	Reset Value	Description
31:0	RW	0x00000000	FRM1_ADDR_UV frame1 uv address

VIP_VIR_LINE_WIDTH
Address: Operational Base + offset (0x00024)
VIP virtual line width

Bit	Attr	Reset Value	Description
31:15	RO	0x0	reserved
14:0	RW	0x0000	VIR_LINE_WIDTH virtual line width

VIP_SET_SIZE
Address: Operational Base + offset (0x00028)
VIP 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

VIP_CROP
Address: Operational Base + offset (0x00044)
VIP 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

VIP_SCL_CTRL
Address: Operational Base + offset (0x00048)
VIP scale control

Bit	Attr	Reset Value	Description

Bit	Attr	Reset Value	Description
31:7	RO	0x0	reserved
6	RW	0x0	MIPI_32B_BP mipi 32 bit bypass 0 : no bypass 1 : bypass
5	RW	0x0	RAW_16B_BP raw 16 bit bypass 0 : no bypass 1 : bypass
4	RW	0x0	YUV_16B_BP YUV 16 bit bypass 0 : no bypass 1 : bypass
3:0	RO	0x0	reserved

VIP_FIFO_ENTRY
Address: Operational Base + offset (0x00054)
VIP FIFO entry

Bit	Attr	Reset Value	Description
31:18	RO	0x0	reserved
17:9	RW	0x000	UV_FIFO_ENTRY valid UV double word in FIFO write 0 clear
8:0	RO	0x000	Y_FIFO_ENTRY valid Y double word in FIFO write 0 clear

VIP_FRAME_STATUS
Address: Operational Base + offset (0x00060)
VIP 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

VIP_CUR_DST
Address: Operational Base + offset (0x00064)
VIP 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.

VIP_LAST_LINE
Address: Operational Base + offset (0x00068)
VIP last frame line number

Bit	Attr	Reset Value	Description
31:13	RO	0x0	reserved
12:0	RO	0x0000	LAST_LINE_NUM line number of last frame

VIP_LAST_PIX
Address: Operational Base + offset (0x0006c)
VIP last line pixel number

Bit	Attr	Reset Value	Description
31:29	RO	0x0	reserved
28:16	RW	0x0000	LAST_UV_NUM y number of last line
15:13	RO	0x0	reserved
12:0	RO	0x0000	LAST_Y_NUM y number of last line

18.5 Interface description

Module Pin	Direction	Pad Name	IOMUX Setting
vip_clkout	O	IO_CIFclkin_HOSTwkack_GPSclk_HSADCclkout_DVPgpio2b2	GRF_GPIO2B_MUX[5:4]==2'b1
vip_clkin	I	IO_CIFclkout_HOSTwkreq_HSADCTSfail_DVPgpio2b3	GRF_GPIO2B_MUX[7:6]==2'b1
vip_href	I	IO_CIFhref_HOSTdin7_HSADCTSvalid_DVPgpio2b1	GRF_GPIO2B_MUX[3:2]==2'b1
vip_vsync	I	IO_CIFvsync_HOSTdin6_HSADCTSsync_DVPgpio2b0	GRF_GPIO2B_MUX[1:0]==2'b1
vip_data0	I	IO_CIFdata0_DVPgpio2b4	GRF_GPIO2B_MUX[9:8]==2'b1
vip_data1	I	IO_CIFdata1_DVPgpio2b5	GRF_GPIO2B_MUX[11:10]==2'b1
vip_data2	I	IO_CIFdata2_HOSTdin0_HSADCdata0_DVPgpio2a0	GRF_GPIO2A_MUX[1:0]==2'b1
vip_data3	I	IO_CIFdata3_HOSTdin1_HSADCdata1_DVPgpio2a1	GRF_GPIO2A_MUX[3:2]==2'b1

vip_data4	I	IO_CIFdata4_HOSTdin2_HSADCdata2_DVPgpio2a2	GRF_GPIO2A_MUX[5:4]==2'b1
vip_data5	I	IO_CIFdata5_HOSTdin3_HSADCdata3_DVPgpio2a3	GRF_GPIO2A_MUX[7:6]==2'b1
vip_data6	I	IO_CIFdata6_HOSTckinp_HSADCdata4_DVPgpio2a4	GRF_GPIO2A_MUX[9:8]==2'b1
vip_data7	I	IO_CIFdata7_HOSTckinn_HSADCdata5_DVPgpio2a5	GRF_GPIO2A_MUX[11:10]==2'b1
vip_data8	I	IO_CIFdata8_HOSTdin4_HSADCdata6_DVPgpio2a6	GRF_GPIO2A_MUX[13:12]==2'b1
vip_data9	I	IO_CIFdata9_HOSTdin5_HSADCdata7_DVPgpio2a7	GRF_GPIO2A_MUX[15:14]==2'b1
vip_data10	I	IO_CIFdata10_DVPgpio2b6	GRF_GPIO2B_MUX[13:12]==2'b1
vip_data11	I	IO_CIFdata11_DVPgpio2b7	GRF_GPIO2B_MUX[15:14]==2'b1

18.6 Application Notes

The biggest 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 VIP. The valid pixel number of scale result in FIFO can be known by read VIP_SCL_VALID_NUM. The line number of last frame and the pixel number of last line can be also known by read the VIP_LAST_LINE and VIP_LAST_PIX.