DATA SHEET

TRUEVIEW™ 5715

The Multimedia Processor for HD Input Progressive CRT TV

Version 1.01

Nov 2005

DS-5715-1.01

NOTE:

Tvia products are not designed for use in life support appliances, devices, or systems where malfunction of a Tvia product can reasonably be expected to result in personal injury. Distributors and/or customers using or selling such products for use in life support equipment do so at their own risk and agree to fully indemnify Tvia of such product liability or any damages resulting from such use or sale.

REVISION HISTORY

Revision	Description	Corresponding Page(s)
Version 1.0	Initial release	

Table of Contents

Table of Figures

Table of Tables

INTRODUCTION

The TrueView™ 5715 is a low pin count and low cost member of the TrueView™ family of advanced and highly integrated Multimedia Display Processors providing the key features needed to design the next generation progressive scan CRT Televisions.

The TrueView 5715 offers a unique solution to support and process video streams up to 1080I. It has a built-in de-Interlacer, advanced scaler, and dithering which enables a video source to stream through and be displayed without compromising the video quality.

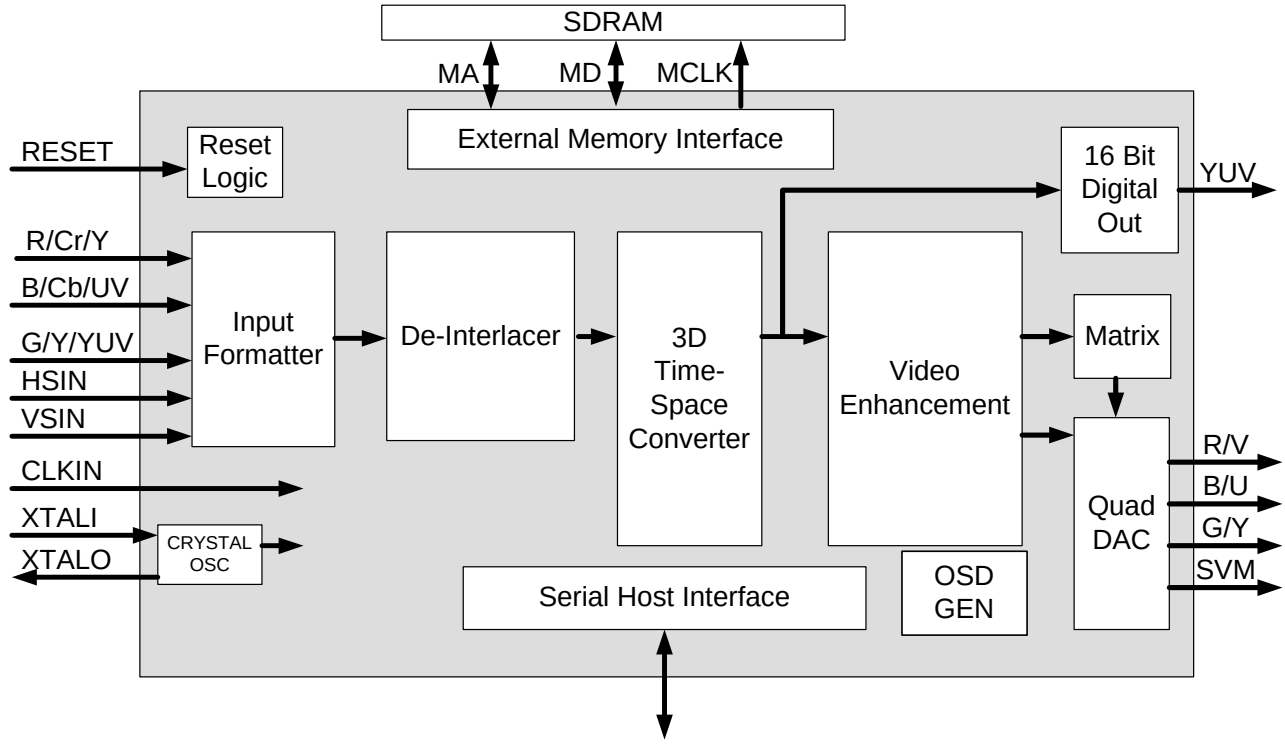
With the high quality video DACs, the video stream is displayed through its de-interlaced RGB/YPbPr outputs.

The I2C host interface enables OEMs to select from many different CPU's in order to meet their system and technical requirements.

OEMs can easily design a very low cost solution, while minimizing their software development time by leveraging on the TrueView 5715's level of integration and software support.

The TrueView 5715 provides 3D motion adaptive de-interlacing with diagonal detection. It performs high quality line doubling and high-accuracy motion estimation. It performs pixel- based motion estimation using two-field buffers. The Video Deinterlacing Processor automatically detects and can respond to noise thresholds (for improved mode detection) and still images (for reduced signal processing errors).

Additionally the TrueView 5715 includes an integrated OSD generator.



FEATURES

- **Input Formats**
  - 24-bit YCbCr/RGB
  - Single Video Port
  - 8 ITU-R BT.656
  - 8 or 16 bit ITU-R BT.601
  - Progressive/Interlaced
  - NTSC and PAL formats
  - 1080I, 1080P, 720P HD
  - Automatic format detection
  - 24 bit VGA, SVGA, XGA

- **Output Formats**
  - 480P, 540P, 576P, 720P, 1080I
  - 28-38K Hz Horizontal frequency
  - 50/60Hz to 100/120Hz double scan interlaced
  - 75Hz single scan interlaced
  - 50 - 75 Hz Scan Rate Conversion
  - YPbPr/RGB
  - Progressive 16 bit YCbCr with syncs

- **Video Enhancements**
  - Black and white level expansion
  - Color Transient Improvement (DCTI)
  - Dynamic Range expansion
  - Brightness, Saturation, Contrast
  - Dynamic Peaking
  - 2D coring
  - Scan Velocity Modulation (SVM)
  - Independent 2D up/down scaling
  - 3D Noise Reduction
  - Hue control
  - Vertical Peaking

- **De-Interlacer**
  - Direct Edge Correction De-interlace (DECD)
  - Motion Adaptive
  - Edge Adaptive
  - Mode Adaptive
  - Motion detection
  - Auto Noise detection
  - Still mode detection
  - 3:2/2:2 pull down mode detection

- **OSD**
  - OSD generator

- **Digital to Analog Conversion**
  - 10 bit quality
  - Quadruple DAC

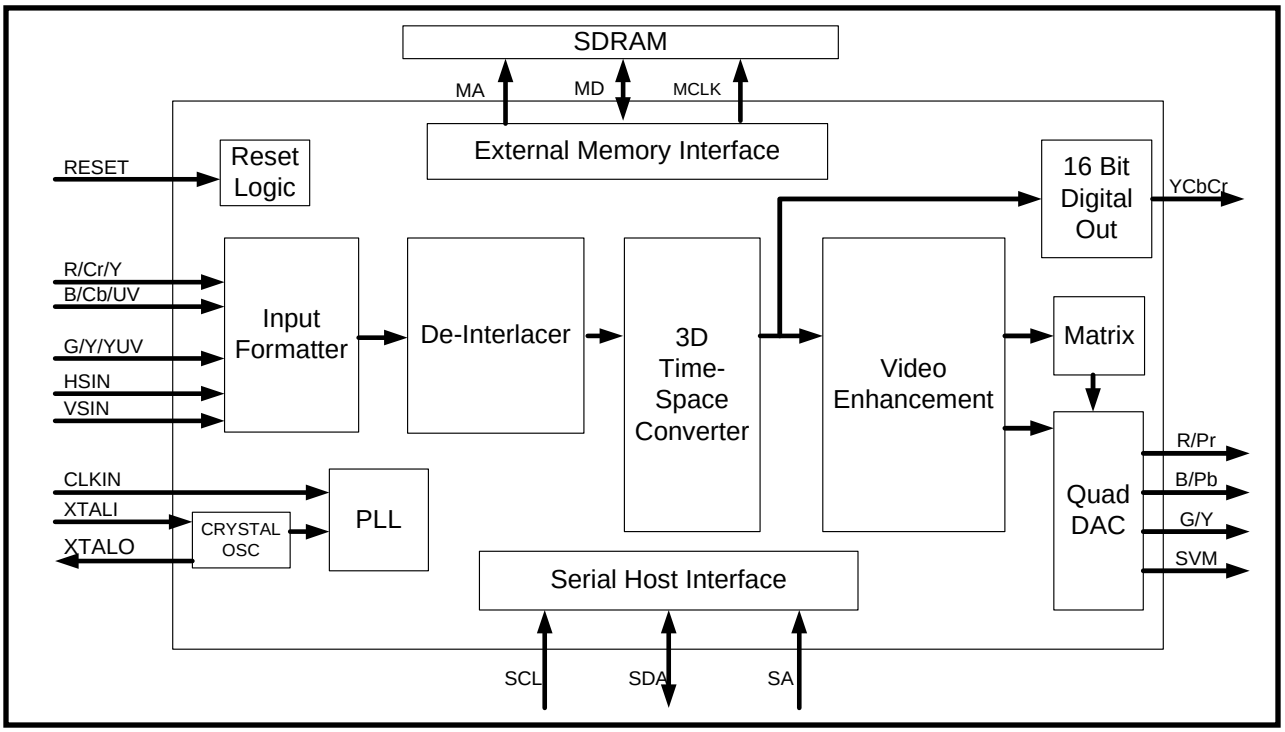
- **Memory**
  - SDRAM
  - 16 bits data access
  - 2/4/6/8 Mbytes

- **Host Interface and I/O**
  - 2 Wire serial bus
  - GPIO ports

- **100-Pin PQFP Package**
  - 0.35 Micron
  - 3.3V power supply

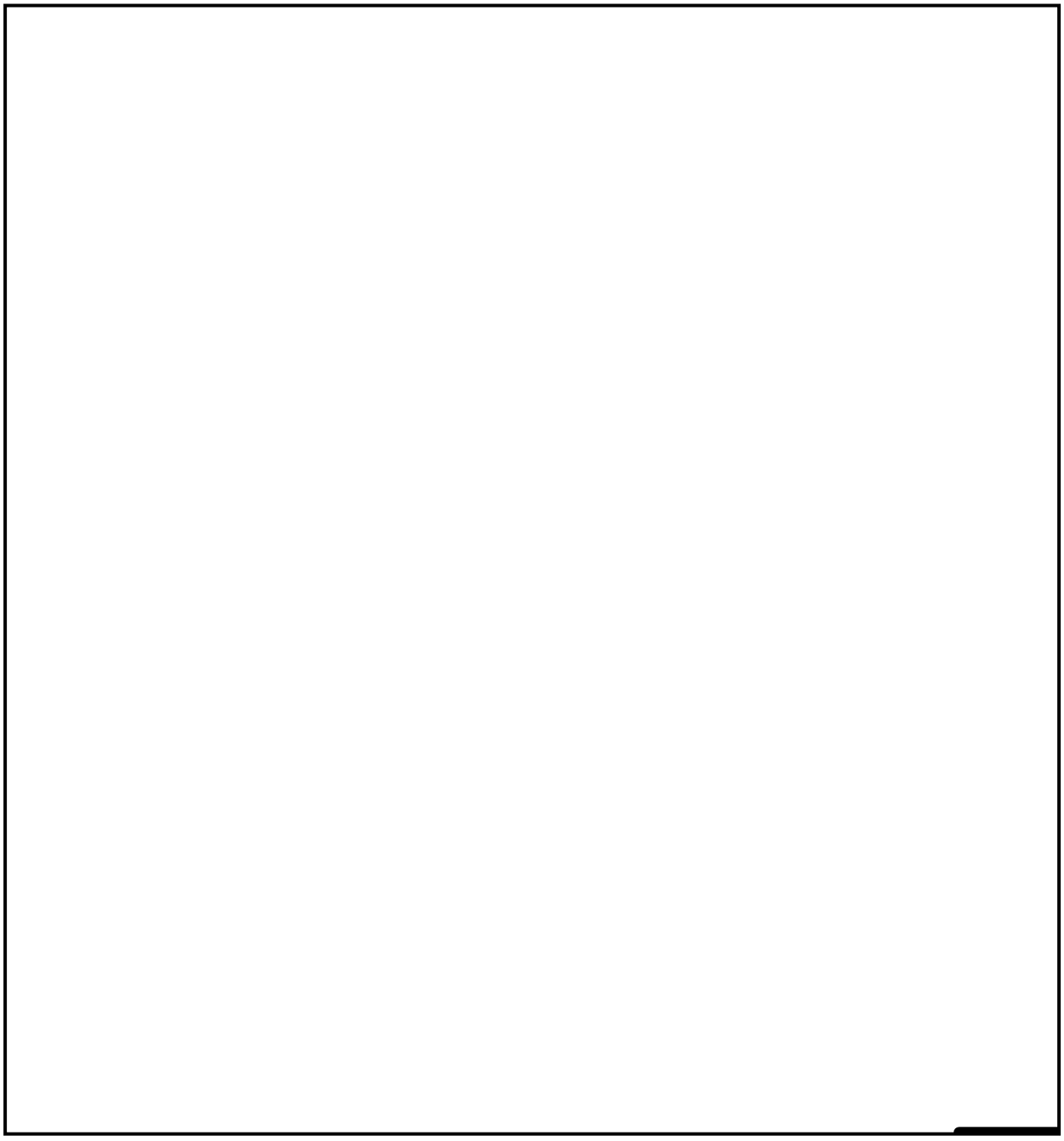
BLOCK DIAGRAM

Figure 1: Block Diagram



PINOUT DIAGRAM

Figure 2: 100-Pin QFP Pinout Diagram



PIN DESCRIPTION

Table 1: Pin List

Pin #	Pin Name	Pin Description	I/O	Note
1	AGPb	Analog Blue/Pb output	O	
2	AGY	Analog Green/Y output	O	
3	AGPr	Analog Red/Pr output	O	
4	VG0	Video Green Input data0	I/O	Video Green
5	VG1	Video Green Input data1	I/O	Y data for YCbCr mode
6	VG2	Video Green Input data2	I/O	Y data for 8-bit 4:2:2 mode.
7	VG3	Video Green Input data3	I/O	
8	VG4	Video Green Input data4	I/O	
9	VG5	Video Green Input data5	I/O	
10	VG6	Video Green Input data6	I/O	
11	VG7	Video Green Input data7	I/O	
12	GPIO	GPIO	I/O	
13	SCLSA	Serial bus slave address selection	I/O	GPIO pin
14	VSS	Core Power GND	G	
15	VDD	3.3V Core Power	P	
16	SCLCK	Serial bus clock	I	
17	SCLDA	Serial bus data	I/O	(External pull up)
18	VB0	Video Blue Input data0	I/O	Video Blue
19	VB1	Video Blue Input data1	I/O	Cb data for YCbCr mode
20	VB2	Video Blue Input data2	I/O	CbCr data for 16-bit 4:2:2 mode.
21	CLKIN	Input Pixel Clock	I	Input clock
22	HSIN	Video Input HSYNC	I	Horizontal sync
23	VSIN	Video Input VSYNC	I	Vertical sync
24	RST	Asynchronous reset	I	Active low
25	PVDD	3.3V I/O Power	P	
26	XTALO	External crystal output	O	27 MHz crystal out (bypass open)
27	XTALI	External crystal input	I	27 MHz crystal out (bypass clock in)
28	PVSS	I/O GND	G	
29	PAVD	Analog power for PLL	P	
30	PAVS	Analog ground for PLL	G	
31	VB3	Video Blue Input data3	I/O	
32	VB4	Video Blue Input data4	I/O	
33	VB5	Video Blue Input data5	I/O	
34	VB6	Video Blue Input data6	I/O	
35	VB7	Video Blue Input data7	I/O	
36	VR0	Video Red Input data0	I/O	Video Red
37	VR1	Video Red Input data1	I/O	Cr data for YCbCr mode

Pin #	Pin Name	Pin Description	I/O	Note
38	VR2	Video Red Input data2	I/O	YCBCR data for 16-bit 4:2:2 mode.
39	VR3	Video Red Input data3	I/O	
40	VR4	Video Red Input data4	I/O	
41	VR5	Video Red Input data5	I/O	
42	VR6	Video Red Input data6	I/O	
43	VR7	Video Red Input data7	I/O	
44	PVDD	3.3V Digital Power	P	
45	PVSS	Digital GND	G	
46	MD0	Memory Data Bus 0	I/O	
47	MD15	Memory Data Bus 15	I/O	
48	MD1	Memory Data Bus 1	I/O	
49	MD14	Memory Data Bus 14	I/O	
50	MD2	Memory Data Bus 2	I/O	
51	MD13	Memory Data Bus 13	I/O	
52	MD3	Memory Data Bus 3	I/O	
53	MD12	Memory Data Bus 12	I/O	
54	MD4	Memory Data Bus 4	I/O	
55	PVDD	3.3V I/O Power	P	
56	PVSS	I/O GND	G	
57	MD11	Memory Data Bus 11	I/O	
58	MD5	Memory Data Bus 5	I/O	
59	MD10	Memory Data Bus 10	I/O	
60	MD6	Memory Data Bus 6	I/O	
61	MD9	Memory Data Bus 9	I/O	
62	MD7	Memory Data Bus 7	I/O	
63	MD8	Memory Data Bus 8	I/O	
64	DQM#	SDRAM data qualify signal	O	
65	PVDD	3.3V I/O Power	P	
66	SDCLK	SDRAM clock	O	
67	PVSS	I/O GND	G	
68	WE#	Write control for SDRAM	O	
69	RAS#	Row address strobe	O	
70	CAS#	Column address strobe	O	
71	FBCLK	Feed back clock for memory controller	I	
72	BA	SDRAM bank select	O	
73	CS1#	Chip Selection 1	O	
74	VSS	Core Power GND	G	
75	VDD	3.3V Core Power	P	
76	CS0#	Chip Selection 0	O	
77	MA9	Memory address bus 9	I/O	
78	MA10	Memory address bus 10	I/O	
79	PVDD	3.3V I/O Power	P	
80	PVSS	I/O GND	G	

Pin #	Pin Name	Pin Description	I/O	Note
81	MA8	Memory address bus 8	I/O	
82	MA0	Memory address bus 0	I/O	
83	MA7	Memory address bus 7	I/O	
84	MA1	Memory address bus 1	I/O	
85	MA6	Memory address bus 6	I/O	
86	MA2	Memory address bus 2	I/O	
87	MA5	Memory address bus 5	I/O	
88	MA3	Memory address bus 3	I/O	
89	MA4	Memory address bus 4	I/O	
90	HALF	Half contrast control input	I/O	GPIO pin
91	PVDD	3.3V I/O Power	P	
92	PVSS	I/O GND	G	
93	CLKOUT	Output Pixel Clock	O	Display clock output
94	HSOUT	Video Output HSYNC	O	Horizontal sync
95	VSOUT	Video Output VSYNC	O	Vertical sync
96	DVSS	DAC GND	G	
97	ASVM	Analog SVM output	O	
98	IREF	Full-scale adjust resistor	I	
99	DAVS	Analog ground for DAC	G	
100	DAVD	Analog power for DAC	P	

Note

- Pin Type Definition

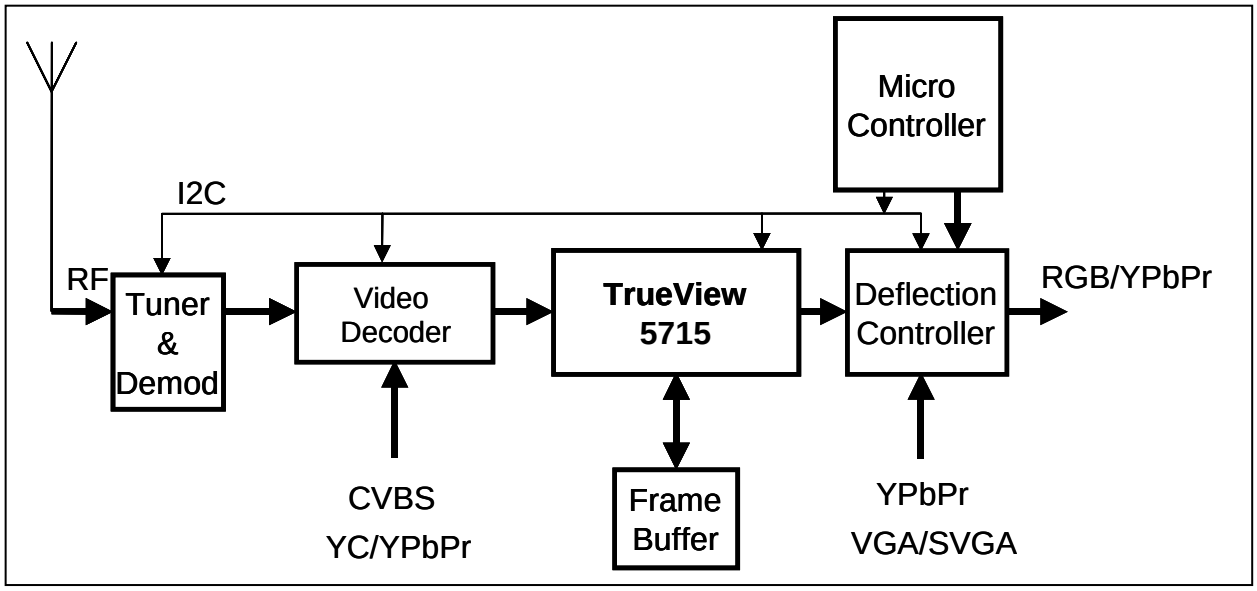
  I = Input
  O = Output
  G = Ground
  P = Power

APPLICATION: TV SYSTEMS

The following diagrams show three possible TV system configurations.

TV SYSTEM 1

Figure 3: TV system 1



TV system 1 shows a basic TV system that uses a standard video decoder to provide the interlaced input to the TrueView 5715.

TV SYSTEM 2

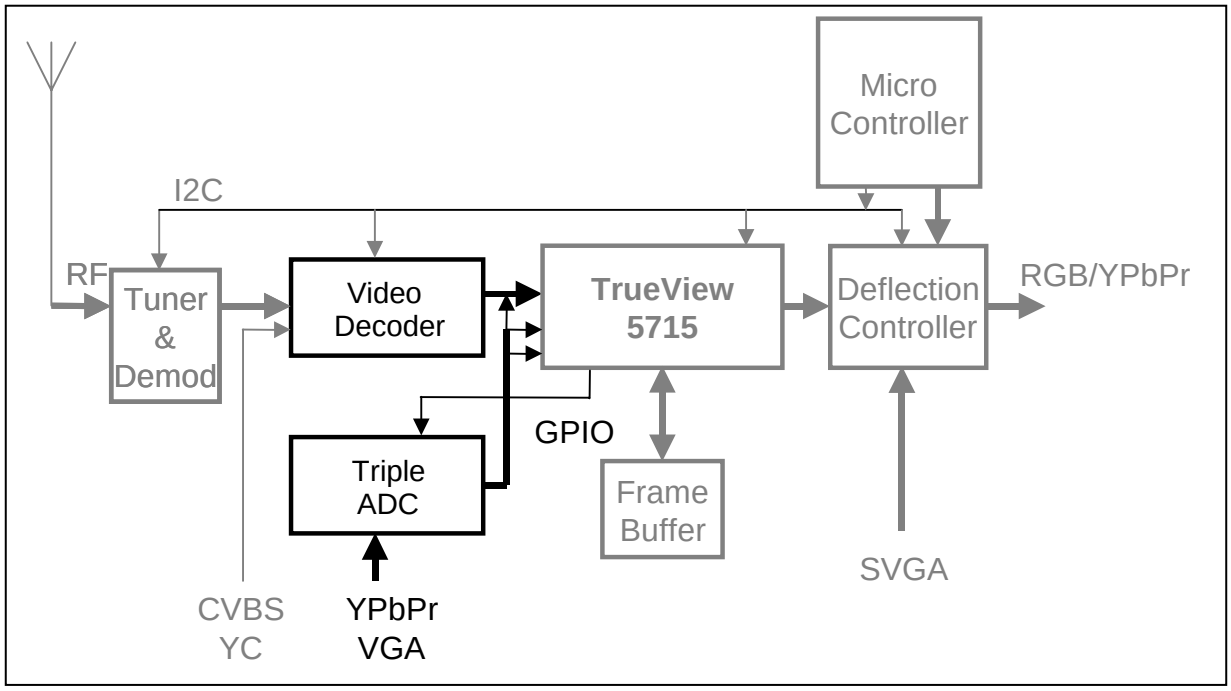
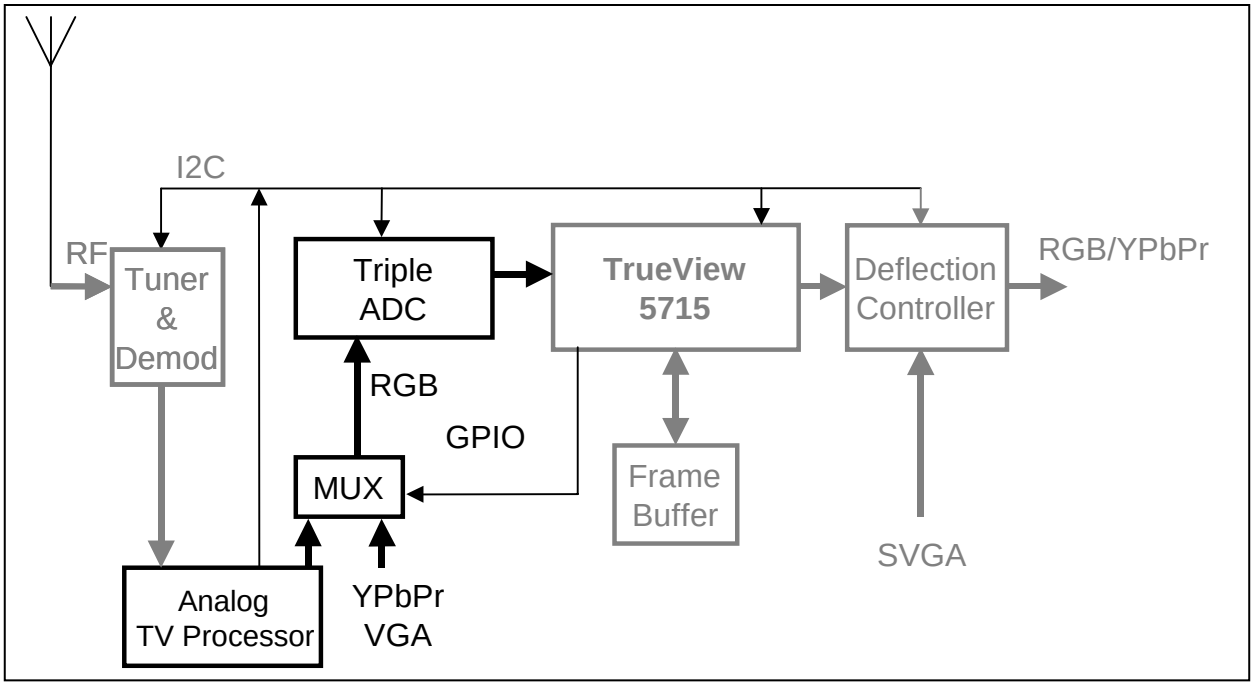


Figure 4: TV system 2

TV System 2 shows that with the addition of a triple Analog to Digital converter to enable VGA, RGB or YPbPr to be input to the TrueView 5715.

TV SYSTEM 3

Figure 5: TV system 3



TV System 3 shows the block diagram of a TV that uses a standard TV system processor that could contain a microcontroller and VBI decoder. The analog output of the TV processor is input to a triple analog to digital converter through an optional multiplexer that allows the input of external analog TV signals.

FUNCTIONAL DESCRIPTION

The TrueView 5715 TV Display Processor consists the following functional blocks:

- Input Formatter
- De-interlacer
- 3D Time-Space Converter
- Video Enhancer
- Output Formatter
- Analog and Digital Output
- Memory Controller
- PLL and Oscillator
- Serial Host Interface
- OSD Engine

INPUT FORMATTER

The Input Formatter accepts video data in several formats.

- 24-bit YCbCr or RGB
- 4:4:4 or 4:2:2
- 8 bit ITU-R BT.656
- 8 or 16 or 24 bit ITU-R BT.601
- NTSC or PAL
- 24 bit VGA (800*525@60Hz), SVGA, XGA
- 1080i, 720p, 1080p, HD

The incoming video format can be automatically detected and can be interlaced or progressive.

DE-INTERLACER

The TrueView 5715 deinterlacer uses motion-detection techniques to produce a progressive scan video output. Noise which otherwise would be seen as motion can be removed during motion processing. It also uses an edge detection algorithm that enables the smoothing of jagged edges that could occur in the de-interlacing process. Several different detection angles can be programmed.

FILM MODE

The De-interlace Processor can detect video that has originated from film and detects 3:2 or 2:2 pull down. If film originated video is detected the two fields from the original frame are combined into one progressive frame.

Thirty frames per second interlaced video can be output as 60 or 72Hz progressive frames.

Twenty-five frames per second interlaced video can be output as 50 or 75Hz progressive frames, or 100Hz-interlaced fields.

The de-interlacer continuously monitors the incoming video to detect any changes to the pull-down sequence. If a change is detected the deinterlacer quickly adapts its de-interlacing algorithm to compensate.

3D TIME – SPACE CONVERTER

The 3D Time – Space converter is used to alter the size of the picture depending on the output resolution. It contains an Up-Down scaler and Rate converter.

VIDEO ENHANCER

The Video Enhancer is a high-quality programmable processor that brings out details and color in the video.

Luminance Peaking Filter

The Luminance Peaking Filter can alter the frequency response of the luminance signal by selectively increasing the amplitude.

DCTI

Enhanced Chroma Transient improvement (ELCTI) improves video by replacing slower edges of the video with edges that have steeper rise and fall times. DCTI turns sloped or sinusoidal waveforms into rectangular or square waveforms with the same duty cycles and peak-to-peak amplitude. It improves the color transitions of vertical objects and reduces color smearing introduced by a video decoder.

Brightness and Contrast

Brightness and contrast can be programmed by adjusting the offset and gain of the video signal. Contrast is adjusted by multiplying the luminance by a constant. Brightness is adjusted by adding or subtracting a constant, from the luminance value.

Saturation

Saturation can be changed by changing the UV gain of the color signal

Coring

Coring is a noise reduction technique used in the TrueView 5715. If a signal is less than a certain threshold value, it is assumed noise and is set to zero. Coring is used to remove low amplitude signals in the high-pass plus band-pass signal.

3D Noise Reduction

3D noise reduction is used for reducing noise on Y & UV and mostly reducing the cross color on Chroma.

Black & white Level Expansion

The black and white-level expander enhances the contrast of the picture. The luminance signal is modified with an adjustable, non-linear function. Dark areas of the picture are made blacker, while bright areas of the picture are made whiter.

Vertical Peaking

Vertical peaking is used to improve Y and C sharpness in the vertical direction. It uses one line buffer which is shared with vertical scaling up and so vertical scaling up and vertical peaking can not be active at the same time.

OUTPUT FORMATTER

The Output formatter formats the output to give the required levels for RGB or YPbPr. It contains a 2X interpolator to increase sharpness that results from a smaller aperture.

ANALOG AND DIGITAL OUTPUT

The Analog Display Port generates the analog RGB or YPbPr with triple Digital-to-Analog Converters (DACs). The DAC's have 10-bit quality. The analog RGB or YPbPr output is generated in synchronization with H and V timing signals. The 16-bit YCbCr digital output is ITU-R BT.601 compatible with a 1 X clock. The output comes from the 3D time-space converter block and is in 4:2:2 format.

The Scan Velocity Modulation (SVM) output is used to modulate the horizontal CRT scan timing. It has the effect of sharpening the transitions from dark to light and light to dark that results in a clearer sharper picture.

MEMORY CONTROLLER

The internal Memory Controller supports the addressing and control of up to 8MB of external SDRAM. The external SDRAM memory buffer is used to store video fields and motion data. The Memory Controller also supports frame rate up-conversion with different input and output refresh rates.

PLL AND OSCILLATOR

The TrueView 5715 integrates a PLL to generate the MCLK to the Memory interface and the VCLK to the display. Only an external crystal is required to be connected between the XTALI and XTALO pins. On power-up, the PLL is initialized to provide a 108 MHz MCLK and a 27 MHz VCLK when a 27MHz reference is used. Alternatively, the MCLK and VCLK can be driven directly by external clocks.

HOST BUS INTERFACE

Access to the TrueView 5715 registers is provided by an I²C 2-wire serial bus interface. The Interface supports standard and fast modes, up to 400kbits/S. Only slave mode is supported in the TrueView 5715 TV Display Processor.

The slave address is hardware selectable.

Table 2: I²C Slave Addresses

TrueView 5715 I²C slave addresses	
Read	Write
2F	2E
AF	AE

REGISTERS DEFINITION

INPUT FORMATTER REGISTERS

INPUT_FORMATTER 00 **REG 10, R/W**

Bit	7	6	5	4	3	2	1	0
	Ini_st[0]	Matrix_byps	Progressive_ sel	Vs_sel	Sel_16bit Y_pipe_dly[1]	Sel_656	Y/UV flip UV_filter_ byps	In_dreg_by ps

Bit	Name	Function
0	In_dreg_byps	**Input pipe by pass** Use the falling or rising edge of clock to get the input data. 0: Clock input data on the falling edge of ICLK. 1: Clock input date on the rising edge of ICLK.
1	Y/UV Flip	**Y/UV Flip** Available only when input source is 8bit 0: Regular Y, UV order (Default). 1: Flip Y, UV
	UV_filter_byps	**UV filter bypass** Select filter when 444 format is converted to 422 format in 24bit mode, 0: enable the filter; 1: bypass the filter.
2	Sel_656	**Select CCIR656 data** If input data is 8bit CCIR656 mode, choose the 656 data path. 0: input is CCIR 601 mode. Choose the CCIR601mode timing. 1: input is CCIR 656 mode. Choose the CCIR656 mode timing.
3	Sel_16bit	**Select 16bit data** If source data is 16bit. Choose the 16bits data path. Use in conjunction with register sel_24bit to choose the input data format.

	Sel_16bit	Sel_24bit
8bit 656/601 input	0	0
16bit 601 input	1	0
24bit yuv/rgb 601 input	*	1

	Y_pipe_dly[1]	**Y path pipe delay** In 24bit mode, Y pipe delay to match the process of utap3.
4	Vs_sel	**Vertical sync select** Choose the periodical or virtual vertical timing. 0: choose the VCR mode timing generation. 1: choose the normal mode timing generation.
5	Progressive_sel	**Select progressive data** Progressive mode. Choose the progressive data. 0: source is interlaced. 1: source is progressive.
6	Matrix_byps	**Rgb2yuv matrix by pass**

Bit	Name	Function
		If source is yuv24bit, bypass the rgb2yuv matrix. 0:source is 24bit RGB. Do rgb2yuv. 1: data bypass.
7	Ini_st[0]	**Initial position** Initial position indicator for vertical blanking. For the internal line_counter, the pixel shift number that the line_counter counts compared to the horizontal sync.

INPUT_FORMATTER 01 — REG 11, R/W

Bit	7	6	5	4	3	2	1	0
	Ini_st [8:1]							

Bit	Name	Function
7-0	Ini_st[8:1]	**Initial position** Start position indicator of vertical blanking. For the internal line_counter, the detail pixel's shift that the line_counter count compare to the horizontal sync.

INPUT_FORMATTER 02 — REG 12, R/W

Bit	7	6	5	4	3	2	1	0
	Hs_flip	Test_cntrl[3:0]				Test_cntrl_en	Ini_st[10:9]	

Bit	Name	Function
1-0	Ini_st[10:9]	**Initial position** Start position indicator of vertical blanking. For the internal line_counter, the pixel shift number that the line_counter counts compared to the horizontal sync.
2	Test_cntrl_en	**IF test bus control enable** Enable test signal.
6-3	Test_cntrl[3:0]	**Test signals select bits.** Select which signal to the test bus.
7	Hs_flip	**Hsync flip** Invert the horizontal sync timing in Non-LLC decoder mode. 0: normal mode. Use the rising edge of horizontal sync to generate the initial timing. 1: invert the timing. Use the falling edge of horizontal sync to generate the initial timing.

INPUT_FORMATTER 03 — REG 13, R/W

Bit	7	6	5	4	3	2	1	0
	Hb_st[6:0]							Sel24bit

Bit	Name	Function
0	Sel24bit	**Select 24bit input data** Choose the 24bit data path if the source is 24bits.
7-1	Hb_st[6:0]	**Horizontal blanking start position (set 0)** Horizontal blanking (set 0) start position [6:0].

INPUT_FORMATTER 04 — REG 14, R/W

Bit	7	6	5	4	3	2	1	0
	Hb_sp[3:0]				Hb_st[10:7]			

Bit	Name	Function
3-0	Hb_st[10:7]	**Horizontal blanking start position (set 0)** Horizontal blanking (set 0) start position[10:7].
7-4	Hb_sp[3:0]	**Horizontal blanking stop position (set 0)** Horizontal blanking (set 0) stop position[3:0].

INPUT_FORMATTER 05 — REG 15, R/W

Bit	7	6	5	4	3	2	1	0
	Line_st[0]	Hb_sp[10:4]						

Bit	Name	Function
6-0	Hb_sp[10:4]	**Horizontal blanking stop position (set 0)** Horizontal blanking stop(set 0) position[10:4].
7	Line_st[0]	**Line signal start position** Progressive line start position.

INPUT_FORMATTER 06 — REG 16, R/W

Bit	7	6	5	4	3	2	1	0
	Line_st[8:1]							

Bit	Name	Function
7-0	Line_st[8:1]	**Line signal start position** Progressive line start position.

INPUT_FORMATTER 07 — REG 17, R/W

Bit	7	6	5	4	3	2	1	0
	Line_sp[5:0]						Line_st[10:9]	

Bit	Name	Function
1-0	Line_st[10:9]	**Line signal start position** Progressive line start position. Use together with line_sp to get one_progressive_line width timing.
7-2	Line_sp[5:0]	**Line signal stop position** Progressive line stop position. Use together with line_sp to get one_progressive_line width timing.

INPUT_FORMATTER 08 — REG 18, R/W

Bit	7	6	5	4	3	2	1	0
	Hsync_rst[2:0]			Line_sp[10:6]				

Bit	Name	Function
4-0	Line_sp[10:6]	**Line signal stop position** Progressive line stop position. Use together with line_st to get one_progressive_line width timing.
7-5	Hsync_rst[2:0]	**Total pixel number per line** Pix per line. Use to generate progressive timing if interlaced data is input. [2:0]

INPUT_FORMATTER 09 — REG 19, R/W

Bit	7	6	5	4	3	2	1	0
	Hsync_rst[10:3]							

Bit	Name	Function
7-0	Hsync_rst[10:3]	**Total pixel number per line** Pixel per line. Use to generate progressive timing if interlaced data is input. [10:3]

INPUT_FORMATTER 10 — REG 1A, R/W

Bit	7	6	5	4	3	2	1	0
	Vb_st[7:0]							

Bit	Name	Function
7-0	Vb_st[7:0]	**Vertical blanking start position** Vertical blanking start position.

INPUT_FORMATTER 11 — REG 1B, R/W

Bit	7	6	5	4	3	2	1	0
	Vb_sp[5:0]						Vb_st[9:8]	

Bit	Name	Function
1-0	Vb_st[9:8]	**Vertical blanking start position** Vertical blanking start position.
7-2	Vb_sp[5:0]	**Vertical blanking stop position** Vertical blanking stop position.

INPUT_FORMATTER 12 — REG 1C, R/W

Bit	7	6	5	4	3	2	1	0
	Hb_st1[1:0]		Vs_flip	Y_pipe_dly[0]	Vb_sp[9:6]			

Bit	Name	Function
3-0	Vb_sp[9:6]	**Vertical blanking stop position** Vertical blanking stop position[9:6].
4	Y_pipe_dly[0]	**Y path pipe delay** In 24bit mode, Y pipe delay to match the process of UV.
5	Vs_flip	**Vsync flip** Invert the vertical sync timing. 0: normal mode. Use the rising edge of vertical sync to generate the initial timing. 1: invert the timing. Use the falling edge of vertical sync to generate the initial timing.
7-6	Hb_st1[1:0]	**Horizontal blanking start position (set 1)** Horizontal blanking 1 starting position[1:0].

INPUT_FORMATTER 13 — REG 1D, R/W

Bit	7	6	5	4	3	2	1	0
	Hb_st1[9:2]							

Bit	Name	Function
7-0	Hb_st1[9:2]	**Horizontal blanking start position (set 1)** Horizontal blanking 1 start position[9:2].

INPUT_FORMATTER 14 — REG 1E, R/W

Bit	7	6	5	4	3	2	1	0
	Hb_sp1[6:0]							Hb_st1[10]

Bit	Name	Function
0	Hb_st1[10]	**Horizontal blanking start position (set 1)** Horizontal blanking 1 start position[10].
7-1	Hb_sp1[6:0]	**Horizontal blanking stop position (set 1)** Horizontal blanking 1 stop position[6:0].

INPUT_FORMATTER 15 — REG 1F, R/W

Bit	7	6	5	4	3	2	1	0
	Hb_st2[3:0]				Hb_sp1[10:7]			

Bit	Name	Function
3-0	Hb_sp1[10:7]	**Horizontal blanking stop position (set 1)** Horizontal blanking 1 stop position[10:7].
7-4	Hb_st2[3:0]	**Horizontal blanking start position (set 2)** Horizontal blanking 2 start position[3:0].

INPUT_FORMATTER 16 — REG 48, R/W

Bit	7	6	5	4	3	2	1	0
Bit	Hb_sp2[0]	Hb_st2[10:4]						

Bit	Name	Function
6-0	Hb_st2[10:4]	**Horizontal blanking start position (set 2)** Horizontal blanking 2 start position[10:4]
7	Hb_sp2[0]	**Horizontal blanking stop position (set 2)** Horizontal blanking 2 stop position[0]

INPUT_FORMATTER 17 — REG 49, R/W

Bit	7	6	5	4	3	2	1	0
Bit	Hb_sp2[8:1]							

Bit	Name	Function
7-0	Hb_sp2[8:1]	**Horizontal blanking stop position (set 2)** Horizontal blanking 2 stop position [8:1]

INPUT_FORMATTER 18 — REG 4A, R/W

Bit	7	6	5	4	3	2	1	0
Bit	Reserved						Hb_sp2[10:9]	

Bit	Name	Function
1-0	Hb_sp2[10:9]	**Horizontal blanking stop (set 2)** Horizontal blanking 2 stop position[10:9].
7-2	Reserved	

INPUT_FORMATTER 19 — REG 4B, R/W

Bit	7	6	5	4	3	2	1	0
Bit	Reserved							

Bit	Name	Function
7-0	Reserved	

INPUT_FORMATTER 20 **REG 4C, R/W**

Bit	7	6	5	4	3	2	1	0
	Reserved		Vb_sp_[10]	Vb_st_[10]	Yuv_lpf_byps	uv_sign2unsigh	Reserved	

Bit	Name	Function
1-0	Reserved	
2	u_sign2unsigh	**UV sign bit revert** Change the UV sign bit to match different kinds of source.
3	Yuv_lpf_byps	**YUV low pass filter bypass** When this bit sets 0, in 24 bit mode, y will through a 11-tape LPF, and u, v will through a 3-tape LPF, when this bit sets 1, these LPF will be bypassed
4	Vb_st_[10]	**Vertical blank start position** Vertical blank start position bit [10]
5	Vb_sp_[10]	**Vertical blank stop position** Vertical blank stop position bit [10]
7-6	Reserved	

MODE_DET REGISTERS

MODE_DET 00 **REG 7E_01/60, R/W**

Bit	7	6	5	4	3	2	1	0
	HPERIOD_UNLOCK_VALUE [2:0]			HPERIOD_LOCK_VALUE [4:0]				

Bit	Name	Function
4-0	HPERIOD_LOCK_VALUE [4:0]	**Mode Detect Horizontal Period Lock Value** If the continuous stabled line number is equal to the defined value, the horizontal stable indicator will be high
7-5	HPERIOD_UNLOCK_VALUE [2:0]	**Mode Detect Horizontal Period Unlock Value** If the continuous unstable line number is equal to the defined value, the horizontal stable indicator will be low

MODE_DET 01 **REG 7E_01/61, R/W**

Bit	7	6	5	4	3	2	1	0
	VPERIOD_UNLOCK_VALUE [2:0]			VPERIOD_LOCK_VALUE [4:0]				

Bit	Name	Function
4-0	VPERIOD_LOCK_VALUE [4:0]	**Mode Detect Vertical Period Lock Value** If the continuous stabled frame number is equal to the defined value, the vertical stable indicator will be high
7-5	VPERIOD_UNLOCK_VALUE [2:0]	**Mode Detect Vertical Period Unlock Value** If the continuous unstable frame number is equal to the defined value, the vertical stable indicator will be low

MODE_DET 02 — REG 7E_01/62, R/W

Bit	7	6	5	4	3	2	1	0
	WEN_CNTRL [1:0]		NTSC_INT_CNTRL [5:0]					

Bit	Name	Function
5-0	NTSC_INT_CNTRL [5:0]	**NTSC Interlace Mode Detect Value** If the vertical period number is equal to the defined value, This mode is NTSC Interlace mode
7-6	WEN_CNTRL [1:0]	**Horizontal Stable Estimation Error Range Control** The continuous line is stable in the defined error range. Range Table:

WEN_CNTRL [1]	WEN_CNTRL [0]	Error Range
0	0	1
0	1	2
1	0	3
1	1	4

MODE_DET 03 — REG 7E_01/63, R/W

Bit	7	6	5	4	3	2	1	0
	VS_FLIP	HS_FLIP	PAL_INT_CNTRL [5:0]					

Bit	Name	Function
5-0	PAL_INT_CNTRL [5:0]	**PAL Interlace Mode Detect Value** If the vertical period number is equal to the defined value, This mode is PAL interlace mode
6	HS_FLIP	**Input Horizontal sync polarity Control** When set it to 1, the input horizontal sync will be inverted.
7	VS_FLIP	**Input Vertical sync polarity Control** When set it to 1, the input vertical sync will be inverted.

MODE_DET 04 — REG 7E_01/64, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED	NTSC_PRG_CNTRL [6:0]						

Bit	Name	Function
6-0	NTSC_PRG_CNTRL [6:0]	**NTSC Progressive Mode Detect Value** If the vertical period number is equal to the defined value, This mode is NTSC progressive mode or VGA 60HZ mode
7	RESERVED	

MODE_DET 05 — REG 7E_01/65, R/W

Bit	7	6	5	4	3	2	1	0
	SEL_VGA60	VGA_CNTRL [6:0]						

Bit	Name	Function
6-0	VGA_CNTRL [6:0]	**VGA Mode Vertical Detect Value** If the vertical period number is equal to the defined value, this mode is VGA mode, except VGA 60HZ mode.
7	SEL_VGA60	**Select VGA 60HZ mode** Program this bit to distinguish between VGA 60Hz mode and NTSC progressive mode; When set to 1, select VGA 60Hz mode When set to 0, select NTSC progressive mode

MODE_DET 06 — REG 7E_01/66, R/W

Bit	7	6	5	4	3	2	1	0
	VGA_75HZ_CNTRL [7:0]							

Bit	Name	Function
7-0	VGA_75HZ_CNTRL [7:0]	**VGA 75Hz Horizontal Detect Value** If the horizontal period number is equal to the defined value, in VGA mode, this mode Is VGA 75Hz mode.

MODE_DET 07 — REG 7E_01/67, R/W

Bit	7	6	5	4	3	2	1	0
	VGA_85HZ_CNTRL [7:0]							

Bit	Name	Function
7-0	VGA_85HZ_CNTRL [7:0]	**VGA 85Hz Horizontal Detect Value** If the horizontal period number is equal to the defined value, in VGA mode, this mode Is VGA 85Hz mode.

MODE_DET 08 — REG 7E_01/68, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED	V1250_VCNTRL [6:0]						

Bit	Name	Function
6-0	V1250_VCNTRL [6:0]	**Vertical 1250 Line Mode Vertical Detect Value** All vertical 1250 lines mode vertical detect value
7	RESERVED	

MODE_DET 09 — REG 7E_01/69, R/W

Bit	7	6	5	4	3	2	1	0
	V1250_HCNTRL [7:0]							

Bit	Name	Function
7-0	V1250_HCNTRL [7:0]	**Vertical 1250 Line Mode Horizontal Detect Value** Vertical 1250 lines, horizontal 866 pixels mode detect value

MODE_DET 10 — REG 7E_01/6A, R/W

Bit	7	6	5	4	3	2	1	0
	SVGA_60HZ_CNTRL [7:0]							

Bit	Name	Function
7-0	SVGA_60HZ_CNTRL [7:0]	**SVGA 60HZ Mode Horizontal Detect Value** If the horizontal period number is equal to the defined value, in SVGA mode, it's SVGA 60Hz mode.

MODE_DET 11 — REG 7E_01/6B, R/W

Bit	7	6	5	4	3	2	1	0
	SVGA_75HZ_CNTRL [7:0]							

Bit	Name	Function
7-0	SVGA_75HZ_CNTRL [7:0]	**SVGA 75HZ Mode Horizontal Detect Value** If the horizontal period number is equal to the defined value, in SVGA mode, it's SVGA 75Hz mode.

MODE_DET 12 — REG 7E_01/6C, R/W

Bit	7	6	5	4	3	2	1	0
	SVGA_85HZ_CNTRL [7:0]							

Bit	Name	Function
7-0	SVGA_85HZ_CNTRL [7:0]	**SVGA 85HZ Mode Horizontal Detect Value** If the horizontal period number is equal to the defined value, in SVGA mode, it's SVGA 85Hz mode.

MODE_DET 13 — REG 7E_01/6D, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED	XGA_CNTRL [6:0]						

Bit	Name	Function
6-0	XGA_CNTRL [6:0]	**XGA Mode Vertical Detect Value** If the vertical period number is equal to the defined value, it's XGA mode.
7	RESERVED	

MODE_DET 14 — REG 7E_01/6E, R/W

Bit	7	6	5	4	3	2	1	0
	XGA_60HZ_CNTRL [7:0]							

Bit	Name	Function
7-0	XGA_60HZ_CNTRL [7:0]	**XGA 60Hz Mode Horizontal Detect Value** If the horizontal period number is equal to the defined value, in XGA modes, It's XGA 60Hz mode.

MODE_DET 15 — REG 7E_01/6F, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED	XGA_70HZ_CNTRL [6:0]						

Bit	Name	Function
6-0	XGA_70HZ_CNTRL [6:0]	**XGA 70Hz Mode Horizontal Detect Value** If the horizontal period number is equal to the defined value, in XGA modes, It's XGA 70Hz mode.
7	RESERVED	

MODE_DET 16 — REG 7E_02/60, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED	XGA_75HZ_CNTRL [6:0]						

Bit	Name	Function
6-0	XGA_75HZ_CNTRL [6:0]	**XGA 75Hz Mode Horizontal Detect Value** If the horizontal period number is equal to the defined value, in XGA modes, It's XGA 75Hz mode.
7	RESERVED	

MODE_DET 17 — REG 7E_02/61, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED	XGA_85HZ_CNTRL [6:0]						

Bit	Name	Function
6-0	XGA_85HZ_CNTRL [6:0]	**XGA 85Hz Mode Horizontal Detect Value** If the horizontal period number is equal to the defined value, in XGA modes, It's XGA 85Hz mode.
7	RESERVED	

MODE_DET 18 — REG 7E_02/62, R/W

Bit	7	6	5	4	3	2	1	0
	SXGA_CNTRL [7:0]							

Bit	Name	Function
7-0	SXGA_CNTRL [7:0]	**SXGA Mode Vertical Detect Value** If the vertical period number is equal to the defined value, It's SXGA mode.

MODE_DET 19 — REG 7E_02/63, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED	SXGA_60HZ_CNTRL [6:0]						

Bit	Name	Function
6-0	SXGA_60HZ_CNTRL [6:0]	**SXGA 60Hz Mode Horizontal Detect Value** If the horizontal period number is equal to the defined value, in SXGA modes, It's SXGA 60Hz mode.
7	RESERVED	

MODE_DET 20 — REG 7E_02/64, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED	SXGA_75HZ_CNTRL [6:0]						

Bit	Name	Function
6-0	SXGA_75HZ_CNTRL [6:0]	**SXGA 75Hz Mode Horizontal Detect Value** If the horizontal period number is equal to the defined value, in SXGA modes, It's SXGA 75Hz mode.
7	RESERVED	

MODE_DET 21 — REG 7E_02/65, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED	SXGA_85HZ_CNTRL [6:0]						

Bit	Name	Function
6-0	SXGA_85HZ_CNTRL [6:0]	**SXGA 85Hz Mode Horizontal Detect Value** If the horizontal period number is equal to the defined value, in SXGA modes, It's SXGA 85Hz mode.
7	RESERVED	

MODE_DET 22 — REG 7E_02/66, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED	HD720P_CNTRL [6:0]						

Bit	Name	Function
6-0	HD720P_CNTRL [6:0]	**HD720P Vertical Detect Value** If the vertical period number is equal to the defined value, It's HD720P mode.
7	RESERVED	

MODE_DET 23 — REG 7E_02/67, R/W

Bit	7	6	5	4	3	2	1	0
	HD720P_60HZ_CNTRL [7:0]							

Bit	Name	Function
7-0	HD720P_60HZ_CNTRL [7:0]	**HD720P 60Hz Mode Horizontal Detect Value** If the horizontal period number is equal to the defined value, in HD720P mode. It is HD720P 60Hz mode.

MODE_DET 24 — REG 7E_02/68, R/W

Bit	7	6	5	4	3	2	1	0
	HD720P_50HZ_CNTRL [7:0]							

Bit	Name	Function
7-0	HD720P_50HZ_CNTRL [7:0]	**HD720P 50Hz Mode Horizontal Detect Value** If the horizontal period number is equal to the defined value, in HD720P mode. It is HD720P 50Hz mode.

MODE_DET 25 — REG 7E_02/69, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED	HD1125I_CNTRL [6:0]						

Bit	Name	Function
6-0	HD1125I_CNTRL [6:0]	**1080I Mode 1125 Line Vertical Detect Value** If the vertical period number is equal to the defined value, It's 1125I mode.
7	RESERVED	

MODE_DET 26 — REG 7E_02/6A, R/W

Bit	7	6	5	4	3	2	1	0
	HD2200_1125I_CNTRL [7:0]							

Bit	Name	Function
7-0	HD2200_1125I_CNTRL [7:0]	**1080I Mode 2200x1125I Horizontal Detect Value** If the horizontal period number is equal to the defined value, in 1080I mode. It is HD2200x1125I mode.

MODE_DET 27 — REG 7E_02/6B, R/W

Bit	7	6	5	4	3	2	1	0
	HD2640_1125I_CNTRL [7:0]							

Bit	Name	Function
7-0	HD2640_1125I_CNTRL [7:0]	**1080I Mode 2640x1125I Horizontal Detect Value** If the horizontal period number is equal to the defined value, in 1080I mode. It is HD2640x1125I mode.

MODE_DET 28 — REG 7E_02/6C, R/W

Bit	7	6	5	4	3	2	1	0
	HD1125P_CNTRL [7:0]							

Bit	Name	Function
7-0	HD1125P_CNTRL [7:0]	**1080P Mode 1125 Line Vertical Detect Value** If the vertical period number is equal to the defined value, It is HD1125P mode.

MODE_DET 29 — REG 7E_02/6D, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED	HD2200_1125P_CNTRL [6:0]						

Bit	Name	Function
6-0	HD2200_1125P_CNTRL [6:0]	**1080P Mode 2200x1125P Horizontal Detect Value** If the horizontal period number is equal to the defined value, in 1080P mode, It is HD2200x1125P mode
7	RESERVED	

MODE_DET 30 — REG 7E_02/6E, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED	HD2640_1125P_CNTRL [6:0]						

Bit	Name	Function
6-0	HD2640_1125P_CNTRL [6:0]	**1080P Mode 2640x1125P Horizontal Detect Value** If the horizontal period number is equal to the defined value, in 1080P mode, It is HD2640x1125P mode
7	RESERVED	

MODE_DET 31 **REG 7E_02/6F, R/W**

Bit	7	6	5	4	3	2	1	0
	HD1250P_CNTRL [7:0]							

Bit	Name	Function
7-0	HD1250P_CNTRL [7:0]	**1080P Mode 2376x1250P Vertical Detect Value** If the vertical period number is equal to the defined value, It is HD2376x1250P mode

DEINTERLACER REGISTERS

DEINTERLACER 00 — REG 20, R/W

Bit	7	6	5	4	3	2	1	0
	Reserved	LIN_DB_RAM_BYPS	Reserved	DIAG_DET_BYPS[1]	DIAG_DET_BYPS[0]	WEAVE_BYPS	DIAG_COEF_SEL	DIAG_MIN_BYPS

Bit	Name	Function
0	DIAG_MIN_BYPS	**Diagonal Function Bypass Control** When set to 1, bypass diagonal min selection. No diagonal detection, just vertically two pixels average.
1	DIAG_COEF_SEL	**Diagonal Bob Low pass Filter Coefficient Selection** Select coefficients for pixel difference low pass filter DIAG_COEF_SEL: 1 → 15/16; 0 → 14/16
2	WEAVE_BYPS	**Weave Function Bypass Control** When set to 1, weave function will bypass. Just repeat original data.
3	DIAG_DET_BYPS[0]	**Diagonal Bob Deinterlacer Angle Detect Bypass Control** When set to 1, bypass the detection of angle arctan(1/4).
4	DIAG_DET_BYPS[1]	**Diagonal Bob Deinterlacer Angle Detect Bypass Control** When set to 1, bypass the detection of angle arctan(1/6).
5	Reserved	
6	LIN_DB_RAM_BYPS	**Bypass Control For Line_doubler FIFO** When set to 1, bypass FIFO for line_doubler.
7	Reserved	

DEINTERLACER 01 — REG 21, R/W

Bit	7	6	5	4	3	2	1	0
	DIAG_PLDY_SP[2:0]			DELAY_ON	LIN_DB_ST[3:0]			

Bit	Name	Function
3-0	LIN_DB_ST[3:0]	**The Distance Control of Line Double Reset** In line_doubler, adjust the delay between read reset and write reset. Actual value = Programmed value + 2
4	DELAY_ON	**Line Signal Delay Control** When set to 1, delay line signal 5 clocks (Only used in Non-LLC decoder mode).
7-5	DIAG_PLDY_SP[2:0]	**The Distance Control of Pldelayer Reset [2:0]** In pldelayer, adjust the delay between read reset and write reset.

DEINTERLACER 02 — REG 22, R/W

Bit	7	6	5	4	3	2	1	0
	Reserved	DELAY_BYPS	DIAG_PLDY_SP[8:3]					

Bit	Name	Function
5-0	DIAG_PLDY_SP[8:3]	**The Distance Control of Pldelayer Reset [8:3]** In pldelayer, adjust the delay between read reset and write reset.
6	DELAY_BYPS	**Line Reset Delay Control** When set to 1, reset signals (write and read) could be set 2 clocks ahead. When set to 0 (Only supposed to be used in Non-LLC decoder mode).
7	Reserved	

DEINTERLACER 03 — REG 23, R/W

Bit	7	6	5	4	3	2	1	0
	Y_MI_GEN_BYPS	DIAG_TEST_BUS_CNTRL[4:0]					DIAG_PLDY_RAM_BYPS	Reserved

Bit	Name	Function
0	Reserved	
1	DIAG_PLDY_RAM_BYPS	**Bypass Control For Pldelayer FIFO** When set to 1, bypass FIFO for pldelayer.
6-2	DIAG_TEST_BUS_CNTRL [4:0]	**Diagonal Bob Test Bus Control** Internal hardware debugging use only.
7	Y_MI_GEN_BYPS	**Y motion index generation bypass** When set to 1, Y motion index generation is in manual mode

DEINTERLACER 04 — REG 24, R/W

Bit	7	6	5	4	3	2	1	0
	Y_MI_GAIN[0]	Y_MI_OFST[6:0]						

Bit	Name	Function
6-0	Y_MI_OFST[6:0]	**Y motion index offset** In auto mode, Y motion index's offset. In manual mode, Y motion index's user value.
7	Y_MI_GAIN[0]	**Y motion index gain bit[0]** Y motion index gain bit[0]

DEINTERLACER 05 — REG 25, R/W

Bit	7	6	5	4	3	2	1	0
	UV_MI_OFST[3:0]				UV_MI_GEN_BYPS	Y_MI_GAIN[3:1]		

Bit	Name	Function
2-0	Y_MI_GAIN[3:1]	**Y motion index gain bit[3:1]** Y motion index gain bit [3:1]
3	UV_MI_GEN_BYPS	**UV motion index generation bypass** When set to one, UV motion index generation is in manual mode.
7-4	UV_MI_OFST[3:0]	**UV motion index offset bit [3:0]** In auto mode, UV motion index offset bit [3:0] In manual mode, UV motion index user value bit [3:0]

DEINTERLACER 06 — REG 26, R/W

Bit	7	6	5	4	3	2	1	0
	MI_DELAY[0]	UV_MI_GAIN[3:0]				UV_MI_OFST[6:4]		

Bit	Name	Function
2-0	UV_MI_OFST[6:4]	**UV motion index offset bit [6:4]** In auto mode, UV motion index offset bit [6:4] In manual mode, UV motion index user value bit [6:4]
6-3	UV_MI_GAIN[3:0]	**UV motion index gain** UV motion index gain.
7	MI_DELAY[0]	**Motion index delay control bit [0]**

DEINTERLACER 07 — REG 27, R/W

Bit	7	6	5	4	3	2	1	0
	VTAP2_COEF				VTAP2_ROUND_SEL	VTAP2_BYPS	MI_DELAY[2:1]	

Bit	Name	Function
1-0	MI_DELAY[2:1]	**Motion index delay control bit [2:1]** Control motion index (both Y and UV's) delay pipes

MI_DELAY[2:0]	Motion index delay pipes
000	0 pipe
001	1 pipe
010	2 pipes
011	3 pipes
100	4 pipes
101	5 pipes
110	6 pipes
111	7 pipes

Bit	Name	Function
2	VTAP2_BYPS	**Motion index vertical filter bypass** When = 1, motion index's vertical filter will be bypass.
3	VTAP2_ROUND_SEL	**Motion index vertical filter round selection**
7-4	VTAP2_COEF	**Motion index vertical filter coefficient**

DEINTERLACER 08 — REG 28, R/W

Bit	7	6	5	4	3	2	1	0
	DIVID_SEL	DIVID_BYPS	EN_UVDEINT		HTAP_COEF			HTAP_BYPS

Bit	Name	Function
0	HTAP_BYPS	**Motion index horizontal filter bypass** When =1, motion index horizontal filter will be bypass
4-1	HTAP_COEF	**Motion index horizontal filter coefficient** Motion index horizontal filter coefficient.
5	EN_UVDEINT	**Enable UV deinterlacer** When = 1, enable UV deinterlacer
6	DIVID_BYPS	**Motion index divide bypass** When = 1, motion index no divide. When = 0, motion index will by divided by 2 or 4.
7	DIVID_SEL	**Motion index divide selection** When = 1, motion index will be divided by 2 in still. When = 0, motion index will be divided by 4 in still.

DEINTERLACER 09 — REG 29, R/W

Bit	7	6	5	4	3	2	1	0
	STILL_LOCK				STILL_ID	STILL_DET_EN	VT_FILTER_CNTRL	MO_ADP_EN

Bit	Name	Function
0	MO_ADP_EN	**Enable pull-down in motion adaptive** When set to 1, enable pull-down for motion adaptive.
1	VT_FILTER_CNTRL	**Vertical Temporal Filter Control** When set to 1, do motion adaptive in interpolated line only. When set to 0, do motion adaptive in every line.
2	STILL_DET_EN	**Still detection enable** When set to 1, still detection is in auto mode. When set to 0, still detection is in manual mode.
3	STILL_ID	**Still indicator defined by user (in manual mode only)** Still indicator defined by user, only useful in STILL_DET_EN =0.
7-4	STILL_LOCK	**Still detection's auto lock value** When lock counter equals lock value, "still" will go active.

DEINTERLACER 10 — REG 2A, R/W

Bit	7	6	5	4	3	2	1	0
	STILL_NOISE_VALUE[4:0]					STILL_NOISE_EST_EN	STILL_UNLOCK	

Bit	Name	Function
2-0	STILL_UNLOCK[3:0]	**Still detection's auto unlock value** When unlock counter equals unlock value, "still" will go inactive.
3	STILL_NOISE_EST_EN	**Still-noise auto detection enable** When set to 1, still-noise is in auto detection; When set to 0, still-noise is in manual mode.
7-4	STILL_NOISE_VALUE[4:0]	In manual mode, still-noise value bit [4:0] In auto-detect mode, still-noise's offset bit [4:0]

DEINTERLACER 11 — REG 2B, R/W

Bit	7	6	5	4	3	2	1	0
	STILL_FOR_PULLDOWN	STILL_NOISE_GAIN				STILL_NOISE_VALUE[7:5]		

Bit	Name	Function
2-0	STILL_NOISE_VALUE[7:5]	In manual mode, still-noise value bit [7:5] In auto-detect mode, still-noise's offset bit [7:5]
6-3	STILL_NOISE_GAIN	**Still-noise gain (in auto-detect mode)**
7	STILL_FOR_PULLDOWN	**Enable STILL to reset pull-down detection** When set to 1, still will be used to reset 3:2/2:2 pull-down detection.

DEINTERLACER 12 — REG 2C, R/W

Bit	7	6	5	4	3	2	1	0
	LESS_STILL_UNLOCK[0]	LESS_STILL_LOCK				LESS_STILL_ID	LESS_STILL_DET_EN	STILL_FOR_NRD

Bit	Name	Function
0	STILL_FOR_NRD	**Enable still for noise reduction** When set to 1, still will be used by background noise reduction.
1	LESS_STILL_DET_EN	**Less-still detection enable** When set to 1, less-still detection is in auto mode. When set to 0, less-still detection is in manual mode.
2	LESS_STILL_ID	**Less-still indicator defined by user (in manual mode)** Less-still indicator defined by user, only useful in LESS_STILL_DET_EN =0
6-3	LESS_STILL_LOCK	**Less-still auto lock value** When less-still lock counter equals lock value, less-still will be active
7	LESS_STILL_UNLOCK[0]	**Less-still auto unlock value bit [0]** When less-still unlock counter equals unlock value, less-still will be unactive

DEINTERLACER 13 — REG 2D, R/W

Bit	7	6	5	4	3	2	1	0
	LESS_STILL_NOISE[6:0]							LESS_STILL_UNLOCK[1]

Bit	Name	Function
0	LESS_STILL_UNLOCK[1]	**Less-still auto unlock value bit [1]** When less-still unlock counter equals unlock value, less-still will be unactive
7-1	LESS_STILL_NOISE[6:0]	**Less-still noise value bit [6:0]** The noise value for less-still detection.

DEINTERLACER 14 — REG 2E, R/W

Bit	7	6	5	4	3	2	1	0
	HFREQ_UNLOCK[0]	HFREQ_LOCK				HFREQ_ID	HFREQ_DET_EN	LESS_STILL_NOISE[7]

Bit	Name	Function
0	LESS_STILL_NOISE[7]	**Less-still noise value bit [7]** The noise value for less-still detection.
1	HFREQ_DET_EN	**High-frequency detection enable** When set to 1, high-frequency detection is in auto mode. When set to 0, high-frequency detection is in manual mode.
2	HFREQ_ID	**High-frequency indicator by user (in manual mode)** High-frequency indicator by user, only useful in HFREQ_DET_EN =0
6-3	HFREQ_LOCK	**High-frequency auto lock value** When high-frequency lock counter equals lock value, high-frequency will be active
7	HFREQ_UNLOCK[0]	**High-frequency auto unlock value bit [0]** When high-frequency unlock counter equals unlock value, high-frequency will be inactive

DEINTERLACER 15 — REG 2F, R/W

Bit	7	6	5	4	3	2	1	0
	HFREQ_NOISE[5:0]						HFREQ_UNLOCK[2:1]	

Bit	Name	Function
1-0	HFREQ_UNLOCK[2:1]	**High-frequency auto unlock value bit [2:1]** When high-frequency unlock counter equals unlock value, high-frequency will be inactive.
7-2	HFREQ_NOISE[5:0]	**High-frequency detection noise value bit [5:0]** The noise value for high-frequency detection.

DEINTERLACER 16 — REG 30, R/W

Bit	7	6	5	4	3	2	1	0
	Reserved	MADPT_TEST_SEL					HFREQ_NOISE[7:6]	

Bit	Name	Function
1-0	HFREQ_NOISE[7:6]	**High-frequency detection noise value bit [7:6]** The noise value for high-frequency detection.
6-2	MADPT_TEST_SEL	**Motion Adaptive Test Bus Control** Internal hardware debugging use only.
7	Reserved	

DEINTERLACER 17 — REG 31, R/W

Bit	7	6	5	4	3	2	1	0
	GM_NOISE[5:0]						DIAG_MIN_CBYPS	DIAG_YTAP3_BYPS

Bit	Name	Function
0	DIAG_YTAP3_BYPS	**Diagonal BOB Y TAP3 Filter Bypass Control** When set to 1, bypass the diagonal BOB interpolate filter.
1	DIAG_MIN_CBYPS	**Diagonal BOB UV Bypass control** When set to 1, bypass diagonal min UV selection. No diagonal detection, just vertically two pixels UV average.
7-2	GM_NOISE[5:0]	In global motion noise manual mode, global motion detection noise bit [5:0] In global motion noise auto-detect mode, global motion noise's offset bit [5:0]

DEINTERLACER 18 **REG 32, R/W**

Bit	7	6	5	4	3	2	1	0
	SEL_22	GM_NOISE_EST_EN	GM_NOISE_GAIN				GM_NOISE[7:6]	

Bit	Name	Function
1-0	GM_NOISE[7:6]	In global motion noise manual mode, global motion detection noise bit [7:6] In global motion noise auto-detect mode, global motion noise's offset bit [7:6]
5-1	GM_NOISE_GAIN	**Global motion noise gain (in auto-detect mode)** Global motion noise gain in noise auto-detect mode
6	GM_NOISE_EST_EN	**Global motion noise auto detection enable** When set to 1, global motion noise is in auto-detect mode.
7	SEL_22	**2:2 pull-down selection** When set to 1, enable 2:2 pull-down detection When set to 0, enable 3:2 pull-down detection.

DEINTERLACER 19 — REG 33, R/W

	7	6	5	4	3	2	1	0
Bit	32PULLDOWN_LOCK[2:0]			32PULLDOWN_OFST			32PULLDOWN_ID	32PULLDOWN_EN
Bit	22PULLDOWN_THRESHOLD[1:0]		22PULLDOWN_DET			22PULLDOWN_OFST	22PULLDOWN_ID	22PULLDOWN_EN

Bit	Name	Function (in SEL_22 = 0, 3:2 pull-down detection is selected)
0	32PULLDOWN_EN	**3:2 pull-down detection enable** When set to 1, 3:2 pull-down detection is in auto mode. When set to 0, 3:2 pull-down detection is in manual mode.
1	32PULLDOWN_ID	**3:2 pull-down indicator defined by user (in manual mode)** 3:2 pull-down indicator by user, only useful in 32PULLDOWN_EN =0
4-2	32PULLDOWN_OFST	**3:2 pull-down sequence offset** 3:2 pull-down sequence offset
7-5	32PULLDOWN_LOCK[2:0]	**3:2 pull-down auto lock value bit [2:0]** When lock counter equals lock value, 3:2 pull-down is in active.

Bit	Name	Function (in SEL_22 = 1, 2:2 pull-down detection is selected)
0	22PULLDOWN_EN	**2:2 pull-down detection enable** When set to 1, 2:2 pull-down detection is in auto mode. When set to 0, 2:2 pull-down detection is in manual mode.
1	22PULLDOWN_ID	**2:2 pull-down indicator defined by user (in manual mode)** 2:2 pull-down indicator by user, only useful in 22PULLDOWN_EN =0
2	22PULLDOWN_OFST	**2:2 pull-down sequence offset** 2:2 pull-down sequence offset
5-3	22PULLDOWN_DET	**2:2 pull-down detection control bit** 2:2 pull-down accumulation result control
7-6	22PULLDOWN_THRESHOLD[1:0]	**2:2 pull-down detection threshold bit [1:0]** 2:2 pull-down detection threshold bit [1:0]

DEINTERLACER 20 — REG 34, R/W

	7	6	5	4	3	2	1	0
Bit	Reserved				32PULLDOWN_LOCK[6:3]			
Bit	22PULLDOWN_THRESHOLD[9:2]							

Bit	Name	Function (in SEL_22 =0, 3:2 pull-down detection is selected)
3-0	32PULLDOWN_LOCK [6:3]	**3:2 pull-down auto lock value bit [6:3]** When lock counter equals lock value, 3:2 pull-down is in active.
7-4	Reserved	

Bit	Name	Function (in SEL_22 = 1, 2:2 pull-down detection is selected)
7-0	22PULLDOWN_THRESHOLD [9:2]	**2:2 pull-down detection threshold bit [9:2]** 2:2 pull-down detection threshold bit [9:2]

DEINTERLACER 21 — REG 35, R/W

	7	6	5	4	3	2	1	0
Bit	Reserved							
Bit	22PULLDOWN_THRESHOLD[17:10]							

Bit	Name	Function (in SEL_22 =0, 3:2 pull-down detection is selected)
7-0	Reserved	

Bit	Name	Function (in SEL_22 = 1, 2:2 pull-down detection is selected)
7-0	22PULLDOWN_THRESHOLD [9:2]	**2:2 pull-down detection threshold bit [9:2]** 2:2 pull-down detection threshold bit [9:2]

DEINTERLACER 22 — REG 36, R/W

Bit	7	6	5	4	3	2	1	0
	VSCALE_IIR_COEF[2:0]			VSCALE_IIR_ROUND_SEL	VSCALE_IIR_BYPS	VSCALE_PHASE_INI	UV_VSCALE_BYPS	Y_VSCALE_BYPS

Bit	Name	Function
0	Y_VSCALE_BYPS	**Bypass Y phase adjustment in vertical scaling down** When set to 1, Y phase adjustment in vertical scaling down will be bypass
1	UV_VSCALE_BYPS	**Bypass UV phase adjustment in vertical scaling down** When set to 1, UV phase adjustment in vertical scaling down will be bypass
2	VSCALE_PHASE_INI	**Vertical scaling down initial phase selection**
3	VSCALE_IIR_BYPS	**Bypass V-IIR filter in vertical scaling down** When set to 1, V-IIR filter in vertical scaling down will be bypass.
4	VSCALE_IIR_ROUND_SEL	**V-IIR filter in vertical scaling down round selection**
7-5	VSCALE_IIR_COEF[2:0]	**V-IIR filter coefficient bit [2:0]**

DEINTERLACER 23 — REG 37, R/W

Bit	7	6	5	4	3	2	1	0
	VSCALE_RATE[3:0]				VSCALE_IIR_COEF[6:3]			

Bit	Name	Function
3-0	VSCALE_IIR_COEF[6:3]	**V-IIR filter coefficient bit [6:3]**
7-4	VSCALE_RATE[3:0]	**Vertical scaling down rate bit [3:0]** If M line down to N line, it should be (M-N)/N*1024

DEINTERLACER 24 — REG 38, R/W

Bit	7	6	5	4	3	2	1	0
	MI_1BIT_DELAY		VSCALE_RATE[9:4]					

Bit	Name	Function
5-0	VSCALE_RATE[9:4]	**Vertical scaling down rate bit [9:4]** If M line down to N line, it should be (M-N)/N*1024
7-6	MI_1BIT_DELAY	**Delay pipe control for motion index feedback-bit**

MI_1BIT_DELAY[1:0]	MI feedback-bit delay pipes
00	0
01	1
10	2
11	3

DEINTERLACER 25 — REG 39, R/W

Bit	7	6	5	4	3	2	1	0
	MI_1BIT_THRESHOLD[5:0]						MI_1BIT_FRAME2_EN	MI_1BIT_BYPS

Bit	Name	Function
0	MI_1BIT_BYPS	**Motion index feedback-bit bypass** When set to 1, motion index feedback-bit function will be bypass
1	MI_1BIT_FRAME2_EN	**Enable Frame-two feedback-bit** When set to 1, enable frame-two feedback-bit.
7-2	MI_1BIT_THRESHOLD [5:0]	**Motion index feedback-bit generation's threshold bit [5:0]**

DEINTERLACER 26 — REG 3A, R/W

Bit	7	6	5	4	3	2	1	0
	MI_1BIT_FIX_VALUE							MI_1BIT_THRESHOLD[6]

Bit	Name	Function
0	MI_1BIT_THRESHOLD [0]	**Motion index feedback-bit generation's threshold bit [5:0]**
7-1	MI_1BIT_FIX_VALUE	**Motion index fixed value**

DEINTERLACER 27 — REG 3B, R/W

Bit	7	6	5	4	3	2	1	0
	PIXEL_STILL_THRESHOLD_1							

Bit	Name	Function
7-0	PIXEL_STILL_THRESHOLD_1	**Pixel base still threshold level one**

DEINTERLACER 28 — REG 3C, R/W

Bit	7	6	5	4	3	2	1	0
	PIXEL_STILL_THRESHOLD_2							

Bit	Name	Function
7-0	PIXEL_STILL_THRESHOLD_2	**Pixel base still threshold level two**

DEINTERLACER 29 — REG 3D, R/W

Bit	7	6	5	4	3	2	1	0
	NOISE_DET_THRESHOLD[1:0]		NOISE_DET_RST		NOISE_DET_SHIFT		NOISE_DET_SEL	PIXEL_STILL_EN

Bit	Name	Function
0	PIXEL_STILL_EN	**Enable pixel base still** When set to 1, pixel base still function will be enable.
1	NOISE_DET_SEL	**Noise detection selection** When set to 1, noise detection is in video active period. When set to 0, noise detection is in video blanking period.
3-2	NOISE_DET_SHIFT	**Noise detection shift** When set to 3, noise detection drop 15bits When set to 2, noise detection drop 16bits When set to 1, noise detection drop 17bits When set to 0, noise detection drop 18bits
5-4	NOISE_DET_RST	**Noise detection reset control**
7-6	NOISE_DET_THRESHOLD [1:0]	**Noise detection threshold bit [1:0]** Threshold for NOUT signal.

DEINTERLACER 30 — REG 3E, R/W

Bit	7	6	5	4	3	2	1	0
	ENABLE_NOUT_NRD	ENABLE_NOUT_LESS_STILL	ENABLE_NOUT_STILL	NOISE_DET_THRESHOLD[6:2]				

Bit	Name	Function
4-0	NOISE_DET_THRESHOLD [6:2]	**Noise detection threshold bit [6:2]** Threshold for NOUT signal.
5	ENABLE_NOUT_STILL	**Enable NOUT for still detection**
6	ENABLT_NOUT_LESS_STILL	**Enable NOUT for less-still detection**
7	ENABLE_NOUT_NRD	**Enable NOUT for background noise reduction**

DEINTERLACER 31 — REG 3F, R/W

Bit	7	6	5	4	3	2	1	0
	Y_DELAY[0]	NOISE_THRESHOLD_VDS						

Bit	Name	Function
6-0	NOISE_THRESHOLD_VDS	**Auto noise threshold for nout_vds_proc**
7	Y_DELAY [0]	**Y delay pipe control bit [0]**

DEINTERLACER 32 — REG 70, R/W

Bit	7	6	5	4	3	2	1	0
	PD_SP[2:0]			UV_DELAY			Y_DELAY[2:1]	

Bit	Name	Function
1-0	Y_DELAY[2:1]	**Y delay pipe control bit [2:1]** Y_DELAY[2:0] / Y data delay pipes 000 / 0 001 / 1 010 / 2 011 / 3 100 / 4 101 / 5 110 / 6 111 / 7
4-2	UV_DELAY	**UV delay pipe control** UV_DELAY[2:0] / UV data delay pipes 000 / 0 001 / 1 010 / 2 011 / 3 100 / 4 101 / 5 110 / 6 111 / 7
7-5	PD_SP[2:0]	**Scaling down line buffer WRSTZ position adjustment bit [2:0]**

DEINTERLACER 33 — REG 71, R/W

Bit	7	6	5	4	3	2	1	0
	PD_RAM_BYPS	PD_ST					PD_SP[4:3]	

Bit	Name	Function
1-0	PD_SP[4:3]	**Scaling down line buffer WRSTZ position adjustment bit [4:3]**
6-2	PD_ST	**Scaling down line buffer RRSTZ position adjustment**
7	PD_RAM_BYPS	**Bypass scaling down's line buffer** When set to 1, scaling down's line buffer will be bypass.

DEINTERLACER 34 — REG 72, R/W

Bit	7	6	5	4	3	2	1	0
	NRD_SEL	Y_ VTAP_CNTRL			Y_HTAP_CNTRL			

Bit	Name	Function
3-0	Y_HTAP_CNTRL	**Y horizontal filter control for background reduction** Y_HTAP_CNTRL[3:0] could bypass four tap3 FIR filter.
6-4	Y_VTAP_CNTRL	**Y vertical filter control for background reduction** Y_VTAP_CNTRL[0]: when set to1, bypass vertical filter Y_VTAP_CNTRL[1]: when set to 1, enable FIR filter Y_VTAP_CNTRL[2]: when set to 1, bypass IIR filter
7	NRD_SEL	**Background reduction selection control** Only set it to 1 in huge noise condition

DEINTERLACER 35 — REG 73, R/W

Bit	7	6	5	4	3	2	1	0
	Y_NRD_EN	M_VTAP_CNTRL			M_HTAP_CNTRL			

Bit	Name	Function
3-0	M_HTAP_CNTRL	**Background noise reduction H filter control in huge noise condition** M_HTAP_CNTRL[3:0] could bypass four tap3 FIR filter.
6-4	M_VTAP_CNTRL	**Background noise reduction V filter control in huge noise condition** M_VTAP_CNTRL[0]: when set to1, bypass vertical filter M_VTAP_CNTRL[1]: when set to 1, enable FIR filter M_VTAP_CNTRL[2]: when set to 1, bypass IIR filter
7	Y_NRD_EN	**Enable background noise reduction in Y domain** When set to 1, enable background noise reduction in Y domain.

DEINTERLACER 36 — REG 74, R/W

Bit	7	6	5	4	3	2	1	0
	UV_WOUT[1:0]		UV_WOUT_BYPS	Y_WOUT			Y_WOUT_BYPS	UV_NRD_EN

Bit	Name	Function
0	UV_NRD_EN	**Enable background noise reduction in UV domain** When set to 1, enable background noise reduction in UV domain.
1	Y_WOUT_BYPS	**Bypass Y WOUT**
4-2	Y_WOUT	**Coefficient for Y noise reduction**
5	UV_WOUT_BYPS	**Bypass UV WOUT**
7-6	UV_WOUT[1:0]	**Coefficient for UV noise reduction bit [1:0]**

DEINTERLACER 37 — REG 75, R/W

Bit	7	6	5	4	3	2	1	0
	LO_CMP_CNTRL[4:0]					CMP_ID	CMP_EN	UV_WOUT[2]

Bit	Name	Function
0	UV_WOUT[2]	**Coefficient for UV noise reduction bit [2]**
1	CMP_EN	**Motion compare enable** When set to 1, enable motion compare When set to 0, motion compare is in manual mode
2	CMP_ID	**Motion compare result defined by user (in manual mode)** Motion compare result defined by user when CMP_EN = 0
7-3	LO_CMP_CNTRL [4:0]	**Motion compare low level threshold bit [4:0]**

DEINTERLACER 38 — REG 76, R/W

Bit	7	6	5	4	3	2	1	0
	HI_CMP_CNTRL[4:0]					LO_CMP_CNTRL[7:5]		

Bit	Name	Function
2-0	LO_CMP_CNTRL[7:5]	**Motion compare low level threshold bit [7:5]**
7-3	HI_CMP_CNTRL[4:0]	**Motion compare high level threshold bit [4:0]**

DEINTERLACER 39 — REG 78, R/W

Bit	7	6	5	4	3	2	1	0
	NRD_VIIR_PD_ST[0]	NRD_VIIR_PD_SP					HI_CMP_CNTRL[7:5]	

Bit	Name	Function
2-0	HI_CMP_TNTRL [7:5]	Motion compare high level threshold bit [7:5]
6-3	NRD_VIIR_PD_SP	NRD line buffer WRSTZ position adjustment
7	NRD_VIIR_PD_ST[0]	NRD line buffer RRSTZ position adjustment bit [0]

DEINTERLACER 40 — REG 79, R/W

Bit	7	6	5	4	3	2	1	0
	UVDLY_PD_SP				NRD_VIIR_PD_BYPS	NRD_VIIR_PD_ST[3:1]		

Bit	Name	Function
2-0	NRD_VIIR_PD_ST[3:1]	NRD line buffer RRSTZ position adjustment bit [3:1]
3	NRD_VIIR_PD_BYPS	Bypass NRD line buffer
7-4	UVDLY_PD_SP	UV delay line buffer WRSTZ position adjustment

DEINTERLACER 41 **REG 7A, R/W**

Bit	7	6	5	4	3	2	1	0
	EN_PULLDOWN_FOR_NRD	DD0_SEL	NRD_OUT_SEL	UVDLY_PD_BYPS	UVDLY_PD_ST			

Bit	Name	Function
3-0	UVDLY_PD_ST	**UV delay line buffer RRSTZ position adjustment**
4	UVDLY_PD_BYPS	**Bypass UV delay line buffer**
5	NRD_OUT_SEL	**NRD output selection** Only set it to 1 in huge noise condition.
6	DD0_SEL	Set it to 1 when background noise reduction enable Set it to 0 when background noise reduction disable
7	EN_PULLDOWN_FOR_NRD	**Enable pull-down to block STILL for NRD** Set it to 1, background noise reduction will in low noise level when in 32/22 pull-down source.

HD PRODUCTION REGISTER

HD_PROC NORMAL MODE REGISTERS

HD_PROC 00 **REG 77, R/W**

Bit	7	6	5	4	3	2	1	0
	HSCALE0 [1:0]		SEL_HD_TIME	IN_DREG_BYPS	MATRIX_BYPS	SEL_HD_BYPS	SEL_HD_MODE_DET	SEL_HD

Bit	Name	Function
0	SEL_HD	**Select HD Processing Engine Normal Mode** When set to 1, HD processing engine normal mode will be selected.
1	Reserved	
2	SEL_HD_BYPS	**Select HD Processing Engine Bypass Mode** When set to 1, HD processing engine bypass mode will be selected.
3	MATRIX_BYPS	**YUV2RGB and RGB2YUV Convert Matrix Bypass Control** When set to 1, YUV2RGB and RGB2YUV convert function will be bypassed.
4	IN_DREG_BYPS	**Control Input Signals Using Clock Falling Edge:** Use clock falling edge to catch the input data and timing
5	SEL_HD_TIME	**Select HD Processing Engine timing as vds_proc Input Timing** When set to 1, HD processing engine timing will be sent to vds_proc as input timing.
7-6	HSCALE0 [1:0]	**Horizontal Non-linear Scaling Down First Segment DDA Increment [1:0] (total 6 bits)** Assume the scaling ratio is n/m, then the value should be (n/m) x63

HD_PROC 01 **REG 78, R/W**

Bit	7	6	5	4	3	2	1	0
	HSCALE1 [3:0]				HSCALE0 [5:2]			

Bit	Name	Function
3-0	HSCALE0 [5:2]	**Horizontal Non-linear Scaling Down First Segment DDA Increment [5:2] (total 6 bits)**
7-4	HSCALE1 [3:0]	**Horizontal Non-linear Scaling Down Second Segment DDA Increment [3:0] (total 6 bits)**

HD_PROC 02 — REG 79, R/W

Bit	7	6	5	4	3	2	1	0
	HSCALE2 [5:0]						HSCALE1 [5:4]	

Bit	Name	Function
1-0	HSCALE1 [5:4]	Horizontal Non-linear Scaling Down Second Segment DDA Increment [5:4] (total 6 bits)
7-2	HSCALE2 [5:0]	Horizontal Non-linear Scaling Down Third Segment DDA Increment

HD_PROC 03 — REG 7A, R/W

Bit	7	6	5	4	3	2	1	0
	HSCALE4 [1:0]		HSCALE3 [5:0]					

Bit	Name	Function
5-0	HSCALE3 [5:0]	Horizontal Non-linear Scaling Down Fourth Segment DDA Increment
7-6	HSCALE4 [1:0]	Horizontal Non-linear Scaling Down Fifth Segment DDA Increment [1:0] (total 6 bits)

HD_PROC 04 — REG 7B, R/W

Bit	7	6	5	4	3	2	1	0
	HSCALE5 [3:0]				HSCALE4 [5:2]			

Bit	Name	Function
3-0	HSCALE4 [5:2]	Horizontal Non-linear Scaling Down Fifth Segment DDA Increment [5:2] (total 6 bits)
7-4	HSCALE5 [3:0]	Horizontal Non-linear Scaling Down Sixth Segment DDA Increment [3:0] (total 6 bits)

HD_PROC 05 — REG 20, R/W

Bit	7	6	5	4	3	2	1	0
	HSCALE6 [5:0]						HSCALE5 [5:4]	

Bit	Name	Function
1-0	HSCALE5 [5:4]	**Horizontal Non-linear Scaling Down Sixth Segment DDA Increment [5:4] (total 6 bits)**
7-2	HSCALE6 [5:0]	**Horizontal Non-linear Scaling Down Seventh Segment DDA Increment**

HD_PROC 06 — REG 21, R/W

Bit	7	6	5	4	3	2	1	0
	HS_PSHIFT_BYPS	HS_DEC_BYPS	HSCALE7 [5:0]					

Bit	Name	Function
5-0	HSCALE7 [5:0]	**Horizontal Non-linear Scaling Down Eighth Segment DDA Increment**
6	HS_DEC_BYPS	**Horizontal Y/U/V Low Pass Filter Bypass Control** When set to 1, horizontal Tap7 Y, Tap3 U&V low pass filter will be bypassed.
7	HS_PSHIFT_BYPS	**Horizontal Phase Adjustment Bypass Control** When set to 1, horizontal phase adjustment for Y, U and V will be bypassed.

HD_PROC 07 — REG 22, R/W

Bit	7	6	5	4	3	2	1	0
	SEL_PROV	UV_DELAY [1:0]		Y_DELAY [1:0]		Reserved	DEINT_LD_RAM_BYPS	Reserved

Bit	Name	Function
0	RESERVED	
1	DEINT_LD_RAM_BYPS	**Deint Line Double Bypass Control** When set to 1, bypass FIFO for deint line double
2	RESERVED	
4-3	Y_DELAY [1:0]	**Y Pipes Delay Control With Enable** Delay pipes Table:

Y_DELAY [1]	Y_DELAY [0]	Delay Pipes Number
0	0	1
0	1	2
1	0	3
1	1	4

Bit	Name	Function
6-5	UV_DELAY [1:0]	**UV Pipes Delay Control With Enable** Delay pipes Table:

UV_DELAY [1]	UV_DELAY [0]	Delay Pipes Number
0	0	1
0	1	2
1	0	3
1	1	4

Bit	Name	Function
7	SEL_PROV	**Select Progressive Line Reset** When set to 1, select progressive line reset. Else select interlaced line reset.

HD_PROC 08 — REG 23, R/W

Bit	7	6	5	4	3	2	1	0
	DEINT_PD_SP [6:0]							SEL_WEN

Bit	Name	Function
0	SEL_WEN	**Select Progressive Write Enable** When set to 1, select progressive write enable.
7-1	DEINT_PD_SP [6:0]	**The Position of Write Reset Control For Deint Pdelayer [6:0]** Adjust the position of write reset in line buffer for deint pdelayer

HD_PROC 09 — REG 24, R/W

Bit	7	6	5	4	3	2	1	0
	VS_PD_SP [0]	VSCALE_BYPS	DEINT_BYPS	DEINT_PD_RAM_BYPS	Reserved			DEINT_PD_SP [8:7]

Bit	Name	Function
1-0	DEINT_PD_SP [8:7]	**The Position of Write Reset Control For Deint Pdelayer [8:7](total 9 bits)**
3-2	Reserved	
4	DEINT_PD_RAM_BYPS	**Deint Pdelayer Bypass Control** When set to 1, bypass FIFO for deint pdelayer
5	DEINT_BYPS	**Deint Bypass control** When set to 1, bypass Deinterlace function.
6	VSCALE_BYPS	**Vertical Non-linear Scaling Down Bypass Control** When set to 1, bypass the vertical non-linear scaling down.
7	VS_PD_SP [0]	**The Position of Write Reset Control For Vertical Scaling Pdelayer[0]** Adjust the position of write reset in line buffer for vertical scaling down pdelayer

HD_PROC 10 — REG 25, R/W

Bit	7	6	5	4	3	2	1	0
	VS_PD_SP [8:1]							

Bit	Name	Function
7-0	VS_PD_SP [8:1]	**The Position of Write Reset Control For Vertical Scaling Pdelayer [8:1] (total 10 bits)**

HD_PROC 11 — REG 26, R/W

Bit	7	6	5	4	3	2	1	0
	VSCALE0 [3:0]				VS_PD_RAM_BYPS			VS_PD_SP [9]

Bit	Name	Function
0	VS_PD_SP [9]	The Position of Write Reset Control For Vertical Scaling Pdelayer [9] (total 10 bits)
2-1	Reserved	
3	VS_PD_RAM_BYPS	Vertical Scaling Pdelayer Bypass Control When set to 1, bypass FIFO for vertical scaling pdelayer
7-4	VSCALE0 [3:0]	Vertical Non-linear Scaling Down First Segment DDA Increment[3:0] (total 6 bits) Assume the vertical scaling ratio is n/m, the value is (n/m) x63

HD_PROC 12 — REG 27, R/W

Bit	7	6	5	4	3	2	1	0
	VSCALE1 [5:0]						VSCALE0 [5:4]	

Bit	Name	Function
1-0	VSCALE0 [5:4]	Vertical Non-linear Scaling Down First Segment DDA Increment [5:4] (total 6 bits)
7-2	VSCALE1 [5:0]	Vertical Non-linear Scaling Down Second Segment DDA Increment

HD_PROC 13 — REG 28, R/W

Bit	7	6	5	4	3	2	1	0
	VSCALE3 [1:0]		VSCALE2 [5:0]					

Bit	Name	Function
5-0	VSCALE2 [5:0]	Vertical Non-linear Scaling Down Third Segment DDA Increment
7-6	VSCALE3 [1:0]	Vertical Non-linear Scaling Down Fourth Segment DDA Increment [1:0] (total 6 bits)

HD_PROC 14 — REG 29, R/W

Bit	7	6	5	4	3	2	1	0
	VSCALE4 [3:0]				VSCALE3 [5:2]			

Bit	Name	Function
3-0	VSCALE3 [5:2]	Vertical Non-linear Scaling Down Fourth Segment DDA Increment [5:2] (total 6 bits)
7-4	VSCALE4 [3:0]	Vertical Non-linear Scaling Down Fifth Segment DDA Increment [3:0] (total 6 bits)

HD_PROC 15 — REG 2A, R/W

Bit	7	6	5	4	3	2	1	0
	VSCALE5 [5:0]						VSCALE4 [5:4]	

Bit	Name	Function
1-0	VSCALE4 [5:4]	Vertical Non-linear Scaling Down Fifth Segment DDA Increment [5:4] (total 6 bits)
7-2	VSCALE5 [5:0]	Vertical Non-linear Scaling Down Sixth Segment DDA Increment

HD_PROC 16 — REG 2B, R/W

Bit	7	6	5	4	3	2	1	0
	VSCALE7 [1:0]		VSCALE6 [5:0]					

Bit	Name	Function
5-0	VSCALE6 [5:0]	Vertical Non-linear Scaling Down Seventh Segment DDA Increment
7-6	VSCALE7 [1:0]	Vertical Non-linear Scaling Down Eighth Segment DDA Increment [1:0] (total 6 bits)

HD_PROC 17 — REG 2C, R/W

Bit	7	6	5	4	3	2	1	0
	HSYNC_RST [3:0]				VSCALE7 [5:2]			

Bit	Name	Function
3-0	VSCALE7 [5:2]	Vertical Non-linear Scaling Down Eighth Segment DDA Increment [5:2] (total 6 bits)
7-4	HSYNC_RST [3:0]	Sync Generation Horizontal Counter Reset Value [3:0] (total 11 bits) Reset the sync generation internal horizontal counter at half-line position

HD_PROC 18 — REG 2D, R/W

Bit	7	6	5	4	3	2	1	0
	INI_ST [0]	HSYNC_RST [10:4]						

Bit	Name	Function
6-0	HSYNC_RST [10:4]	Sync Generation Horizontal Counter Reset Value [10:4] (total 11 bits)
7	INI_ST [0]	Sync Generation Horizontal Reset Pulse Start Position [0] (total 11 bits) When the internal counter equals the defined value, the horizontal reset pulse will be high. (Internal ini_sp = ini_st + 1)

HD_PROC 19 — REG 2E, R/W

Bit	7	6	5	4	3	2	1	0
	INI_ST [8:1]							

Bit	Name	Function
7-0	INI_ST [8:1]	Sync Generation Horizontal Reset Pulse Start Position [8:1] (total 11 bits)

HD_PROC 20 — REG 2F, R/W

Bit	7	6	5	4	3	2	1	0
	HBOUT_ST [5:0]						INI_ST [10:9]	

Bit	Name	Function
1-0	INI_ST [10:9]	**Sync Generation Horizontal Reset Pulse Start Position [10:9] (total 11 bits)**
7-2	HBOUT_ST [5:0]	**Sync Generation Progressive Output Horizontal Sync Start Position [5:0] (total 11 bits)** When the internal counter equals the defined value, the output horizontal sync will be high

HD_PROC 21 — REG 30, R/W

Bit	7	6	5	4	3	2	1	0
	HBOUT_SP [2:0]			HBOUT_ST [10:6]				

Bit	Name	Function
4-0	HBOUT_ST [10:6]	**Sync Generation Progressive Output Horizontal Sync Start Position [10:6] (total 11 bits)**
7-5	HBOUT_SP [2:0]	**Sync Generation Progressive Output Horizontal Sync Stop Position [2:0] (total 11 bits)** When the internal counter equals the defined value, the output horizontal sync will be low

HD_PROC 22 — REG 31, R/W

Bit	7	6	5	4	3	2	1	0
	HBOUT_SP [10:3]							

Bit	Name	Function
7-0	HBOUT_SP [10:3]	**Sync Generation Progressive Output Horizontal Sync Stop Position [10:3] (total 11 bits)**

HD_PROC 23 — REG 32, R/W

Bit	7	6	5	4	3	2	1	0
	LINE_ST [7:0]							

Bit	Name	Function
7-0	LINE_ST [7:0]	**Sync Generation Line Start Position [7:0] (total 11 bits)** When the internal counter equals the defined value, the line signal will be high.

HD_PROC 24 — REG 33, R/W

Bit	7	6	5	4	3	2	1	0
	LINE_SP [4:0]					LINE_ST [10:8]		

Bit	Name	Function
2-0	LINE_ST [10:8]	**Sync Generation Line Start Position [10:8] (total 11 bits)**
7-3	LINE_SP [4:0]	**Sync Generation Line Stop Position [4:0] (total 11 bits)** When the internal counter equals the defined value, the line signal will be low.

HD_PROC 25 — REG 34, R/W

Bit	7	6	5	4	3	2	1	0
	HBIN_ST [1:0]		LINE_SP [10:5]					

Bit	Name	Function
5-0	LINE_SP [10:5]	**Sync Generation Line Stop Position [10:5] (total 11 bits)**
7-6	HBIN_ST [1:0]	**Sync Generation Internal Interlace Horizontal Sync Start Position [1:0] (total 12 bits)** When the internal counter equals the defined value, the internal interlace horizontal sync will be high.

HD_PROC 26 — REG 35, R/W

Bit	7	6	5	4	3	2	1	0
	HBIN_ST [9:2]							

Bit	Name	Function
7-0	HBIN_ST [9:2]	Sync Generation Internal Interlace Horizontal Sync Start Position [9:2] (total 12 bits)

HD_PROC 27 — REG 36, R/W

Bit	7	6	5	4	3	2	1	0
	HBIN_SP [5:0]						HBIN_ST [11:10]	

Bit	Name	Function
1-0	HBIN_ST [11:10]	Sync Generation Internal Interlace Horizontal Sync Start Position [11:10] (total 12 bits)
7-2	HBIN_SP [5:0]	Sync Generation Internal Interlace Horizontal Sync Stop Position [5:0] (total 12 bits) When the internal counter equals the defined value, the internal interlace horizontal sync will be low.

HD_PROC 28 — REG 37, R/W

Bit	7	6	5	4	3	2	1	0
	VBOUT_ST [1:0]		HBIN_SP [11:6]					

Bit	Name	Function
5-0	HBIN_SP [11:6]	Sync Generation Internal Interlace Horizontal Sync Stop Position [11:6] (total 12 bits)
7-6	VBOUT_ST [1:0]	Sync Generation Progressive Output Vertical Sync Start Position [1:0] (total 11 bits) When the internal counter equals the defined value, the output vertical sync will be high.

HD_PROC 29 — REG 38, R/W

Bit	7	6	5	4	3	2	1	0
	VBOUT_ST [9:2]							

Bit	Name	Function
7-0	VBOUT_ST [9:2]	Sync Generation Progressive Output Vertical Sync Start Position [9:2] (total 11 bits)

HD_PROC 30 — REG 39, R/W

Bit	7	6	5	4	3	2	1	0
	VBOUT_SP [6:0]							VBOUT_ST [10]

Bit	Name	Function
0	VBOUT_ST [10]	Sync Generation Progressive Output Vertical Sync Start Position [10] (total 11 bits)
7-1	VBOUT_SP [6:0]	Sync Generation Progressive Output Vertical Sync Stop Position [6:0] (total 11 bits) When the internal counter equals the defined value, the output vertical sync will be low.

HD_PROC 31 — REG 3A, R/W

Bit	7	6	5	4	3	2	1	0
	TIME_HSYNC_RST [3:0]				VBOUT_SP [10:7]			

Bit	Name	Function
3-0	VBOUT_SP [10:7]	Sync Generation Progressive Output Vertical Sync Stop Position [10:7] (total 11 bits)
7-4	TIME_HSYNC_RST [3:0]	Timing Generation Horizontal Counter Reset Value [3:0] (total 11 bits) In timing generation module, reset the horizontal counter at the desired position

HD_PROC 32 — REG 3B, R/W

Bit	7	6	5	4	3	2	1	0
	TIME_VSYNC_RST[0]	TIME_HSYNC_RST [10:4]						

Bit	Name	Function
6-0	TIME_HSYNC_RST [10:4]	**Timing Generation Horizontal Counter Reset Value [10:4] (total 11 bits)**
7	TIME_VSYNC_RST [0]	**Timing Generation Vertical Counter Reset Value [0] (total 11 bits)** In timing generation module, reset the vertical counter at the desired position.

HD_PROC 33 — REG 3C, R/W

Bit	7	6	5	4	3	2	1	0
	TIME_VSYNC_RST [8:1]							

Bit	Name	Function
7-0	TIME_VSYNC_RST [8:1]	**Timing Generation Vertical Counter Reset Value [8:1] (total 11 bits)**

HD_PROC 34 — REG 3D, R/W

Bit	7	6	5	4	3	2	1	0
	TEST_SEL [3:0]				TEST_EN	FID_FLIP	TIME_VSYNC_RST [10:9]	

Bit	Name	Function
1-0	TIME_VSYNC_RST [10:9]	**Timing Generation Vertical Counter Reset Value [10:9] (total 11 bits)**
2	FID_FLIP	**Timing Generation Field ID Flip Control** When set to 1, the filed ID will be inverted before output.
3	TEST_EN	**Test Bus Enable** When set to 1, enable test bus output
7-4	TEST_SEL [3:0]	**Test Bus selection** Select the test bus to output

HD_PROC BYPASS MODE SHARED REGISTERS

HD_PROC 00 — REG 77, R/W

Bit	7	6	5	4	3	2	1	0
	OUT_DREG_BYPS	DYN_BYPS	SEL_HD_TIME	IN_DREG_BYPS	MATRIX_BYPS	SEL_HD_BYPS	SEL_HD_MODE_DET	SEL_HD

Bit	Name	Function
0	SEL_HD	**Select HD Processing Engine Normal Mode** When set to 1, HD processing engine normal mode will be selected.
1	SEL_HD_MODE_DET	
2	SEL_HD_BYPS	**Select HD Processing Engine Bypass Mode** When set to 1, HD processing engine bypass mode will be selected.
3	MATRIX_BYPS	**YUV2RGB and RGB2YUV Convert Matrix Bypass Control** When set to 1, YUV2RGB and RGB2YUV convert function will be bypassed.
4	IN_DREG_BYPS	**Control Input Signals Using Clock Falling Edge:** Use clock falling edge to catch the input data and timing.
5	SEL_HD_TIME	**Select HD Processing Engine timing as vds_proc Input Timing** When set to 1, HD processing engine timing will be send to vds_proc as input timing.
6	DYN_BYPS	**Dynamic Range Bypass Control** When set to 1, Y/U/V dynamic range function will be bypassed.
7	OUT_DREG_BYPS	**Output RGB Data Clock Edge Control** When set to 1, the output R/G/B data will not be catch by clock falling edge; When set to 0, the output R/G/B data will be catch by clock falling edge.

HD_PROC 01 — REG 78, R/W

Bit	7	6	5	4	3	2	1	0
	Y_GAIN [7:0]							

Bit	Name	Function
7-0	Y_GAIN [7:0]	**Y Dynamic Range Gain Value** The gain value of Y dynamic range in bypass mode.

HD_PROC 02 — REG 79, R/W

Bit	7	6	5	4	3	2	1	0
	Y_OFFSET [7:0]							

Bit	Name	Function
7-0	Y_OFFSET [7:0]	**Y Dynamic Range Offset Value** The offset value of Y dynamic range in bypass mode.

HD_PROC 03 — REG 7A, R/W

Bit	7	6	5	4	3	2	1	0
	U_GAIN [7:0]							

Bit	Name	Function
7-0	U_GAIN [7:0]	**U Dynamic Range Gain Value** The gain value of U dynamic range in bypass mode.

HD_PROC 04 — REG 7B, R/W

Bit	7	6	5	4	3	2	1	0
	U_OFFSET [7:0]							

Bit	Name	Function
7-0	U_OFFSET [7:0]	**U Dynamic Range Offset Value** The offset value of U dynamic range in bypass mode.

HD_PROC 05 — REG 20, R/W

Bit	7	6	5	4	3	2	1	0
	V_GAIN [7:0]							

Bit	Name	Function
7-0	V_GAIN [7:0]	**V Dynamic Range Gain Value High** The gain value of V dynamic range in bypass mode.

HD_PROC 06 — REG 21, R/W

Bit	7	6	5	4	3	2	1	0
	V_OFFSET [7:0]							

Bit	Name	Function
7-0	V_OFFSET [7:0]	**V Dynamic Range Offset Value High** The offset value of V dynamic range in bypass mode.

HD_PROC 07 — REG 22, R/W

Bit	7	6	5	4	3	2	1	0
	BLK_DATA_GY [7:0]							

Bit	Name	Function
7-0	BLK_DATA_GY [7:0]	**Blank Insertion Data for GY** Force the output GY data in blank to the defined data.

HD_PROC 08 — REG 23, R/W

Bit	7	6	5	4	3	2	1	0
	BLK_DATA_BU [7:0]							

Bit	Name	Function
7-0	BLK_DATA_BU [7:0]	**Blank Insertion Data for BU** Force the output BU data in blank to the defined data.

HD_PROC 09 — REG 24, R/W

Bit	7	6	5	4	3	2	1	0
	BLK_DATA_RV [7:0]							

Bit	Name	Function
7-0	BLK_DATA_RV [7:0]	**Blank Insertion Data for RV** Force the output RV data in blank to the defined data.

HD_PROC 10 — REG 25, R/W

Bit	7	6	5	4	3	2	1	0
	HB_ST [7:0]							

Bit	Name	Function
7-0	HB_ST [7:0]	**Bypass Mode Output Inserted Horizontal Blank Start Position [7:0] (total 12 bits)** When the internal counter equals the defined value, the output horizontal blank will be high.

HD_PROC 11 — REG 26, R/W

Bit	7	6	5	4	3	2	1	0
	HB_SP [3:0]				HB_ST [11:8]			

Bit	Name	Function
3-0	HB_ST [11:8]	**Bypass Mode Output Inserted Horizontal Blank Start Position [11:8] (total 12 bits)**
7-4	HB_SP [3:0]	**Bypass Mode Output Inserted Horizontal Blank Stop Position [3:0] (total 12 bits)** When the internal counter equals the defined value, the output horizontal blank will be low.

HD_PROC 12 — REG 27, R/W

Bit	7	6	5	4	3	2	1	0
	HB_SP [11:4]							

Bit	Name	Function
7-0	HB_SP [11:4]	**Bypass Mode Output Inserted Horizontal Blank Stop Position [11:4] (total 12 bits)** When the internal counter equals the defined value, the output horizontal blank will be low.

HD_PROC 13 — REG 28, R/W

7	6	5	4	3	2	1	0

Bit	VB_ST [7:0]

Bit	Name	Function
7-0	VB_ST [7:0]	**Bypass Mode Output Inserted Vertical Blank Start Position [7:0] (total 11 bits)** When the internal counter equals the defined value, the output vertical blank will be high.

HD_PROC 14 — REG 29, R/W

Bit	7	6	5	4	3	2	1	0
	VB_SP [4:0]					VB_ST [10:8]		

Bit	Name	Function
2-0	VB_ST [10:8]	**Bypass Mode Output Inserted Vertical Blank Start Position [10:8] (total 11 bits)**
7-3	VB_SP [4:0]	**Bypass Mode Output Inserted Vertical Blank Stop Position [4:0] (total 11 bits)** When the internal counter equals the defined value, the output vertical blank will be low.

HD_PROC 15 — REG 2A, R/W

Bit	7	6	5	4	3	2	1	0
	Reserved		VB_SP [10:5]					

Bit	Name	Function
2-0	VB_SP [10:5]	**Bypass Mode Output Inserted Vertical Blank Stop Position [10:5] (total 11 bits)**
7-6	Reserved	

MISCELLANEOUS REGISTERS

MISCELLANEOUS 00 — REG 00, RO

Bit	7	6	5	4	3	2	1	0
	NC_STATUS [3:0]	NC_STATUS[2:0]			Reserved			LOCK

Bit	Name	Function
0	LOCK	**PLL lock status bit** PLL testing bit When = 1, in lock status When = 0, in unlock status
3-1	Reserved	
6-4	NC_STATUS[2:0]	**Video port status** NC_STATUS[2:0] / Video port status: 000 — Floating 100 — 656 source connection 111 — 601 source connection other — ------
7	NC_STATUS[3:0]	**Auto detection status bit**

MISCELLANEOUS 01 — REG 05, RO

Bit	7	6	5	4	3	2	1	0
	IF_STATUS[7:0]							

Bit	Name	Function
7-0	IF_STATUS[7:0]	

MISCELLANEOUS 02 — REG 06, RO

Bit	7	6	5	4	3	2	1	0
	IF_STATUS[15:8]							

Bit	Name	Function
7-0	IF_STATUS[15:8]	

MISCELLANEOUS 03 — REG 07, RO

Bit	7	6	5	4	3	2	1	0
	IF_STATUS[23:16]							

Bit	Name	Function
7-0	IF_STATUS[23:16]	

MISCELLANEOUS 04 — REG 08, RO

Bit	7	6	5	4	3	2	1	0
	IF_STATUS[31:24]							

Bit	Name	Function
7-0	IF_STATUS[31:24]	

MISCELLANEOUS 05 — REG 09, RO

Bit	7	6	5	4	3	2	1	0
	IF_STATUS[39:32]							

Bit	Name	Function
7-0	IF_STATUS[39:32]	

MISCELLANEOUS 06 — REG 0A, RO

Bit	7	6	5	4	3	2	1	0
	Reserved	ARESETZ	ARESETZ_D	Reserved		GPIO_2_ST	GPIO_1_ST	GPIO_0_ST

Bit	Name	Function
0	GPIO_0_ST	**GPIO bit0 status** Please read the document of GPIO usage.
1	GPIO_1_ST	**GPIO bit1 status** Please read the document of GPIO usage.
2	GPIO_2_ST	**GPIO bit2 status** Please read the document of GPIO usage.
4-3	Reserved	
5	ARESETZ_D	**Power on delay reset status** When =1, power on delay reset is finished. When =0, in power on delay reset.
6	ARESETZ	**Power on reset status** When =1, power on reset is finished. When =0, in power on reset.
7	Reserved	**Reserved**

MISCELLANEOUS 07 — REG 0B, RO

Bit	7	6	5	4	3	2	1	0
	CHIP_ID_[7:0]							

Bit	Name	Function
7-0	CHIP_ID_[7:0]	**Chip ID low 8 bits** Reserved.

MISCELLANEOUS 08 — REG 0C, RO

Bit	7	6	5	4	3	2	1	0
	CHIP_ID_[15:8]							

Bit	Name	Function
7-0	CHIP_ID_[15:8]	**Chip ID middle 8 bits** Reserved.

MISCELLANEOUS 09 — REG 0D, RO

Bit	7	6	5	4	3	2	1	0
	CHIP_ID_[23:16]							

Bit	Name	Function
7-0	CHIP_ID_[23:16]	**Chip ID high 8 bits** Reserved.

MISCELLANEOUS 10 — REG 0E, RO

Bit	7	6	5	4	3	2	1	0
	MISC_STATUS[7:0]							

Bit	Name	Selection Control	Function
0	MISC_STATUS[0]	MISC_ST_CNTRL=000 (REG_4D[7:6], REG_4D[1])	**MEM_BA_INI_CTL** When =1, memory initial config is finished.
		MISC_ST_CNTRL=001 (REG_4D[7:6], REG_4D[1])	**WFF_EMPTY** When =1, Write FIFO is empty.
		MISC_ST_CNTRL=010 (REG_4D[7:6], REG_4D[1])	**IF_HPERIOD_[0]** Input source H-total bit[0]
		MISC_ST_CNTRL=011 (REG_4D[7:6], REG_4D[1])	**IF_VPERIOD_[0]** Input source V-total bit[0]
		MISC_ST_CNTRL=100 (REG_4D[7:6], REG_4D[1])	**TEST_BUS_[0]** Test bus bit[0]
		MISC_ST_CNTRL=other (REG_4D[7:6], REG_4D[1])	**Reserved**
1	MISC_STATUS[1]	MISC_ST_CNTRL=000 (REG_4D[7:6], REG_4D[1])	**DE32LOCK** When = 1, input source is 3:2 pull-down pattern.
		MISC_ST_CNTRL=001 (REG_4D[7:6], REG_4D[1])	**WFF_FULL** When =1, Write FIFO is full.
		MISC_ST_CNTRL=010 (REG_4D[7:6], REG_4D[1])	**IF_HPERIOD_[1]** Input source H-total bit[1]
		MISC_ST_CNTRL=011 (REG_4D[7:6], REG_4D[1])	**IF_VPERIOD_[1]** Input source V-total bit[1]
		MISC_ST_CNTRL=100 (REG_4D[7:6], REG_4D[1])	**TEST_BUS_[1]** Test bus bit[1]
		MISC_ST_CNTRL=other (REG_4D[7:6], REG_4D[1])	**Reserved**
2	MISC_STATUS[2]	MISC_ST_CNTRL=000 (REG_4D[7:6], REG_4D[1])	**D_VBOUT** When =1, in display vertical blanking.
		MISC_ST_CNTRL=001 (REG_4D[7:6], REG_4D[1])	**CAP_EMPTY** When =1, capture is empty.

Bit	Name	Selection Control	Function
		MISC_ST_CNTRL=010 (REG_4D[7:6], REG_4D[1])	**IF_HPERIOD_[2]** Input source H-total bit[2]
		MISC_ST_CNTRL=011 (REG_4D[7:6], REG_4D[1])	**IF_VPERIOD_[2]** Input source V-total bit[2]
		MISC_ST_CNTRL=100 (REG_4D[7:6], REG_4D[1])	**TEST_BUS_[2]** Test bus bit[2]
		MISC_ST_CNTRL=other (REG_4D[7:6], REG_4D[1])	**Reserved**

Bit	Name	Selection Control	Function
3	MISC_STATUS[3]	MISC_ST_CNTRL=000 (REG_4D[7:6], REG_4D[1])	**D_HBOUT** When =1, in display horizontal blanking
		MISC_ST_CNTRL=001 (REG_4D[7:6], REG_4D[1])	**CAP_FULL** When =1, Write FIFO is full.
		MISC_ST_CNTRL=010 (REG_4D[7:6], REG_4D[1])	**IF_HPERIOD_[3]** Input source H-total bit[3]
		MISC_ST_CNTRL=011 (REG_4D[7:6], REG_4D[1])	**IF_VPERIOD_[3]** Input source V-total bit[3]
		MISC_ST_CNTRL=100 (REG_4D[7:6], REG_4D[1])	**TEST_BUS_[3]** Test bus bit[3]
		MISC_ST_CNTRL=other (REG_4D[7:6], REG_4D[1])	**Reserved**
4	MISC_STATUS[4]	MISC_ST_CNTRL=000 (REG_4D[7:6], REG_4D[1])	**VSOUT_2PAD** When = 1, in display vertical sync.
		MISC_ST_CNTRL=001 (REG_4D[7:6], REG_4D[1])	**RFF_EMPTY** When =1, Read FIFO is empty.
		MISC_ST_CNTRL=010 (REG_4D[7:6], REG_4D[1])	**IF_HPERIOD_[4]** Input source H-total bit[4]
		MISC_ST_CNTRL=011 (REG_4D[7:6], REG_4D[1])	**IF_VPERIOD_[4]** Input source V-total bit[4]
		MISC_ST_CNTRL=100 (REG_4D[7:6], REG_4D[1])	**TEST_BUS_[4]** Test bus bit[4]
		MISC_ST_CNTRL=other (REG_4D[7:6], REG_4D[1])	**Reserved**
5	MISC_STATUS[5]	MISC_ST_CNTRL=000 (REG_4D[7:6], REG_4D[1])	**HSOUT_2PAD** When =1, in display horizontal sync.
		MISC_ST_CNTRL=001 (REG_4D[7:6], REG_4D[1])	**RFF_FULL** When =1, Read FIFO is full.

Bit	Name	Selection Control	Function
		MISC_ST_CNTRL=010 (REG_4D[7:6], REG_4D[1])	**IF_HPERIOD_[5]** Input source H-total bit[5]
		MISC_ST_CNTRL=011 (REG_4D[7:6], REG_4D[1])	**IF_VPERIOD_[5]** Input source V-total bit[5]
		MISC_ST_CNTRL=100 (REG_4D[7:6], REG_4D[1])	**TEST_BUS_[5]** Test bus bit[5]
		MISC_ST_CNTRL=other (REG_4D[7:6], REG_4D[1])	**Reserved**
6	MISC_STATUS[6]	MISC_ST_CNTRL=000 (REG_4D[7:6], REG_4D[1])	**VBOUT_3SET_[2]** When =1, in input source's vertical blanking
		MISC_ST_CNTRL=001 (REG_4D[7:6], REG_4D[1])	**PB_EMPTY** When =1, playback is empty.
		MISC_ST_CNTRL=010 (REG_4D[7:6], REG_4D[1])	**IF_HPERIOD_[6]** Input source H-total bit[6]
		MISC_ST_CNTRL=011 (REG_4D[7:6], REG_4D[1])	**IF_VPERIOD_[6]** Input source V-total bit[6]
		MISC_ST_CNTRL=100 (REG_4D[7:6], REG_4D[1])	**TEST_BUS_[6]** Test bus bit[6]
		MISC_ST_CNTRL=other (REG_4D[7:6], REG_4D[1])	**Reserved**
7	MISC_STATUS[7]	MISC_ST_CNTRL=000 (REG_4D[7:6], REG_4D[1])	**HBOUT_3SET_[2]** When = 1, in input source's horizontal blanking.
		MISC_ST_CNTRL=001 (REG_4D[7:6], REG_4D[1])	**PB_FULL** When =1, playback is full
		MISC_ST_CNTRL=010 (REG_4D[7:6], REG_4D[1])	**IF_HPERIOD_[7]** Input source H-total bit[7]
		MISC_ST_CNTRL=011 (REG_4D[7:6], REG_4D[1])	**IF_VPERIOD_[7]** Input source V-total bit[7]
		MISC_ST_CNTRL=100 (REG_4D[7:6], REG_4D[1])	**TEST_BUS_[7]** Test bus bit[7]
		MISC_ST_CNTRL=other (REG_4D[7:6], REG_4D[1])	**Reserved**

MISCELLANEOUS 11 **REG 0F, RO**

Bit	7	6	5	4	3	2	1	0
	MISC_STATUS[15:8]							

Bit	Name	Selection Control	Function
0	MISC_STATUS[8]	MISC_ST_CNTRL=000 (REG_4D[7:6], REG_4D[1])	FRAM_SYN_RST When =1, in frame's sync-lock reset
		MISC_ST_CNTRL=001 (REG_4D[7:6], REG_4D[1])	Reserved
		MISC_ST_CNTRL=010 (REG_4D[7:6], REG_4D[1])	IF_HPERIOD_[8] Input source H-total bit[8]
		MISC_ST_CNTRL=011 (REG_4D[7:6], REG_4D[1])	IF_VPERIOD_[8] Input source V-total bit[8]
		MISC_ST_CNTRL=100 (REG_4D[7:6], REG_4D[1])	TEST_BUS_[8] Test bus bit[8]
		MISC_ST_CNTRL=other (REG_4D[7:6], REG_4D[1])	Reserved
1	MISC_STATUS[9]	MISC_ST_CNTRL=000 (REG_4D[7:6], REG_4D[1])	D_VRENOUT When = 1, in playback's vertical line enable.
		MISC_ST_CNTRL=001 (REG_4D[7:6], REG_4D[1])	Reserved
		MISC_ST_CNTRL=010 (REG_4D[7:6], REG_4D[1])	Reserved
		MISC_ST_CNTRL=011 (REG_4D[7:6], REG_4D[1])	IF_VPERIOD_[9] Input source V-total bit[9]
		MISC_ST_CNTRL=100 (REG_4D[7:6], REG_4D[1])	TEST_BUS_[9] Test bus bit[9]
		MISC_ST_CNTRL=other (REG_4D[7:6], REG_4D[1])	Reserved
2	MISC_STATUS[10]	MISC_ST_CNTRL=000 (REG_4D[7:6], REG_4D[1])	D_HRENOUT When =1, in playback's horizontal pixel enable
		MISC_ST_CNTRL=001 (REG_4D[7:6], REG_4D[1])	Reserved
		MISC_ST_CNTRL=010 (REG_4D[7:6], REG_4D[1])	Reserved
		MISC_ST_CNTRL=011 (REG_4D[7:6], REG_4D[1])	IF_VPERIOD_[10] Input source V-total bit[10]
		MISC_ST_CNTRL=100 (REG_4D[7:6], REG_4D[1])	TEST_BUS_[10] Test bus bit[10]
		MISC_ST_CNTRL=other (REG_4D[7:6], REG_4D[1])	Reserved

Bit	Name	Selection Control	Function
3	MISC_STATUS[11]	MISC_ST_CNTRL=000 (REG_4D[7:6], REG_4D[1])	D_FDOUT When =1, in display top filed (interlace mode only).
		MISC_ST_CNTRL=001 (REG_4D[7:6], REG_4D[1])	Reserved
		MISC_ST_CNTRL=010 (REG_4D[7:6], REG_4D[1])	Reserved
		MISC_ST_CNTRL=011 (REG_4D[7:6], REG_4D[1])	Reserved
		MISC_ST_CNTRL=100 (REG_4D[7:6], REG_4D[1])	TEST_BUS_[11] Test bus bit[11]
		MISC_ST_CNTRL=other (REG_4D[7:6], REG_4D[1])	Reserved
4	MISC_STATUS[12]	MISC_ST_CNTRL=000 (REG_4D[7:6], REG_4D[1])	VBOUT_3SET_[1] When = 1, in write FIFO vertical blanking.
		MISC_ST_CNTRL=001 (REG_4D[7:6], REG_4D[1])	Reserved
		MISC_ST_CNTRL=010 (REG_4D[7:6], REG_4D[1])	Reserved
		MISC_ST_CNTRL=011 (REG_4D[7:6], REG_4D[1])	Reserved
		MISC_ST_CNTRL=100 (REG_4D[7:6], REG_4D[1])	TEST_BUS_[12] Test bus bit[12]
		MISC_ST_CNTRL=other (REG_4D[7:6], REG_4D[1])	Reserved
5	MISC_STATUS[13]	MISC_ST_CNTRL=000 (REG_4D[7:6], REG_4D[1])	HBOUT_3SET_[1] When =1, in write FIFO horizontal blanking.
		MISC_ST_CNTRL=001 (REG_4D[7:6], REG_4D[1])	Reserved
		MISC_ST_CNTRL=010 (REG_4D[7:6], REG_4D[1])	Reserved
		MISC_ST_CNTRL=011 (REG_4D[7:6], REG_4D[1])	Reserved
		MISC_ST_CNTRL=100 (REG_4D[7:6], REG_4D[1])	TEST_BUS_[13] Test bus bit[13]
		MISC_ST_CNTRL=other (REG_4D[7:6], REG_4D[1])	Reserved

Bit	Name	Selection Control	Function
6	MISC_STATUS[14]	MISC_ST_CNTRL=000 (REG_4D[7:6], REG_4D[1])	**FIDOUT**
		MISC_ST_CNTRL=001 (REG_4D[7:6], REG_4D[1])	**Reserved**
		MISC_ST_CNTRL=010 (REG_4D[7:6], REG_4D[1])	**Reserved**
		MISC_ST_CNTRL=011 (REG_4D[7:6], REG_4D[1])	**Reserved**
		MISC_ST_CNTRL=100 (REG_4D[7:6], REG_4D[1])	**TEST_BUS_[14]** Test bus bit[14]
		MISC_ST_CNTRL=other (REG_4D[7:6], REG_4D[1])	**Reserved**
7	MISC_STATUS[15]	MISC_ST_CNTRL=000 (REG_4D[7:6], REG_4D[1])	**LINEOUT** When = 1, in input original line period.
		MISC_ST_CNTRL=001 (REG_4D[7:6], REG_4D[1])	**Reserved**
		MISC_ST_CNTRL=010 (REG_4D[7:6], REG_4D[1])	**Reserved**
		MISC_ST_CNTRL=011 (REG_4D[7:6], REG_4D[1])	**Reserved**
		MISC_ST_CNTRL=100 (REG_4D[7:6], REG_4D[1])	**TEST_BUS_[15]** Test bus bit[15]
		MISC_ST_CNTRL=other (REG_4D[7:6], REG_4D[1])	**Reserved**

MISCELLANEOUS 12 — REG 40, R/W

Bit	7	6	5	4	3	2	1	0
	PLL_CNTRL [7:0]							

Bit	Name	Function
1-0	PLL_CNTRL[1:0]	**VS[1:0], VCLK/V2CLK output control** Please read PLL_CNTRL[13]'s description.
3-2	PLL_CNTRL[3:2]	**MS[1:0], MCLK output control (MS[2] is in 41/[6])**
4	PLL_CNTRL[4]	**VCORST** When set to 1, Reset VCO control voltage to about half of AVD
5	PLL_CNTRL[5]	**IS, ICLK output control** When set to 1, ICLK will use input video clock directly. When set to 0, ICLK will use PLL's output clock.
6	PLL_CNTRL[6]	**CKIS, PLL source clock selection** When set to 1, PLL source clock will use input video clock as source. When set to 0, PLL source clock will use OSC clock as source.
7	PLL_CNTRL[7]	**DIVBY2Z, PLL source clock frequency control** When set to 1, PLL source clock will not be divided. When set to 0, PLL source clock will be divided by 2.

MS[2]	MS[1]	MS[0]	MCLK
X	0	0	108Mhz
X	0	1	81Mhz
X	1	0	External memory clock
0	1	1	162Mhz
1	1	1	129.6Mhz

MISCELLANEOUS 13 **REG 41, R/W**

Bit	7	6	5	4	3	2	1	0
	PLL_CNTRL [15:8]							

Bit	Name	Function
1-0	PLL_CNTRL [9:8]	**R[1:0], skew control** Skew control for testing, default 00
3-2	PLL_CNTRL [11:10]	**S[1:0], skew control** Skew control for testing, default 00
4	PLL_CNTRL [12]	**LEN, lock enable** When =0, disable lock. When =1, enable lock.
5	PLL_CNTRL [13]	**BYPS, v2clk bypass control**

VS[2]	VS[1:0]	BYPS	VCLK	V2CLK
X	00	0	27MHz	54MHz
X	00	1	27MHz	27MHz
X	01	0	32.4MHz	64.8MHz
X	01	1	32.4MHz	32.4MHz
0	10	1	CLKIN	CLKIN
1	10	0	40.5MHz	81MHz
1	10	1	40.5MHz	40.5MHz
X	11	X	XTOUT	XTOUT

6	PLL_CNTRL[14]	**MS[2], MCLK output control** Please read MS[1:0] bit description
7	PLL_CNTRL[15]	**HDS** When =0, ICLK will be divided by two. When =1, ICLK will be not divided

MISCELLANEOUS 14 **REG 42, R/W**

Bit	7	6	5	4	3	2	1	0
	DAC_CNTRL[7:0]							

Bit	Name	Function
0	DAC_CNTRL[0]	**RSPWDN** When set to 1, whole DAC (include R, G, B, S) will power down
1	DAC_CNTRL[1]	**RPD** When set to 1, RDAC will power down
2	DAC_CNTRL[2]	**GPD** When set to 1, GDAC will power down
3	DAC_CNTRL[3]	**BPD** When set to 1, BDAC will power down
4	DAC_CNTRL[4]	**SPD** When set to 1, SDAC will power down
5	DAC_CNTRL[5]	**R0ENZ** When set to 1, RDAC output is based on input R When set to 0, RDAC output is MIN
6	DAC_CNTRL[6]	**R1EN** When set to 1, RDAC output is MAX When set to 0, RDAC output is based on input R
7	DAC_CNTRL[7]	**Reserved**

MISCELLANEOUS 15 **REG 43, R/W**

Bit	7	6	5	4	3	2	1	0
	CKT_FFCNTRL[1:0]		DAC_CNTRL[13:8]					

Bit	Name	Function
0	DAC_CNTRL[8]	**G0ENZ** When set to 1, GDAC output is based on input G. When set to 0, GDAC output is MIN.
1	DAC_CNTRL[9]	**G1EN** When set to 1, GDAC output is MAX. When set to 0, GDAC output is based on input G.
2	DAC_CNTRL[10]	**B0ENZ** When set to 1, BDAC output is based on input B. When set to 0, BDAC output is MIN.
3	DAC_CNTRL[11]	**B1EN** When set to 1, BDAC output is MAX. When set to 0, BDAC output is based on input B.
4	DAC_CNTRL[12]	**S0ENZ** When set to 1, SDAC output is based on input S. When set to 0, SDAC output is MIN.
5	DAC_CNTRL[13]	**S1EN** When set to 1, SDAC output is MAX. When set to 0, SDAC output is based on input S.
7-6	CKT_FFCNTRL[1:0]	**FIFO control bit used by CKT**

MISCELLANEOUS 16 — REG 44, R/W

Bit	7	6	5	4	3	2	1	0
	MISC_CNTRL[2:0]			TEST_OUT_CNTRL	ARESETZ_CNTRL	CORE_RST_CNTRL	FIFO_RST_CNTRL	MEM_RST_CNTRL

Bit	Name	Function
0	MEM_RST_CNTRL	**Memory controller reset** When set to 1, memory controller will be reset.
1	FIFO_RST_CNTRL	**FIFO reset** When set to 1, the FIFO will be reset.
2	CORE_RST_CNTRL	**CORE reset** When set to 1, the logic will be reset.
3	ARESETZ_CNTRL	**ARESETZ control** When set to 1, power on reset signal (ARESETZ, ARESETZ_D) will be set to one.
4	TEST_OUT_CNTRL	**Digital output control** When set to 1, 16bit digital output will be output. When set to 0, test bus low 16bit will be output.
5	MISC_CNTRL[0]	**V2CLK to DAC control** When set to 1, V2CLK to DAC will be inverted. When set to 0, V2CLK to DAC will be normal.
6	MISC_CNTRL[1]	**Output clock selection** When set to 1, pin93's display output clock is VCLK. When set to 0, pin93's display output clock is V2CLK.
7	MISC_CNTRL[2]	**Output clock invert** When set to 1, pin93(100pin package)'s display output clock will be inverted. When set to 0, pin93's display output clock will be normal.

MISCELLANEOUS 17 **REG 45, R/W**

Bit	7	6	5	4	3	2	1	0
	Reserved	PAD_CNTRL[6:0]						

Bit	Name	Function
0	PAD_CNTRL[0]	When set to 1, enable test_out_[7:0] output When set to 0, disable test_out_[7:0] output
1	PAD_CNTRL[1]	When set to 1, enable test_out_[[15:8] output When set to 0, disable test_out_[15:8] output
2	PAD_CNTRL[2]	When set to 1, enable test_out_[23:16] output When set to 0, disable test_out_[23:16] output
3	PAD_CNTRL[3]	When set to 1, disable sd_adr_in[13:0] output When set to 0, enable sd_adr_in[13:0] output
4	PAD_CNTRL[4]	When set to 1, enable output clock When set to 0, disable output clock
5	PAD_CNTRL[5]	When set to 1, close tri-state for leakage current test, output is high-Z When set to 0, open tri-state
6	PAD_CNTRL[6]	When set to 1, disable pad's internal pull up When set to 0, enable pad's internal pull up
7	Reserved	

MISCELLANEOUS 18 — REG 46, R/W

Bit	7	6	5	4	3	2	1	0
	I2CSA_LAT	GPIO_2_EN	GPIO_2_VAL	HALF_EN	GPIO_1_EN	GPIO_1_VAL	GPIO_0_EN	GPIO_0_VAL

Bit	Name	Function
0	GPIO_0_VAL	**GPIO bit 0 value** GPIO bit 0value.
1	GPIO_0_EN	**GPIO bit 0 output enable** When set to 1, GPIO bit 0 output enable. When set to 0, GPIO bit 0 is used as input port.
2	GPIO_1_VAL	**GPIO bit 1 value** GPIO bit 1value.
3	GPIO_1_EN	**GPIO bit 1 enable** When set to 1, GPIO bit 1 output enable. When set to 0, GPIO bit 1 is used as input port.
4	HALF_EN	**HALF TONE enable** When set to 1, half tone enable, pin HALF input will be used as half tone input; When set to 0, half tone disable
5	GPIO_2_VAL	**GPIO bit2 value** GPIO bit2 value.
6	GPIO_2_EN	**GPIO bit2 enable** When set to 1, GPIO bit2 output enable. When set to 0, GPIO bit2 is used as input port.
7	I2CSA_LAT	**I2C address selection latch** When set to 1, the SCLSA value will be latched to I2C address selection on the rising edge. When set to 0, the SCLSA value will be used as I2C address selection directly.

MISCELLANEOUS 19 **REG 47, R/W**

Bit	7	6	5	4	3	2	1	0
	HD_RST_CNTRL	OSD_RST_CNTRL	HD_OSD_TST_CNTRL	DI_PL_CUT	Reserved	PLL_CNTRL[16]	VSOUT_EN	HSOUT_EN

Bit	Name	Function
0	HSOUT_EN	**HSOUT enable** When set to 1, horizontal sync output enable. When set to 0, horizontal sync output disable and will always be zero.
1	VSOUT_EN	**VSOUT enable** When set to 1, vertical sync output enable. When set to 0, vertical sync output disable and will always be zero.
2	PLL_CNTRL[16]	**PLL VS[2] VCLK control bit[2]** Detail description is in reg41
3	Reserved	
4	DI_PL_CUT	**Diag_bob line buffer cut** When set to 1, the garbage part of Diag_bob line buffer output will be cut. When set to 0, Diag_bob line buffer output data will bypass.
5	HD_OSD_TST_CNTRL	**HD and OSD test bus selection** When set to 1, HD test bus is selected. When set to 0, OSD test bus is selected.
6	OSD_RST_CNTRL	**OSD reset control** When =1, OSD generation will be reset
7	HD_RST_CNTRL	**HD reset control** When =1, HD processor will be reset.

MISCELLANEOUS 20 — REG 4D, R/W

Bit	7	6	5	4	3	2	1	0
	MISC_ST_CNTRL[2:1]		HD_INPUT_ZERO	IF_INPUT_ZERO	VDS_RST_CNTRL	MADPT_RST_CNTRL	MISC_ST_CNTRL[0]	IF_RST_CNTRL

Bit	Name	Function
0	IF_RST_CNTRL	**Input Formatter reset control** When set to 1, input formatter will be reset.
1	MISC_ST_CNTRL[0]	**Miscellaneous status selection bit0** Miscellaneous status selection bits [0] to select REG_0E, REG_0F
2	MADPT_RST_CNTRL	**Deint_madpt3 reset control** When set to 1, deint_madpt3 will be reset.
3	VDS_RST_CNTRL	**Vds_proc reset control** When set to 1, vds_proc will be reset.
4	IF_INPUT_ZERO	**Input Formatter Input Data Force to Zero** When set to 1, input formatter's input data will be forced to zero. The SD channel's data transfer will stop to save power.
5	HD_INPUT_ZERO	**Hd_proc Input Data Force to Zero** When set to 1, hd_proc's input data will be forced to zero. The HD channel's data transfer will stop to save power.
7-6	MISC_ST_CNTRL[2:1]	**Miscellaneous status selection bit[2:1]** Miscellaneous status selection bits [2:1] to select REG_0E, REG_0F

MISCELLANEOUS 21 — REG 69, R/W

Bit	7	6	5	4	3	2	1	0
	GPIO_6_VAL	GPIO_5_VAL	GPIO_4_VAL	GPIO_3_VAL	GPIO_6_SEL	GPIO_5_SEL	GPIO_4_SEL	GPIO_3_SEL

Bit	Name	Function
0	GPIO_3_SEL	**GPIO BIT3 SELECTION** When set to 1, PIN93(CLKOUT) will be used as GPIO bit3
1	GPIO_4_SEL	**GPIO BIT4 SELECTION** When set to 1, PIN76(CS0) will be used as GPIO bit4
2	GPIO_5_SEL	**GPIO BIT5 SELECTION** When set to 1, PIN73(CS1) will be used as GPIO bit5
3	GPIO_6_SEL	**GPIO BIT6 SELECTION** When set to 1, PIN71(fbclk) will be used as GPIO bit6
4	GPIO_3_VAL	**GPIO BIT3 OUTPUT VALUE**
5	GPIO_4_VAL	**GPIO BIT4 OUTPUT VALUE**
6	GPIO_5_VAL	**GPIO BIT5 OUTPUT VALUE**
7	GPIO_6_VAL	**GPIO BIT5 OUTPUT VALUE**

Bit	Name	Function

MISCELLANEOUS 22 — REG 6A, R/W

Bit	7	6	5	4	3	2	1	0
	Reserved				GPIO_6_EN	GPIO_5_EN	GPIO_4_EN	GPIO_3_EN

Bit	Name	Function
0	GPIO_3_EN	**GPIO BIT3 output enable** When set to 1, GPIO bit3 output enable When set to 0, GPIO bit3 is used as input port.
1	GPIO_4_EN	**GPIO BIT4 output enable** When set to 1, GPIO bit4 output enable When set to 0, GPIO bit4 is used as input port
2	GPIO_5_EN	**GPIO BIT5 output enable** When set to 1, GPIO bit5 output enable When set to 0, GPIO bit5 is used as input port
3	GPIO_6_EN	**GPIO BIT6 output enable** When set to 1, GPIO bit6 output enable When set to 0, GPIO bit6 is used as input port
7-4	Reserved	

MISCELLANEOUS 23 — REG 6B, RO

Bit	7	6	5	4	3	2	1	0
	Reserved				GPIO_6_ST (RO)	GPIO_5_ST (RO)	GPIO_4_ST (RO)	GPIO_3_ST (RO)

Bit	Name	Function
0	GPIO_3_ST (RO)	**GPIO BIT3 status bit, read only**
1	GPIO_4_ST (RO)	**GPIO BIT4 status bit, read only**
2	GPIO_5_ST (RO)	**GPIO BIT5 status bit, read only**
3	GPIO_6_ST (RO)	**GPIO BIT6 status bit, read only**
7-4	Reserved	

MEMORY REGISTERS

MEMORY CONTROLLER 00 **REG 80, R/W**

Bit	7	6	5	4	3	2	1	0
	R_MSTINIC	R_MINITYP	RESERVED	R_RSTSQSGC	R_MIREFCK	R_MSLIDL[1:0]		R_MSOFTH

Bit	Name	Function
0	R_MSOFTH	**Software Control SDRAM Idle Period:** When this bit is 1, software programming will control the idle period to access memory. This bit is useful only when the register r_mslidl[1:0] sets 2'b11.
2-1	R_MSLIDL[1:0]	**SDRAM Idle Period Control and IDLE Done Select**: (default 0)

R_MSLIDL [1]	R_MSLIDL [0]	#of VS (vertical syn)
0	0	2
0	1	3
1	0	1
1	1	R_MSOFTH

3	R_MIREFCK	**Initial Cycle Refresh Period Clock Number Select:** When this bit is 1, there are 10 memory refresh cycles will be inserted during initial cycle, otherwise 8 memory refresh cycle will be inserted.
4	R_RSTSQSGC	**SDRAM Reset Signal:** When this bit is 1, will generate 5-mmclk pulse, and reset memory controller timing, data pipe and state machine;
5	RESERVED	**RESERVED**
6	R_MINITYP	**Initial Cycle Mode Select:** When this bit is 1, then during initial period, the mode cycle will go before refresh cycle; otherwise refresh cycle will be before mode cycle.
7	R_MSTINIC	**SDRAM Start Initial Cycle:** This register should work with the register **80/[2:0]**; When this bit is 1, memory controller initial cycle enable; When this bit is 0, memory controller initial cycle disable.

MEMORY CONTROLLER 01 — REG 81, R/W

Bit	7	6	5	4	3	2	1	0
	R_MSGM_[7:0]							

Bit	Name	Function
7-0	R_MSGM_[7:0]	**SDRAM Mode Information Low 8bits:** [2:0] Burst length, [3] Wrap type : 0 = Sequential , 1= interleave ; [6:4] Latency mode, 010: select Latency =2; 011: select Latency =3.

MEMORY CONTROLLER 02 — REG 82, R/W

Bit	7	6	5	4	3	2	1	0
	R_MSELDLYCELL	R_MODE_CS	R_MODE_BA	R_MMODECK	CLK_SEL_CTRL	R_MSGM_[10:8]		

Bit	Name	Function
2-0	R_MSGM_[10:8]	**SDRAM Mode Information High 3 bits** Reserved for future usage.
3	CLK_SEL_CTRL	**Select Clock Feed Back from PAD;** This register will be valid when the register BC/[4] = 1'b0; When this bit is 1, select external pad feed back clock; When this bit is 0, select internal PAD feed back clock. If BC/[4] = 1'b1,this bit should be set 0.
4	R_MMODECK	**Mode Cycle Period Select** When this bit is 1, then mode cycle for memory initialization will be 3 clocks; Otherwise it will be 2 clocks.
5	R_MODE_BA	**Bank Select Value In Load Mode Register Cycle :(default value is 0)** This register 's aim is compatible with more SDRAM chips.
6	R_MODE_CS	**Chip Select Value in Load Mode Register Cycle :(default value is 0)** This register 's aim is compatible with more SDRAM chips.
7	R_MSELDLYCELL	**Select SDRAM Delay Cell:** This register is control the delays of Command, address and data sent to PAD; When it is at 1, select NCASDLY cell, when it is at 0, select DLY8LV cell.

MEMORY CONTROLLER 03 — REG 83, R/W

Bit	7	6	5	4	3	2	1	0
	R_MCSRLAT_[2:0]			R_MSGCKRD_[2:0]]			R_MINVSGRCK	R_MINVSGRCK_2PAD

Bit	Name	Function
0	R_MINVSGRCK_2PAD	**Invert Memory Rising Edge Clock to PAD:** When this bit is 1, invert memory clock and send to PAD; When this bit is 0, will bypass memory clock and send to PAD.
1	R_MINVSGRCK	**Read memory data with Memory Clock rising or falling edge:** When this bit is 1, with Memory clock falling edge; When this bit is 0, with Memory clock rising edge.
4-2	R_MSGCKRD_[2:0]	**SDRAM Rising Edge Clock Delay for Latching Read Data: (default set 3'b 000); With DLY8LV and NCASDLY**
7-5	R_MCSRLAT_[2:0]	**SDRAM Latch Signal Generate Timing For Memory Data Read Cycle:**

R_MSGCKRD_[2]	R_MSGCKRD_[1]	R_MSGCKRD_[0]	#of Mclk
0	0	0	0.00/0.0
0	0	1	0.25/2.0
0	1	0	0.50/4.0
0	1	1	0.75/6.0
1	0	0	1.00
1	0	1	1.50
1	1	0	2.00
1	1	1	3.00

latency =2 Set 3'b000 ;latency =3 set 3'b011 ;

R_MCSRLAT_[2]	R_MCSRLAT_[1]	R_MCSRLAT_[0]	#of Mclk
0	0	0	3
0	0	1	2
0	1	0	1
0	1	1	4
1	0	0	5
1	0	1	6
1	1	0	7
1	1	1	8

MEMORY CONTROLLER 04 — REG 84, R/W

Bit	7 6	5	4 3 2	1 0
	R_MEMREFN_[1:0]	R_MREF5	R_MREFCK_[2:0]	R_MACTCK_[1:0]

Bit	Name	Function
1-0	R_MACTCK_[1:0]	**Number of Memory Clock For SDRAM Active Cycle:**

R_MACTCK_[1]	R_MACTCK_[0]	# of Mclk
0	0	3
0	1	2
1	0	4
1	1	5

Bit	Name	Function
4-2	R_MREFCK_[2:0]	**Number of Memory Clock For SDRAM Refresh Cycle:**

R_MREFCK_[2]	R_MREFCK_[1]	R_MREFCK_[0]	# Of Mclk
0	0	0	9
0	0	1	7
0	1	0	8
0	1	1	10
1	X	X	11

Bit	Name	Function
5	R_MREF5	**For VGA Mode of Refresh Cycle:**

R_MEMREFN_[1]	R_MEMREFN_[0]	R_MREF5	#of refresh
0	0	0	3
0	0	1	5
0	1	X	1
1	0	X	2
1	1	X	4

Bit	Name	Function
7-6	R_MEMREFN_[1:0]	**Memory Refresh Rate:(It is above)**

MEMORY CONTROLLER 05 — REG 85, R/W

Bit	7	6	5	4	3	2	1	0
	R_MPHGCK_[1:0]		R_MRASPD_[2:0]			R_MTWRSL_[2:0]		

Bit	Name	Function
2-0	R_MTWRSL_[2:0]	TWR Period Select (Number of Memory Clock Inserted from Last Write Cycle to Precharge)
5-3	R_MRASPD_[2:0]	TRAS Timing (from active cycle to precharge cycle)
7-6	R_MPHGCK_[1:0]	Number of Memory Clock For SDRAM Precharge Cycle:

R_MTWRSL_[2]	R_MTWRSL_[1]	R_MTWRSL_[0]	#OF MCLK
0	0	0	0
0	0	1	1
0	1	0	2
0	1	1	3
1	0	0	4
1	0	1	5

R_MRASPD_[2]	R_MRASPD_[1]	R_MRASPD_[0]	# OF MCLK
0	0	0	6
0	0	1	4
0	1	0	5
0	1	1	7
1	X	X	3

R_MPHGCK_[1]	R_MPHGCK_[0]	# of Mclk
0	0	3
0	1	2
1	0	4
1	1	5

MEMORY CONTROLLER 06 — REG 86, R/W

Bit	7	6	5	4	3	2	1	0
	R_MR2WNOP_[1:0]		R_MSGW2RSL_[1:0]		R_MCSSEL_[1:0]		R_MBKSEL_[1:0]	

Bit	Name	Function
1-0	R_MBKSEL_[1:0]	**Bank Select Address Mux:**

R_MBKSEL_[1]	R_MBKSEL_[0]	# OF ADDRESS BIT
0	0	ADR 19
0	1	ADR 20
1	0	ADR 21
1	1	ADR 22

Bit	Name	Function
3-2	R_MCSSEL_[1:0]	**Chip Select Address Mux:**

R_MCSSEL_[1]	R_MCSSEL_[0]	#OF ADDRESS BIT
0	0	ADR 20
0	1	ADR 21
1	0	ADR 22

Bit	Name	Function
5-4	R_MSGW2RSL_[1:0]	**Memory Write to Read Dummy Clock Cycle Insertion:**

R_MSGW2RSL_[1]	R_MSGW2RSL_[0]	#OF MCLK
0	0	2
0	1	0
1	0	3
1	1	4

Bit	Name	Function
7-6	R_MR2WNOP_[1:0]	**Number of Dummy Clock For SDRAM Read to Write Cycle:**

R_MR2WNOP_[1]	R_MR2WNOP_[0]	#OF MCLK
0	0	2
0	1	0
1	0	3
1	1	4

MEMORY CONTROLLER 07 — REG 87, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED				R_ROWST9	R_ROWST8	R_COLST1	R_COLST0

Bit	Name	Function
0	R_COLST0	**Col Address Start with address bit 0 (default value 1).** When this bit is 1,column address starts with address bit 0 When this bit is 0,column address will not start with address bit 0
1	R_COLST1	**Col Address Start with address bit 1 (default value 0).** When this bit is 1,column address starts with address bit 1 When this bit is 0,column address will not start with address bit 1
2	R_ROWST8	**Row Address Start with address bit 8 (default value 1).** When this bit is 1,row address starts with address bit 8 When this bit is 0,row address will not start with address bit 8.
3	R_ROWST9	**Row Address Start with address bit 9 (default value 0).** When this bit is 1,row address starts with address bit 9; When this bit is 0,row address will not start with address bit 9.
7:4	RESERVED	

MEMORY CONTROLLER 08 — REG 88, R/W

Bit	7	6	5	4	3	2	1	0
	R_MWOESLPZ	R_MWOEZDLY_[2:0]			RESERVED			

Bit	Name	Function
3:0	RESERVED	
6-4	R_MWOEZDLY_[2:0]	**Data TRI_STATE Enable Delay Control Bits: with DLY8LV and NCASDLY**
7	R_MWOESLPZ	**SDRAM Data TRI_STATE Enable Extend Pipe Select:** When this register is 1: the SDRAM data tri_state enable will extend a pipe; When this register is 0: the SDRAM data tri_state enable will be selected by other registers.

R_MWOEZDLY_[2]	R_MWOEZDLY_[1]	R_MWOEZDLY_[0]	#OF NS
0	0	0	0.00/0.0
0	0	1	0.25/2.0
0	1	0	0.50/4.0
0	1	1	0.75/6.0
1	0	0	1.00
1	0	1	1.50
1	1	0	2.00
1	1	1	3.00

MEMORY CONTROLLER 09 — REG 89, R/W

Bit	7	6	5	4	3	2	1	0
	R_SLPRASZ	R_SLPCASZ	R_SLPWEZ	R_MDATAR_PIPE_[1:0]			R_MDATDLY_[2:0]	

Bit	Name	Function
2-0	R_MDATDLY_[2:0]	**Data Delay Control Bits: with DLY8LV and NCASDLY**

R_MDATADLY_[2]	R_MDATADLY_[1]	R_MDATADLY_[0]	#OF NS
0	0	0	0.00/0.0
0	0	1	0.25/2.0
0	1	0	0.50/4.0
0	1	1	0.75/6.0
1	0	0	1.00
1	0	1	1.50
1	1	0	2.00
1	1	1	3.00

Bit	Name	Function
4-3	R_MDATAR_PIPE_[1:0]	**Memory Write Data (Rising Edge) Pipe Select: default value 2'b00; (In 5705,only 2'b00)**

R_MDATAR_PIPE_[1]	R_MDATAR_PIPE_[0]	# OF PIPE
0	0	0
0	1	1

Bit	Name	Function
5	R_SLPWEZ	**SDRAM WE Command Pipe Select:** When it is at 0,WE signal pass through a pipe, or it will bypass a pipe;
6	R_SLPCASZ	**SDRAM CAS Command Pipe Select:** When it is at 0,CAS signal pass through a pipe, or it will bypass a pipe;
7	R_SLPRASZ	**SDRAM RAS Command Pipe Select:** When it is at 0, RAS signal pass through a pipe, or it will bypass a pipe;

MEMORY CONTROLLER 10 **REG 8A, R/W**

Bit	7	6	5	4	3	2	1	0
	R_MRASDLY_[2:0]			R_MCASDLY_[2:0]			R_MWEDLY_[2:1]	

Bit	Name	Function
1-0	R_MWEDLY_[2:1]	**WE Delay Control bits High 2 bits: default value 3'b000; with DLY8LV and NCASDLY**

R_MWEDLY_[2]	R_MWEDLY_[1]	R_MWEDLY_[0]	#OF NS
0	0	0	0.00/0.0
0	0	1	0.25/2.0
0	1	0	0.50/4.0
0	1	1	0.75/6.0
1	0	0	1.00
1	0	1	1.50
1	1	0	2.00
1	1	1	3.00

Bit	Name	Function
4-2	R_MCASDLY_[2:0]	**CAS Delay Control bits: default value 3'b000; with DLY8LV and NCASDLY**

R_MCASDLY_[2]	R_MCASDLY_[1]	R_MCASDLY_[0]	#OF NS
0	0	0	0.00/0.0
0	0	1	0.25/2.0
0	1	0	0.50/4.0
0	1	1	0.75/6.0
1	0	0	1.00
1	0	1	1.50
1	1	0	2.00
1	1	1	3.00

Bit	Name	Function
7-5	R_MRASDLY_[2:0]	**RAS Delay Control bits: default value 3'b000;with DLY8LV and NCASDLY**

R_MRASDLY_[2]	R_MRASDLY_[1]	R_MRASDLY_[0]	#OF NS
0	0	0	0.00/0.0
0	0	1	0.25/2.0
0	1	0	0.50/4.0
0	1	1	0.75/6.0
1	0	0	1.00
1	0	1	1.50
1	1	0	2.00
1	1	1	3.00

MEMORY CONTROLLER 11 — REG 8B, R/W

Bit	7	6	5	4	3	2	1	0
	R_MWEDLY_[0]	R_MDQMDLY_[2:0]			R_MADRDLY_[2:0]			R_MCSDLY_[2]

Bit	Name	Function
0	R_MCSDLY_[2]	**Chip Select Delay Control bit High 1bit:** Description is following as: with DLY8LV and NCASDLY
3-1	R_MADRDLY_[2:0]	**Address Delay Control bits: with DLY8LV and NCASDLY**
6-4	R_MDQMDLY_[2:0]	**DQM Delay Control bits: with DLY8LV and NCASDLY**
7	R_MWEDLY_[0]	**WE Delay Control bits Low 1bits: following above**

R_MADRDLY_[2]	R_MADRDLY_[1]	R_MADRDLY_[0]	#OF NS
0	0	0	0.00/0.0
0	0	1	0.25/2.0
0	1	0	0.50/4.0
0	1	1	0.75/6.0
1	0	0	1.00
1	0	1	1.50
1	1	0	2.00
1	1	1	3.00

R_MDQMDLY_[2]	R_MDQMDLY_[1]	R_MDQMDLY_[0]	#OF NS
0	0	0	0.00/0.0
0	0	1	0.25/2.0
0	1	0	0.50/4.0
0	1	1	0.75/6.0
1	0	0	1.00
1	0	1	1.50
1	1	0	2.00
1	1	1	3.00

MEMORY CONTROLLER 12 — REG 8C, R/W

Bit	7	6	5	4	3	2	1	0
	R_MCSDLY_[1:0]		R_MBADLY_[2:0]			R_MCLKRDLY_[2:0]		

Bit	Name	Function
2-0	R_MCLKRDLY_[2:0]	Clk of Rising Edge Delay Control bits: with DLY8LV and NCASDLY

R_MCLKDLY_[2]	R_MCLKDLY_[1]	R_MCLKDLY_[0]	#OF NS
0	0	0	0.00/0.0
0	0	1	0.25/2.0
0	1	0	0.50/4.0
0	1	1	0.75/6.0
1	0	0	1.00
1	0	1	1.50
1	1	0	2.00
1	1	1	3.00

Bit	Name	Function
5-3	R_MBADLY_[2:0]	Bank Delay Control bits: with DLY8LV and NCASDLY

R_MBADLY_[2]	R_MBADLY_[1]	R_MBADLY_[0]	#OF NS
0	0	0	0.00/0.0
0	0	1	0.25/2.0
0	1	0	0.50/4.0
0	1	1	0.75/6.0
1	0	0	1.00
1	0	1	1.50
1	1	0	2.00
1	1	1	3.00

Bit	Name	Function
7-6	R_MCSDLY_[1:0]	Chip Select Delay Control Low 2bits: with DLY8LV and NCASDLY

R_MCSDLY_[2]	R_MCSDLY_[1]	R_MCSDLY_[0]	#OF NS
0	0	0	0.00/0.0
0	0	1	0.25/2.0
0	1	0	0.50/4.0
0	1	1	0.75/6.0
1	0	0	1.00
1	0	1	1.50
1	1	0	2.00
1	1	1	3.00

MEMORY CONTROLLER 13 — REG 8D, R/W

Bit	7	6	5	4	3	2	1	0
	R_MQACTOFF	R_MSLFBK	R_SD_SPCL10	R_SD_SPCL9	R_SD_SPCL8	R_M10CQADR	R_M9CQADR	R_M8CQADR

Bit	Name	Function
0	R_M8CQADR	**Memory Column Address Enable For Address bit 8 (default value 0)** For others SDRM chip that column address more than 8bits When this bit is 1,address 8 will act as column address; When this bit is 0,address 8 will not be column address.
1	R_M9CQADR	**Memory Column Address Enable For Address bit 9 (default value 0)** For others SDRM chip that column address more than 8bits When this bit is 1,address 9 will act as column address; When this bit is 0,address 9 will not be column address.
2	R_M10CQADR	**Memory Column Address Enable For Address bit 10 (default value 0)** For others SDRAM chip that column address more than 8bits When this bit is 1,address 10 will act as column address; **When this bit is 0,address 10 will not be column address.**
3	R_SD_SPCL8	**Special Pin8 For Precharge: default value 0** If Memory module Address 8 is special Pin, In initialization cycle, must set this register to 1, and precharge all banks; otherwise, will set 0.
4	R_SD_SPCL9	**Special Pin9 For Precharge: default value 0** If Memory module Address 9 is special Pin, In initialization cycle, must set this register to 1, and precharge all banks otherwise, will set 0.
5	R_SD_SPCL10	**Special Pin10 For Precharge: default value 1** If Memory module Address 10 is special Pin, **In initialization cycle, must set this register to 1, and precharge all banks otherwise, will set 0.**
6	R_MSLFBK	**Select Data Latch Signal through FBK clock:** When this bit is 1, it will capture data with feedback clock path, When this bit is 0, It will capture data with memory clock.
7	R_MQACTOFF	**Turn Off Qualified Active Cycle Done: default value 0;** This register is 1, will shut off active state after refresh cycle. This register is 0, will turn on active state after refresh cycle;

MEMORY CONTROLLER 14 — REG 8E, R/W

Bit	7	6	5	4	3	2	1	0
	R_MCSFST	R_MBKFST	R_PCHGFST	R_MWRFST	R_MDISKQK	RESERVED	R_MSGTURBO	R_MDQMHI

Bit	Name	Function
0	R_MDQMHI	**Control Write Enable (DQM) High Bits** *Don't care*, default value 0, In 5705 chip, this register will not be on effect, it use for future.
1	R_MSGTURBO	**Control Active State before Refresh Cycle** *Don't care*, default value 1, This register will on effect when set "8D/7" is 1. When this bit sets 1 ,will generate a signal to pull down the act cycle done signal after refresh cycle ,make it be one-mmclk pulse ;or will make this signal be level .
2	RESERVED	**RESERVED**
3	R_MDISKQK	**Add No Operation For Precharge Cycle** *Don't care*, default value 0; When this bit sets 0 ,will add NOP for precharge cycle ; Whenj this bit sets 1,will no add NOP for precharge cycle.
4	R_MWRFST	**SDRAM Write and Read Signal Fast Mode Signal** *Don't care*, default value 0;In fast mode, When this bit is 1,DQM signal will advance. When this bit is 0,DQM signal will be normal,
5	R_PCHGFST	**SDRAM Precharge Fast Mode Signal** *Don't care*, default value 1 ;In fast mode, When is this bit 1,precharge will advance. When is this bit 0,precharge will be normal
6	R_MBKFST	**SDRAM Bank Select Fast Mode Signal** *Don't care*, default value 1 ;In fast mode, When this bit is 1,bank select will advance. **When this bit is 0,bank select will be normal**
7	R_MCSFST	**SDRAM Chip Select Fast Mode Signal** *Don't care*; default value 0 In fast mode, When this bit is 1, chip select will advance. When this bit is 0,chip select will be normal

MEMORY CONTROLLER 15 — REG 8F, R/W

Bit	7	6	5	4	3	2	1	0
	R_MINVSGRCK_FBK	R_MWOEZ_DLY	R_MX_TST_[1:0]		R_TST_BIT	R_MLCDWCAPSW	R_CAPSWP	R_VRTHTCRT

Bit	Name	Function
0	R_VRTHTCRT	**Play back high request priority exchange with read FIFO high request** When this bit is 1, read FIFO high request > play back high request; When this bit is 0, play back high request >read FIFO high request;
1	R_CAPSWP	**Capture request exchange with PlayBack low request and Read FIFO low request** When this bit is 0: play back low req > read FIFO low req > capture req When this bit is 1: cap req > play back low req > read FIFO low req
2	R_MLCDWCAPSW	**Write FIFO request priority exchange with capture request** When this bit is 1, capture request >write FIFO request, When this bit is 0, write FIFO request > capture request
5-3	R_MX_TST_[2:0]	**Test Logic Controll** Select four groups test signals (internal hardware debug use only)
6	R_MWOEZ_DLY	**SDRAM DATA TRI_state ENABLE DELAY SELECT:** When this register is 0: will select extension from delay cells; When this register is 1: will select not extension. This register will control sdram data tri_state enable with the register r_mwoeslpz **.**
7	R_MINVSGRCK_FBK	**Control feedback clock register** When this bit is at 1, will invert feedback clock; When it's at 0, will bypass feedback clock;

MEMORY CONTROLLER 16 — REG BB, R/W

Bit	7	6	5	4	3	2	1	0
	R_MSGM_[11]	R_MLAT_PIPE	R_MREFLOW_SEL	R_MTRASCTL	RESERVED			R_M11CQADR

Bit	Name	Function
0	R_M11CQADR	**Memory Column Address Enable For Address bit 11 (default value 0)** For others SDRM chip that column address more than 8bits When this bit is 1,address 11 will act as column address; When this bit is 0,address 11 will not be column address.
3:1	RESERVED	**RESERVED**
4	R_MTRASCTL	**CONTROL TIMING FOR ACTIVE TO PRECHARGE;** When this bit sets 1, will added active to precharge timing; When this bit sets 0, will no change.
5	R_MREFLOW_SEL	**REFRESH CYCLE SIGNAL IS LOW:** When this bit is 1, when refresh more than 2 times, in refresh cycle, make DQM will high; When this bit is 0, only for refresh one time, DQM will high.
6	R_MLAT_PIPE	**LATCH READ DATA ADDED A PIPE** When this register sets 1, latch signal will add a pipe; When this register sets 0, no change.
7	R_MSGM_[11]	**SDRAM Mode Information High 4 bits** Reserved for future usage.

MEMORY CONTROLLER 17 **REG BC, R/W**

Bit	7	6	5	4	3	2	1	0
	R_MSELDLYCELL_CLK	R_MSELDLYCELL_FBK	R_MCLK_SEL	FBKCLK_CS_SEL	R_FDBK_DAT_INV	R_MBASEL8M	R_MCS0SEL8M	R_MCS1SEL8M

Bit	Name	Function
0	R_MCS1SEL8M	**CHIP SELECT 1 PAD SAHRE WITH BANK SELECT 1 :** When this register is 1: chip select 1 pad will be bank select 1 pad, support 1M x 16bits x 4 banks memory chip; When this register is 0: chip select 1 pad will be chip select 1 pad, support 1M x16bits x 2banks memory chip;
1	R_MCS0SEL8M	**CHIP SELECT 0 PAD SAHRE WITH BANK SELECT 0 :** When this register is 1: chip select 0pad will be bank select 0 pad, support 1M x 16bits x 4 banks memory chip; When this register is 0: chip select 0 pad will be chip select 0 pad, support 1M x16bits x 2banks memory chip;
2	R_MBASEL8M	**BANK SELECT PAD SHARE WITH ADDRESS 11 :** When this register is 1: bank select pad will be memory address 11 bit, support 1M x 16bits x4 banks memory chip; When this register is 0: bank select pad will be bank select pad, support 1M x16bits x 2banks memory chip;
3	R_FDBK_DAT_INV	**FEEDBACK CLOCK DATA PATH SELECT**: When this register set 1, it will select falling edge fetch feedback data; When this register set 0, it will select rising edge fetch feedback data.
4	FBKCLK_CS_SEL	**FEEDBACK CLOCK PAD SHARE WITH CHIP SELECT 2:** When this register is 1: Pad will be chip select 2 PAD; When this register is 0: Pad will be feedback clock pad; This register uses only 6M memory, 3 chips.
5	R_MCLK_SEL	**FEEDBACK CLOCK SELECT SOURCE:** When this bit sets 1, feedback clock will select PLL clock; When this bit sets 0, feedback clock will select clock from PAD.
6	R_MSELDLYCELL_FBK	**Select SDRAM Delay Cell:** This register is only control the delay of feed back clock. When it is at 1, select NCASDLY cell, when it is at 0, select DLY8LV cell.
7	R_MSELDLYCELL_CLK	**Select SDRAM Delay Cell:** This register is only control the delay of clock send to PAD When it is at 1, select NCASDLY cell, when it is at 0, select DLY8LV cell.

CAPTURE REGISTERS

CAPTURE 00 **REG 90, R/W**

Bit	7	6	5	4	3	2	1	0
	RESERVED			CAP_CNTRL_[4]	CAP_CNTRL_[3]	CAP_CNTRL_[2]	CAP_CNTRL_[1:0]	

Bit	Name	Function
1-0	CAP_CNTRL_[1:0]	**Capture Test logic control:** Bit [1:0]: select capture internal test bus.
2	CAP_CNTRL_[2]	**Horizontal request end** When this bit set 1, the final capture request of one line is in the horizontal blank rising edge, set 0 capture request will free run
3	CAP_CNTRL_[3]	**CAPTURE FREEZ WRITE ENABLE** When this bit set 1, the de-interlace write enable will be freeze, set 0 will be normal
4	CAP_CNTRL_[4]	**MEM_FF_TOP FIFO STATUS OUTPUT ENABLE** When this bit set 1, all FIFO status will output, set 0, no FIFO status output
7:5	CAP_CNTRL_[7:5]	 **RESERVED**

CAPTURE 01 — REG 91, R/W

Bit	7	6	5	4	3	2	1	0
	CAP_CNTRL_[15]	CAP_CNTRL_[14:13]			CAP_CNTRL_[12:8]			

Bit	Name	Function
0	CAP_CNTRL_[8]	**Enable capture** When it's set 1, capture will be turn on. When it's set 0, capture will be turn off.
1	CAP_CNTRL_[9]	**Request generated when capture FIFO half** When set to 1, request generated when capture FIFO half. When set to 0, request generated when capture FIFO write pointer is 1.
2	CAP_CNTRL_[10]	**Capture double buffer status invert before output** When set to 1, double buffer status invert. When set to 0, double buffer status doesn't change.
3	CAP_CNTRL_[11]	**Enable double buffer** When set to 1, enable double buffer. When set to 0, disable double buffer.
4	CAP_CNTRL_[12]	**Capture request freeze**
5	CAP_CNTRL_[13]	**Enable safe guard function** When set to 1, turn on safe guard function. When set to 0, turn off safe guard function.
6	CAP_CNTRL_[14]	**Enable input v-sync reset FIFO** When set to 1, enable feed back v-sync reset FIFO. When set to 0, disable feed back v-sync reset FIFO.
7	CAP_CNTRL_[15]	**Enable address add by 2** When set to 1,address added by 2 per pixel, When set to 0,added by 1 per pixel.

CAPTURE 02 — REG 92, R/W

Bit	7	6	5	4	3	2	1	0
	CAP_CNTRL_[23:16]							

Bit	Name	Function
7-0	CAP_CNTRL_[23:16]	**Safe Guard Address For Buffer A:** Safe guard address A [7:0], Mapping to 32bits width data bus field.

CAPTURE 03 — REG 93, R/W

Bit	7	6	5	4	3	2	1	0
	CAP_CNTRL_[31:24]							

Bit	Name	Function
7-0	CAP_CNTRL_[31:24]	**Safe Guard Address For Buffer A:** Safe guard address A [15:8]; Mapping to 32bits width data bus field.

CAPTURE 04 — REG 94, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED			CAP_CNTRL_[36:32]				

Bit	Name	Function
4-0	CAP_CNTRL_[36:32]	**Safe Guard Address For Buffer A[20:16]:** Safe guard address A [20:16], Mapping to 32bits width data bus field.
7-5	RESERVED	

CAPTURE 05 — REG 7D, R/W

Bit	7	6	5	4	3	2	1	0
	CAP_NEW_CNTRL_[36:32]							

Bit	Name	Function
6-0	CAP_NEW_CNTRL_[6:0]	**Capture FIFO status register**
7	CAP_NEW_CNTRL_[7]	**Capture FIFO status select** .

PLAY BACK REGISTERS

PLAY BACK 00 — REG 95, RO

Bit	7	6	5	4	3	2	1	0
	PB_CNTRL_[7:0]							

Bit	Name	Function
7-0	PB_CNTRL_[7:0]	**CRC test result output (read only)** For CRC test read out by register read.

PLAY BACK 01 — REG 96, R/W

Bit	7	6	5	4	3	2	1	0
	PB_CNTRL_[15:8]							

Bit	Name	Function
0	PB_CNTRL_[8]	**Disable refresh request generation** When set to 1, disable refresh request generation. When set to 0, enable refresh request generation.
2-1	PB_CNTRL_[10:9]	**Enable playback request mode** 00: PBHREQ 0, PBLREQ Low request 01: PBHREQ 0, PBLREQ High request 10: PBHREQ Low request, PBLREQ 0 11: PBHREQ High request, PBLREQ Low request
3	PB_CNTRL_[11]	**Enable VDS input to select playback output or de-interlace data out** When this bit is 1, select de-interlace data out to VDS. When this bit is 0, select playback output to VDS.
4	PB_CNTRL_[12]	**Enable double field display** When set to 1, enable double field display. When set to 0, disable double field display.
5	PB_CNTRL_[13]	**Enable double buffer** When set to 1, enable double buffer. When set to 0, disable double buffer.
6	PB_CNTRL_[14]	**Exchange playback two frames output data** When set to 1, exchange playback current frame with past frame and output. **When set to 0, don't exchange.**
7	PB_CNTRL_[15]	**Enable Playback** When it's set 1, play back will be on work, or will not work.

PLAY BACK 02 — REG 97, R/W

Bit	7	6	5	4	3	2	1	0
	PB_RBUF_SEL	PB_RBUF_INV	PB_CNTRL_[21:16]					

Bit	Name	Function
5-0	PB_CNTRL_[21:16]	**Master line flag [5:0]** Playback FIFO policy master value: This field will define FIFO high request timing.
6	PB_CNTRL_[22]	**PB_RBUF_INV** When rate convert from up to down, capture FIFO will refer to the play back buffer status, this bit is invert play back buffer status.
7	PB_CNTRL_[23]	**PB_RBUF_SEL** When rate convert from up to down, capture FIFO will refer to the play back buffer status, this bit will be set to 1. Otherwise, it will be set to 0.

PLAY BACK 03 — REG 98, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED		PB_CNTRL_[29:24]					

Bit	Name	Function
5-0	PB_CNTRL_[29:24]	**General line flag [5:0]** Playback FIFO policy general value: This field will define FIFO low request timing.
7-6	RESERVED	

CAPTURE AND PLAY BACK SHARED REGISTERS

CAPTURE AND PLAYBACK SHARED 00 — REG 99, R/W

Bit	7	6	5	4	3	2	1	0
	CAP_PB_CNTRL_[7:0]							

Bit	Name	Function
7-0	CAP_PB_CNTRL_[7:0]	**Capture and Play Back Buffer A START ADDRESS [7:0]:** Start Address buffer A [7:0], Mapping to 32bits width data bus field.

CAPTURE AND PLAYBACK SHARED 01 — REG 9A, R/W

Bit	7	6	5	4	3	2	1	0
	CAP_PB_CNTRL_[15:8]							

Bit	Name	Function
7-0	CAP_PB_CNTRL_[15:8]	**Capture and Play Back Buffer A START ADDRESS [15:8]:** Start Address buffer A [15:8], Mapping to 32bits width data bus field.

CAPTURE AND PLAYBACK SHARED 02 — REG 9B, R/W

Bit	7	6	5	4	3	2	1	0
	Rev.			CAP_PB_CNTRL_[20:16]				

Bit	Name	Function
4-0	CAP_PB_CNTRL_[20:16]	**Capture and Play Back Buffer A START ADDRESS[20:16]:** Start address buffer A [20:16], Mapping to 32bits width data bus field.
7-5	Rev.	**RESERVED**

CAPTURE AND PLAYBACK SHARED 03 — REG 9C, R/W

Bit	7	6	5	4	3	2	1	0
	CAP_PB_CNTRL_[31:24]							

Bit	Name	Function
7-0	CAP_PB_CNTRL_[31:24]	**Buffer B START address [7:0] / Rate convert display frame size [7:0]** Start address buffer B [7:0] / Rate convert display frame size [7:0], When in double buffer mode, this is defined as capture and playback buffer B start address. Mapping to 32bits width data bus field. When in rate convert mode, this is defined as playback frame buffer size (not including of budge line size). Mapping to 32bits width data bus field.

CAPTURE AND PLAYBACK SHARED 04 — REG 9D, R/W

Bit	7	6	5	4	3	2	1	0
	CAP_PB_CNTRL_[39:32]							

Bit	Name	Function
7-0	CAP_PB_CNTRL_[39:32]	**Buffer B START address [15:8] / Rate convert display frame size [15:8]** Start address buffer B [15:8] / Rate convert display frame size [15:8], When in double buffer mode, this is defined as capture and playback buffer B start address. Mapping to 32bits width data bus field. When in rate convert mode, this is defined as playback frame buffer size (not including of budge line size). Mapping to 32bits width data bus field.

CAPTURE AND PLAYBACK SHARED 05 — REG 9E, R/W

Bit	7	6	5	4	3	2	1	0
	Rev.			CAP_PB_CNTRL_[44:40]				

Bit	Name	Function
4-0	CAP_PB_CNTRL_[44:40]	**Capture and Play Back Buffer B START address [20:16]** Start address buffer B [20:16]. Mapping to 32 bits width data bus field.
7-5	Rev.	**RESERVED**

CAPTURE AND PLAYBACK SHARED 06 — REG 9F, R/W

Bit	7	6	5	4	3	2	1	0
	CAP_PB_CNTRL_[55:48]							

Bit	Name	Function
7-0	CAP_PB_CNTRL_[55:48]	**Capture and Play Back Offset [7:0]:** Offset [7:0] will determine next line start address, Mapping to 64bits width data bus field.

CAPTURE AND PLAYBACK SHARED 07 — REG A0, R/W

Bit	7	6	5	4	3	2	1	0
	Rev.		CAP_PB_CNTRL_[61:60]		Rev.		CAP_PB_CNTRL_[57:56]	

Bit	Name	Function
1-0	CAP_PB_CNTRL_[57:56]	**Capture and Play Back Offset [9:8]:** Offset [9:8] will determine next line start address, mapping to 64 bits width data bus field.
3-2	Rev.	**RESERVED**
5-4	CAP_PB_CNTRL_[61:60]	**Fetch number [9:8] (PLAYBACK USE ONLY)** This will determine to fetch the number of pixels from memory each horizontal line. Mapping to 64bits width data bus field.
7-6	Rev.	**RESERVED**

CAPTURE AND PLAYBACK SHARED 08 — REG A1, R/W

Bit	7	6	5	4	3	2	1	0
	CAP_PB_CNTRL_[71:64]							

Bit	Name	Function
7-0	CAP_PB_CNTRL_[71:64]	**Fetch number:** Fetch number [7:0] will determine to fetch the number of pixels from memory, Mapping to 64bits width data bus field.

CAPTURE AND PLAYBACK SHARED 09 — REG A2, R/W

Bit	7	6	5	4	3	2	1	0
	CAP_PB_CNTRL_[79:72]							

Bit	Name	Function
7-0	CAP_PB_CNTRL_[79:72]	**Safe Guard Address Buffer B / Capture and playback Buffer C Start Address [7:0]** Safe guard address B [7:0] / Start address buffer C [7: 0], When in noise reduction mode, this is defined as capture and playback buffer C start address. Mapping to 32 bits width data bus field. When in other mode, this is defined as capture safe guard address buffer B (corresponding double buffer address B). Mapping to 32bits width data bus field.

CAPTURE AND PLAYBACK SHARED 10 — REG A3, R/W

Bit	7	6	5	4	3	2	1	0
	CAP_PB_CNTRL_[87:80]							

Bit	Name	Function
7-0	CAP_PB_CNTRL_[87:80]	**Safe Guard Address Buffer B / Capture and Play Back Buffer C Start Address [15:8]** Safe guard address buffer B / Start address buffer C [15:8], When in noise reduction mode, this is defined as capture and playback buffer C start address. Mapping to 32bits width data bus field. When in other mode, this is defined as capture safe guard address buffer B (corresponding double buffer address buffer B). Mapping to 32bits width data bus field.

CAPTURE AND PLAYBACK SHARED 11 — REG A4, R/W

Bit	7	6	5	4	3	2	1	0
	Reserved			CAP_PB_CNTRL_[92:88]				

Bit	Name	Function
4-0	CAP_PB_CNTRL_[92:88]	**Safe Guard Address Buffer B / Capture and Play Back Buffer C Start Address [20:16]** Safe guard address buffer B / start address buffer C [20:16], When in noise reduction mode, this is defined as capture and playback buffer C start address. Mapping to 32 bits width data bus field. When in other mode, this is defined as capture safe guard address buffer B (corresponding double buffer address buffer B). Mapping to 32bits width data bus field.
7-5	Reserved	

CAPTURE AND PLAYBACK SHARED 12 — REG A5, R/W

Bit	7	6	5	4	3	2	1	0
	CAP_PB_CNTRL_[103:96]							

Bit	Name	Function
7-0	CAP_PB_CNTRL_[103:96]	**Capture and Play Back Buffer D Start Address [7:0]** Start address buffer D [7:0] When in noise reduction mode, this is defined as capture and playback buffer D start address. Mapping to 32 bits width data bus field.

CAPTURE AND PLAYBACK SHARED 13 — REG BD, R/W

Bit	7	6	5	4	3	2	1	0
	CAP_PB_CNTRL_[111:104]							

Bit	Name	Function
7-0	CAP_PB_CNTRL_[111:104]	**Capture and Play Back Buffer D Start Address [15:8]** Start address buffer D [15:8] When in noise reduction mode, this is defined as capture and playback buffer D start address. Mapping to 32 bits width data bus field.

CAPTURE AND PLAYBACK SHARED 14 — REG BE, R/W

Bit	7	6	5	4	3	2	1	0
	M75HZ	Reserved		CAP_PB_CNTRL_[116:112]				

Bit	Name	Function
4-0	CAP_PB_CNTRL_[116:112]	**Capture and Play Back Buffer D Start Address [20:16]** Start address buffer D [20:16], When in noise reduction mode, this is defined as capture and playback buffer D start address. Mapping to 32 bits width data bus field.
6-5	Reserved	
7	M75HZ	**PlayBack Read buffer Invert:** When in interlace mode, if this bit is 1, the read buffer will be inverted.

CAPTURE AND PLAYBACK SHARED 15 — REG BF, R/W

Bit	7	6	5	4	3	2	1	0
	CAP_PB_CNTRL_[127:120]							

Bit	Name	Function
3-0	CAP_PB_CNTRL [123:120]	**Select noise reduce all kind of format** 0: disable noise reduce function 1: turn on PAL mode 2 (50hz to 50hz) and storage in memory 5 frames 2: turn on PAL mode 3 5: turn on NTSC mode 2 and storage memory 3 frames 6: turn on NTSC mode 3 9: turn on PAL mode 2 (50hz to 50hz, 50hz to 60hz, 50hz to 100hz) and storage memory 6 frames. D: turn on NTSC mode 2 (60hz to 60hz, 60hz to 120hz) and storage memory 4 frames Note: in 50 to 100hz and 60 to 120,we must turn on 96[4] = 1 In playback
6-4	CAP_PB_CNTRL [126:124]	**Playback test logic** To select playback test bus, total 8 groups can be selected.
7	CAP_PB_CNTRL [127]	**To select playback CRC output** 0 select playback CRC result high 4 bits, CRC long counter enable. 1 select playback CRC result low 8 bits.

WRITE FIFO REGISTERS

WRITE FIFO 00 — REG A6, R/W

Bit	7	6	5	4	3	2	1	0
	WFF_CNTRL_[7:5]			WFF_CNTRL_[4:2]			WFF_CNTRL_[1:0]	

Bit	Name	Function
1-0	WFF_CNTRL_[1:0]	**WRITE FIFO Test logic control:** **Bit[1:0] : select capture internal test bus.**
4-2	WFF_CNTRL_[4:2]	**HORIZONTAL BLANK PROGRAMMING DELAY VALUE**
7:5	WFF_CNTRL_[7:5]	**VERTICAL BLANK PROGRAMMING DELAY VALUE**

WRITE FIFO 01 — REG A7, R/W

Bit	7	6	5	4	3	2	1	0
	WFF_CNTRL_[15:8]							

Bit	Name	Function
0	WFF_CNTRL_[8]	**Enable write FIFO**
1	WFF_CNTRL_[9]	**Request generated when FIFO half** When set to 1, request generated when FIFO half. When set to 0, request generate when FIFO write pointer is 1.
2	WFF_CNTRL_[10]	**Write FIFO status invert** When set to 1, write FIFO status invert. When set to 0, write FIFO status don't change.
3	WFF_CNTRL_[11]	**Enable write FIFO safe guard** When set to 1, enable write FIFO safe guard. When set to 0, disable write FIFO safe guard.
4	WFF_CNTRL_[12]	**Enable input V-sync reset FIFO** When set to 1, enable feedback v-sync reset FIFO. When set to 0, disable feedback v-sync reset FIFO.
5	WFF_CNTRL_[13]	**WRITE FIFO Address count select:** When it's set to 1, address added by 2 per pixel. When it's set to 0, address added by 1 per pixel.
6	WFF_CNTRL_[14]	**WRITE FIFO WRITE ONE FIELD YUV ENABLE** When set 1, write FIFO will write one field YUV, set 0, will write one frame Y.
7	WFF_CNTRL_[15]	**WRITE FIFO STATUS SELECT**

WRITE FIFO 02 — REG A8, R/W

Bit	7	6	5	4	3	2	1	0
	WFF_CNTRL_[23:16]							

Bit	Name	Function
7-0	WFF_CNTRL_[23:16]	**Write FIFO Buffer A Safe Guard Address:** Safe guard address buffer A [7:0], Mapping to 32bits width data bus field.

WRITE FIFO 03 — REG A9, R/W

Bit	7	6	5	4	3	2	1	0
	WFF_CNTRL_[31:24]							

Bit	Name	Function
7-0	WFF_CNTRL_[31:24]	**Write FIFO Buffer A Safe Guard Address:** Safe guard address buffer A [15:8], Mapping to 32bits width data bus field.

WRITE FIFO 04 — REG AA, R/W

Bit	7	6	5	4	3	2	1	0
	WFF_CNTRL_[39:38]	Reserved		WFF_CNTRL_[36:32]				

Bit	Name	Function
4-0	WFF_CNTRL_[36:32]	**Write FIFO Buffer A Safe Guard Address [20:16]** Safe guard address buffer A [20:16], Mapping to 32bits width data bus field.
5	Reserved	
6	WFF_CNTRL_[38]	**LINE FLIP** When set 1, line signal will be invert, set 0 it is normal.
7	WFF_CNTRL_[39]	**WRITE FIFO FULL STATUS CONTROL** When this bit set 1, write FIFO full status will control capture, play back and read FIFO request, set 0, no control.

WRITE FIFO 05 — REG AB, R/W

Bit	7	6	5	4	3	2	1	0
	WFF_CNTRL_[47:40]							

Bit	Name	Function
7-0	WFF_CNTRL_[47:40]	**Write FIFO Buffer B Safe Guard Address:** Safe guard address buffer B [7:0], Mapping to 32bits width data bus field.

WRITE FIFO 06 — REG AC, R/W

Bit	7	6	5	4	3	2	1	0
	WFF_CNTRL_[55:48]							

Bit	Name	Function
7-0	WFF_CNTRL_[55:48]	**Write FIFO Buffer B Safe Guard Address:** Safe guard address buffer B [15:8], Mapping to 32bits width data bus field.

WRITE FIFO 07 — REG AD, R/W

Bit	7	6	5	4	3	2	1	0
	Reserved			WFF_CNTRL_[60:56]				

Bit	Name	Function
4-0	WFF_CNTRL_[60:56]	**Write FIFO Buffer B Safe Guard Address [20:16]** Safe guard address buffer B [20:16], Mapping to 32bits width data bus field.
7-5	Reserved	.

WRITE FIFO 08 — REG 7C, R/W

Bit	7	6	5	4	3	2	1	0
	WFF_NEW_CNTRL_[7:0]							

Bit	Name	Function
7-0	WFF_NEW_CNTRL_[7:0]	**WRITE FIFO STATUS REGISTER**

READ FIFO REGISTERS

READ FIFO 00 — REG AE, RO

Bit	7	6	5	4	3	2	1	0
	RFF_CNTRL_[7:0]							

Bit	Name	Function
7-0	RFF_CNTRL_[7:0]	CRC test result output (read only)

READ FIFO 01 — REG AF, R/W

Bit	7	6	5	4	3	2	1	0
	RFF_CNTRL_[15:8]							

Bit	Name	Function
3-0	RFF_CNTRL_[11:8]	Read buffer page select from 1 to 16

BIT[3:0]	Read buffer page
0	1
1	2
2	3
3	4
4	5
5	6
6	7
7	8
8	9
9	10
A	11
B	12
C	13
D	14
E	15
F	16

4	RFF_CNTRL_[12]	Enable read FIFO address add by 2: Default 0 for added by 1 When set 1, read FIFO address will count by 2, When set 0, read FIFO address will count by 1.
6-5	RFF_CNTRL_[14:13]	Enable read FIFO request mode

RFF_CNTRL	RFFHREQ	RFFLREQ
00	0	Low request
01	0	High request
10	Low request	0
11	High request	Low request

7	RFF_CNTRL_[15]	Enable Read FIFO When set 1, read FIFO will be turned on; When set 0, read FIFO will be turned off.

READ FIFO 02 — REG B0, R/W

Bit	7	6	5	4	3	2	1	0
	Reserved			RFF_CNTRL_[22:16]				

Bit	Name	Function
3-0	RFF_CNTRL_[21:16]	**Master line flag [5:0]** Read FIFO policy master value: This field will define FIFO high request timing.
4	RFF_CNTRL_[22]	**READ FIFO LOW REQUEST CUT ENABLE**
7-5	Reserved	

READ FIFO 03 — REG B1, R/W

Bit	7	6	5	4	3	2	1	0
	RFF_CNTRL_[31]	RFF_CNTRL_[30]	RFF_CNTRL_[29:24]					

Bit	Name	Function
5-0	RFF_CNTRL_[29:24]	**General line flag [5:0]** Read FIFO policy master value: This field will define FIFO low request timing.
6	RFF_CNTRL_[30]	**LINE FLIP** When set 1, line signal will be invert, set 0 it is normal
7	RFF_CNTRL_[31]	**READ FIFO READ TWO FIELD YUV ENABLE** When this bit set 1, read FIFO will be two field YUV, one field the last frame, the other field the last before frame, set 0, will read two frame Y,

READ FIFO AND WRITE FIFO SHARED REGISTERS

READ FIFO AND WRITE FIFO SHARED 00 — REG B2, R/W

Bit	7	6	5	4	3	2	1	0
	RWFF_ADR_CNTRL_[7:0]							

Bit	Name	Function
7-0	RWFF_ADR_CNTRL_[7:0]	**Read FIFO AND Write FIFO START Address buffer A** Start address buffer A [7:0], Mapping to 32bits width data bus field.

READ FIFO AND WRITE FIFO SHARED 01 — REG B3, R/W

Bit	7	6	5	4	3	2	1	0
	RWFF_ADR_CNTRL_[15:8]							

Bit	Name	Function
7-0	RWFF_ADR_CNTRL_[15:8]	**Read FIFO AND Write FIFO START Address Buffer A** Start address buffer A [15:8], Mapping to 32bits width data bus field.

READ FIFO AND WRITE FIFO SHARED 02 — REG B4, R/W

Bit	7	6	5	4	3	2	1	0
	RWFF_ADR_CNTRL_[23:22]		Reserved	RWFF_ADR_CNTRL_[20:16]				

Bit	Name	Function
4-0	RWFF_ADR_CNTRL_[20:16]	**Read FIFO and Write FIFO START Address Buffer A [20:16]** Start address buffer A [20:16], Mapping to 32bits width data bus field.
5	Reserved	
7-6	RWFF_ADR_CNTRL_[23:22]	**Read FIFO test bus select** Read FIFO test bus select low 2 bits (high in B7[6])

READ FIFO AND WRITE FIFO SHARED 03 — REG B5, R/W

Bit	7	6	5	4	3	2	1	0
	RWFF_ADR_CNTRL_[31:24]							

Bit	Name	Function
7-0	RWFF_ADR_CNTRL_[31:24]	**Read FIFO AND Write FIFO START Address Buffer B** Start address buffer B [7:0], Mapping to 32bits width data bus field.

READ FIFO AND WRITE FIFO SHARED 04 — REG B6, R/W

Bit	7	6	5	4	3	2	1	0
	RWFF_ADR_CNTRL_[39:32]							

Bit	Name	Function
7-0	RWFF_ADR_CNTRL_[39:32]	**Read FIFO AND Write FIFO START Address Buffer B** Start address buffer B [15:8], Mapping to 32bits width data bus field.

READ FIFO AND WRITE FIFO SHARED 05 — REG B7, R/W

Bit	7	6	5	4	3	2	1	0
	RWFF_ADR_CNTRL_[47:46]		Reserved	RWFF_ADR_CNTRL_[44:40]				

Bit	Name	Function
4-0	RWFF_ADR_CNTRL_ [44:40]	**Read FIFO AND Write FIFO START Address [20:16]** Start address buffer B [20:16], Mapping to 32 bits width data bus field.
5	RESERVED	
6	RWFF_ADR_CNTRL_[46]	**Read FIFO Test BUS Select** Read FIFO test bus select high 1 bit (low in b4 [7:6])
7	RWFF_ADR_CNTRL_[47]	**To select read FIFO CRC output** 0 select read FIFO CRC result high 4 bits, CRC long counter and CRC enable. 1 select read FIFO CRC result low 8 bits.

READ FIFO AND WRITE FIFO SHARED 06 — REG B8, R/W

Bit	7	6	5	4	3	2	1	0
	RWFF_ADR_CNTRL_[55:48]							

Bit	Name	Function
7-0	RWFF_ADR_CNTRL_[55:48]	**Read FIFO and Write FIFO offset:** Offset [7:0] will determine next line start address, Mapping to 64bits width data bus field.

READ FIFO AND WRITE FIFO SHARED 07 — REG B9, R/W

Bit	7	6	5	4	3	2	1	0
	Reserved		RWFF_ADR_CNTRL_[61:60]		Reserved		RWFF_ADR_CNTRL_[57:56]	

Bit	Name	Function
1-0	RWFF_ADR_CNTRL_[57:56]	**READ FIFO AND WRITE FIFO OFFSET [9:8]** Offset [9:8], will determine next horizontal line start address. Mapping to 64 bits width data bus field.
3-2	Reserved	
5-4	RWFF_ADR_CNTRL_[61:60]	**Fetch number [9:8] (READ FIFO USE ONLY)** This will determine to fetch the number of pixels from memory each horizontal line. Mapping to 64bits width data bus field.
7-6	RESERVED	

READ FIFO AND WRITE FIFO SHARED 08 — REG BA, R/W

Bit	7	6	5	4	3	2	1	0
	RWFF_ADR_CNTRL_[71:64]							

Bit	Name	Function
7-0	RWFF_ADR_CNTRL_[71:64]	**Fetch number [7:0] (READ FIFO USE ONLY)** This will determine to fetch the number of pixels from memory each horizontal line. Mapping to 64bits width data bus field.

VIDEO PROCESSOR REGISTERS

VDS_PROC 00 **REG C0, R/W**

Bit	7	6	5	4	3	2	1	0
	HSYNC_RST[3:0]				FIELD_FLIP	FIELD_EN	DFIELD_EN	SYNC_EN

Bit	Name	Function
0	SYNC_EN	**External sync enable, active high** This bit enable sync lock mode. flock_en (D0[2]) / sync_en / VDS timing: 0 / 0 / Free run 0 / 1 / Sync lock 1 / X / Frame lock
1	FIELDAB_EN	**ABAB double field mode enable** In field double mode, when this bit is 1, VDS works in ABAB mode, otherwise it works in AABB mode.
2	DFIELD_EN	**Double field mode enable active high** This bit enable field double mode, ex, frame rate from 50Hz to 100Hz, or from 60Hz to 120Hz. When this bit is 1, the output timing is interlaced.
3	FIELD_FLIP	**Flip field control.** This bit is field flip control bit, it only used in interlace mode. When it is 1, it inverts the output field.
7-4	HSYNC_RST[3:0]	**Internal Horizontal period control bit [3:0], Half of total pixels in field double mode.** This field contains horizontal total value minus 1. EX: Horizontal pixels is A, then HSYNC_RST[9:0] = A-1, in field double mode, HSYNC_RST[9:0] = (A/2 –1)

VDS_PROC 01 **REG C1, R/W**

Bit	7	6	5	4	3	2	1	0
	VSYNC_RST[1:0]		HSYNC_RST[9:4]					

Bit	Name	Function
5-0	HSYNC_RST[9:4]	**Internal Horizontal period control bit[9:4], Half of total pixels in field double mode.** This field contains horizontal total value minus 1. EX: Horizontal pixels is A, then HSYNC_RST[9:0] = A-1, in field double mode, HSYNC_RST[9:0] = (A/2 –1)
7-6	VSYNC_RST[1:0]	**Internal Vertical period control bit[1:0]** This field contains vertical total value minus 1.

VDS_PROC 02 — REG C2, R/W

Bit	7	6	5	4	3	2	1	0
	VSYNC_RST[9:2]							

Bit	Name	Function
7-0	VSYNC_RST[9:2]	**Internal Vertical period control bit[9:2]** This field contains vertical total value minus 1.

VDS_PROC 03 — REG C3, R/W

Bit	7	6	5	4	3	2	1	0
	HB_ST[7:0]							

Bit	Name	Function
7-0	HB_ST[7:0]	**Horizontal blanking start position control bit[7:0]** This field is used to program horizontal blanking start position, this blanking is used to get data from memory.

VDS_PROC 04 — REG C4, R/W

Bit	7	6	5	4	3	2	1	0
	HB_SP[5:0]						HB_ST[9:8]	

Bit	Name	Function
1-0	HB_ST[9:8]	**Horizontal blanking start position control bit[9:8]** This field is used to program horizontal blanking stop position, this blanking is used to get data from memory.
7-2	HB_SP[5:0]	**Horizontal blanking stop position control bit[5:0]** This field is used to program horizontal blanking stop position, this blanking is used to get data from memory.

VDS_PROC 05 — REG C5, R/W

Bit	7	6	5	4	3	2	1	0
	VB_ST[3:0]				HB_SP[9:6]			

Bit	Name	Function
3-0	HB_SP[9:6]	**Horizontal blanking stop position control bit[9:6]** This field is used to program horizontal blanking stop position, this blanking is used to get data from memory.
7-4	VB_ST[3:0]	**Vertical blanking start position control bit[3:0]** This field is used to program vertical blanking start position.

VDS_PROC 06 — REG C6, R/W

Bit	7	6	5	4	3	2	1	0
	VB_SP[1:0]		VB_ST[9:4]					

Bit	Name	Function
5-0	VB_ST[9:4]	**Vertical blanking start position control bit[9:4]** This field is used to program vertical blanking start position.
7-6	VB_SP[1:0]	**Vertical blanking stop position control bit[1:0]** This field is used to program vertical blanking stop position.

VDS_PROC 07 — REG C7, R/W

Bit	7	6	5	4	3	2	1	0
	VB_SP[9:2]							

Bit	Name	Function
7-0	VB_SP[9:2]	**Vertical blanking stop position control bit[9:2]** This field is used to program vertical blanking stop position.

VDS_PROC 08 — REG C8, R/W

Bit	7	6	5	4	3	2	1	0
	HS_ST [7:0]							

Bit	Name	Function
7-0	HS_ST[7:0]	**Horizontal sync start position control bit [7:0]** This field is used to program horizontal sync start position.

VDS_PROC 09 — REG C9, R/W

Bit	7	6	5	4	3	2	1	0
	HS_SP[5:0]						HS_ST[9:8]	

Bit	Name	Function
1-0	HS_ST[9:8]	**Horizontal sync start position control bit [9:8]** This field is used to program horizontal sync start position.
7-2	HS_SP[5:0]	**Horizontal sync stop position control bit [5:0]** This field is used to program horizontal sync stop position.

VDS_PROC 10 — REG CA, R/W

Bit	7	6	5	4	3	2	1	0
	VS_ST[3:0]				HS_SP[9:6]			

Bit	Name	Function
3-0	HS_SP[9:6]	**Horizontal sync stop position control bit [9:6]** This field is used to program horizontal sync stop position.
7-4	VS_ST[3:0]	**Vertical sync start position control bit [3:0]** This field is used to program vertical sync start position.

VDS_PROC 11 — REG CB, R/W

Bit	7	6	5	4	3	2	1	0
	VS_SP[1:0]		VS_ST[9:4]					

Bit	Name	Function
5-0	VS_ST[9:4]	**Vertical sync start position control bit [9:4]** This field is used to program vertical sync start position.
7-6	VS_SP[1:0]	**Vertical sync stop position control bit [1:0]** This field is used to program vertical sync stop position.

VDS_PROC 12 — REG CC, R/W

Bit	7	6	5	4	3	2	1	0
	VS_SP[9:2]							

Bit	Name	Function
7-0	VS_SP[9:2]	**Vertical sync stop position control bit [9:2]** This field is used to program vertical sync stop position.

VDS_PROC 13 — REG CD, R/W

Bit	7	6	5	4	3	2	1	0
	HSCALE[7:0]							

Bit	Name	Function
7-0	HSCALE[7:0]	**Horizontal scaling up coefficient bit [7:0]** This field indicates the ratio of horizontal scaling up. HSCALE = 1024 * (resolution of input) / (resolution of output) EX: 720*480 → 800*480, HSCALE = 1024 * 720 / 800

VDS_PROC 14 — REG CE, R/W

Bit	7	6	5	4	3	2	1	0
	VSCALE[5:0]						HSCALE[9:8]	

Bit	Name	Function
1-0	HSCALE[9:8]	**Horizontal scaling coefficient bit [9:8]** This field indicates the ratio of scaling up. HSCALE = 1024 * (resolution of input) / (resolution of output) EX: 720 * 480 → 800 * 480, HSCALE = 1024 * 720 / 800
7-2	VSCALE[5:0] / VPK_CGAIN[5:0]	**Vertical scaling up coefficient bit [5:0] / C vertical peaking gain [5:0]** This field indicates the ratio of vertical scaling up. VSCALE = 1024 * (resolution of input / resolution of output) EX: 720*480 → 720*576, VSCALE = 1024 * 480 /576 This field register also used for C vertical peaking high pass signal gain control, its range is: (0~4)x16

VDS_PROC 15 — REG CF, R/W

Bit	7	6	5	4	3	2	1	0
	VB_F_DLY[1:0]		VB_R_DLY[1:0]		VSCALE[9:6]			

Bit	Name	Function
3-0	VSCALE[9:6] / VPK_CORE_LEV[3:0]	**Vertical scaling coefficient bit[9:6] / vertical peaking coring level [3:0]** This field indicates the ratio of vertical scaling up. VSCALE = 1024 * (resolution of input) / (resolution of output) EX: 720 * 480 → 720 * 576, VSCALE = 1024 * 480 / 576 This field register is also used for vertical peaking coring level, bit[1:0] is used for Y, and bit[3:2] is used for C.
5-4	VB_R_DLY[1:0]	**Vertical blanking rising edge delay control bit[1:0]** This field defines the delay value from the vb signal of getting data from mem to the vb signal that final display out after vds_porc process, this field defines the rising edge delay. 00 → 0 VCLK, 01→ 1 VCLK, 10 → 2 VCLKS, 11 → 3 VCLKS.
7-6	VB_F_DLY[1:0]	**Vertical blanking falling edge delay control bit[1:0]** This field defines the delay value from the vb signal of getting data from mem to the vb signal that final display out after vds_porc process, this field defines the falling edge delay. 00 → 0 VCLK, 01→ 1 VCLK, 10 → 2 VCLKS, 11 → 3 VCLKS.

VDS_PROC 17 — REG D0, R/W

Bit	7	6	5	4	3	2	1	0
	FRAME_RST[4:0]					FLOCK_EN	SRESET[1]	SRESET[0]

Bit	Name	Function
0	SRESET[0]	**Software reset for module video_enhance, active high** When this bit is 1, it reset the VDS_PROC internal module ds_video_enhance
1	SRESET[1]	**Software reset for module tri_dithering10_8, active high** When this bit is 1, it reset the VDS_PROC internal module ds_tri_dithering10_8. The low 2 bits will be truncated without dithering when it is set to 1.
2	FLOCK_EN	**Frame lock enable, active high** This bit enables the frame lock mode, when this bit is 1, VDS_PROC output timing will lock with its input timing (from INPUT_FORMATTER) at every 2 or more frames.
7-3	FRAME_RST[4:0]	**Frame reset period control bit [4:0]** This field indicates how many frames VSD_PROC locked at each time, it based on the input vertical sync. EX: FRAME_RST=4, this means VDS_PROC will lock every 5 frames, (This frame number is counts at every input vertical sync, the frame number of VDS_PROC output maybe different)

VDS_PROC 18 **REG D1, R/W**

Bit	7	6	5	4	3	2	1	0
	DELAY1	DELAY0	UV_FLIP	RESVERD	FRAME_RST[8:5]			

Bit	Name	Function
3-0	FRAME_RST[8:5]	**Frame reset period control bit [4:0]** This field indicates how many frames VSD_PROC locked at each time, it based on the input vertical sync. EX: FRAME_RST=4, this means VDS_PROC will lock every 5 frames, (This frame number is counts at every input vertical sync, the frame number of VDS_PROC output maybe different)
4	RESVERD	
5	UV_FLIP	**UV flip control** This bit is used to flip UV, when this bit is 1, UV position will be flipped.
6	DELAY0	**UV 422 to 444 convert U delay** When this bit is 1, U will delay 1 clock, otherwise, no delay for internal pipe.
7	DELAY1	**UV 422 to 444 convert V delay** When this bit is 1, V will delay 1 clock, otherwise, no delay for internal pipe.

VDS_PROC 19 **REG D2, R/W**

Bit	7	6	5	4	3	2	1	0
	STEP_SEL[1:0]		PTAP2_CNTRL	VSCALE_BYPS	HSCALE_BYPS	Y_DELAY[1:0]		TAP6_BYPS

Bit	Name	Function
0	TAP6_BYPS	**Tap6 filter in 422 to 444 conversion bypass control, active high** This bit is the UV interpolation filter enable control; when this bit is 1, UV bypass the filter.
2-1	Y_DELAY[1:0]	**Y delay control bit [1:0] in 422 to 444 conversion** To compensation the pipe of UV, program this field can delay Y from 1 to 4 clocks.
3	HSCALE_BYPS	**Horizontal scale up bypass control, active high** When this bit is 1, data will bypass horizontal scale up process phase adjustment; When this bit is 0, data will do phase adjustment.
4	VSCALE_BYPS	**Vertical scale up bypass control, active high** When this bit is 1, data will bypass vertical scale up process phase adjustment When this bit is 0, data will do phase adjustment.
5	UV_STEP_BYPS	**UV step response bypass control, active high** When this bit is 1, UV data will don't do step response
7-6	STEP_SEL[1:0]	**UV step response data select control bit [1:0]**

Y_DELAY[1]	Y_DELAY[0]	Y delay
0	0	1
0	1	2
1	0	3
1	1	4

STEP_SEL[1]	STEP_SEL[0]	Data selelect
0	0	U/V5 – U/V6
0	1	U/V4 – U/V7
1	0	U/V3 – U/V8
1	1	U/V2 – U/V9

U/V2 is 2 clocks delay of input U/V, UV3 is 3 clocks delay of input U/V, and so on.

VDS_PROC 20 — REG D3, R/W

Bit	7	6	5	4	3	2	1	0
	SDELAY[1:0]		SVM_BPF_CNTRL[1:0]		STEP_GAIN[3:0]			

Bit	Name	Function
3-0	STEP_GAIN[3:0]	**Step response gain control bit [3:0]** This field register can adjust the UV edge improvement, the larger value of this register, the sharper edge will appear, the range of this gain is (0~4)*4.
5-4	SVM_BPF_CNTRL[1:0]	**SVM data generation select control [1:0]** SVM_BPF_CNTRL[1] / SVM_BPF_CNTRL[0] / SVM data 0 / 0 / a0-a4 0 / 1 / a1-a4 1 / 0 / a2-a4 1 / 1 / a3-a4 A1 is one pipe delay of a0, a2 is one pipe delay of a1, a3 is one pipe delay of a2, a4 is one pipe delay of a3, here a* is the input data y for generate SVM signal.
7-6	SDELAY[1:0]	**To match YUV pipe, SVM data delay by VCLK control bit [1:0]** This field define the SVM compensation delay from 1 to 4 VCLKs SDELAY[1] / SDELAY[0] / SVM delay(VCLK) 0 / 0 / 1 0 / 1 / 2 1 / 0 / 3 1 / 1 / 4

VDS_PROC 21 — REG D4, R/W

Bit	7	6	5	4	3	2	1	0
	SVM_GAIN[7:0]							

Bit	Name	Function
7-0	SVM_GAIN[7:0]	**SVM gain control bit[7:0]** This field contains the gain value of SVM data., its range is (0~16)*16

VDS_PROC 22 — REG D5, R/W

Bit	7	6	5	4	3	2	1	0
	SVM_OFFSET [7:0]							

Bit	Name	Function
7-0	SVM_OFFSET[7:0]	**SVM offset control bit [7:0]** This field contains the offset value of SVM data, its range is 0~255.

VDS_PROC 23 — REG D6, R/W

Bit	7	6	5	4	3	2	1	0
	USER_MIN[4:1]				AUTO_DET_EN	S_SIGMOID_BYPS	SVM_FLIP	SVM_2ND_BYPS

Bit	Name	Function
0	SVM_2ND_BYPS	**2nd order SVM signal generation bypass, active high** When this bit is 1, SVM signal is 1st order, otherwise, it is 2nd order derivative signal.
1	SVM_FLIP	**SVM polarity flip control bit** When this bit is 1, the SVM signal's polarity will be flipped, otherwise, SVM remains the original phase.
2	S_SIGMOID_BYPS	**SVM bypass the sigmoid function, active high** When this bit is 1, SVM signal bypass a sigmoid function. This function can make the SVM signal sharper.
3	AUTO_DET_EN	**Y minimum and maximum level auto detection enable, active high** This bit is the Y min and max auto detection enable bit for black/white level expansion, when this bit is 1, the min and max value of Y in every frame will be detected, otherwise, the min and max value are defined by register.
7-4	USER_MIN[4:1]	**Programmable minimum value control bit [4:1]** This field is the user defined min value for black level expansion.

VDS_PROC 24 — REG D7, R/W

Bit	7	6	5	4	3	2	1	0
	BLACK_LEV[3:0]				USER_MAX[7:4]			

Bit	Name	Function
3-0	USER_MAX [7:4]	**Programmable maximum value control bit [7:4]** This field is the user defined max value for black level expansion.
7-4	BLACK_LEV [3:0]	**Black level expansion level control bit [3:0]** This field defines the black level expansion threshold level value, data larger than this level will have no black level expansion process.

VDS_PROC 25 **REG D8, R/W**

Bit	7	6	5	4	3	2	1	0
	BCORE_NKL[0]	HPF_BYPS	LPF_BYPS	BLEV_BYPS	BLACK_LEV[7:4]			

Bit	Name	Function
3-0	BLACK_LEV[7:4]	**Black level expansion level control bit [7:4]** This field defines the black level expansion threshold level value; data larger than this level will have no black level expansion process.
4	BLEV_BYPS	**Black level expansion bypass control, active high** This bit is the bypass control bit of black level expansion, when it is 1, data will bypass black level expansion process.
5	LPF_BYPS	**Peaking low pass filter bypass control, active high** When this bit is 1, data will bypass the low-pass filter in peaking process.
6	HPF_BYPS	**Peaking high pass filter bypass control, active high** When this bit is 1, data will bypass the high-pass filter in peaking process.
7	BCORE_NKL[0]	**Band pass coring noise level control bit [0]** This field defines the coring noise threshold level value of band-pass component in peaking process.

VDS_PROC 26 **REG D9, R/W**

Bit	7	6	5	4	3	2	1	0
	HCORE_NKL[0]	BCORE_NKL[7:1]						

Bit	Name	Function
6-0	BCORE_NKL[7:1]	**Band pass coring noise level control bit [7:1]** This field defines the coring noise threshold level value of band-pass component in peaking process.
7	HCORE_NKL[0]	**High pass coring noise level control bit [0]** This field defines the coring noise threshold level value of high-pass component in peaking process.

VDS_PROC 27 — REG DA, R/W

Bit	7	6	5	4	3	2	1	0
	CORE_BYPS0	HCORE_NKL[7:1]						

Bit	Name	Function
6-0	HCORE_NKL[7:1]	**High pass coring noise level control bit [7:1]** This field defines the coring noise threshold level value of high-pass component in peaking process.
7	CORE_BYPS0	**Band pass coring bypass control, active high** This bit is the bypass control bit of peaking band-pass coring process for peaking, when this bit is 1, band-pass component will bypass this coring process for peaking.

VDS_PROC 28 — REG DB, R/W

Bit	7	6	5	4	3	2	1	0
	PEAK_B_GAIN[7:0]							CORE_BYPS1

Bit	Name	Function
0	CORE_BYPS1	**High pass coring bypass control, active high** This bit is the bypass control bit of peaking high-pass coring process for peaking, when this bit is 1, high-pass component will bypass this coring process for peaking.
7-1	PEAK_B_GAIN[7:0]	**Band-pass signal gain control bit [6:0]** This field contains band-path signal gain control in peaking process, its range is (0 ~ 16)*16.

VDS_PROC 29 — REG DC, R/W

Bit	7	6	5	4	3	2	1	0
	Y_GAIN[5:0]						PEAK_BYPS	PEAK_B_GAIN[7]

Bit	Name	Function
0	PEAK_B_GAIN[7]	**Peaking B gain control bit [7]** This field contains band-path signal gain control in peaking process, its range is (0 ~ 16)*16.
1	PEAK_BYPS	**Peaking bypass control, active high** When this bit is 1, data (Y) will bypass the peaking process.
7-2	Y_GAIN[5:0]	**Y dynamic range expansion gain control bit [5:0]** This field contains the Y gain value in dynamic range expansion process, its range is (0 ~ 2)*128.

VDS_PROC 30 — REG DD, R/W

Bit	7	6	5	4	3	2	1	0
	UCOS_GAIN[5:0]						Y_GAIN[7:6]	

Bit	Name	Function
1-0	Y_GAIN[7:6]	**Y dynamic range expansion gain control bit [7:6]** This field contains the Y gain value in dynamic range expansion process, its range is (0 ~ 2)*128.
7-2	UCOS_GAIN[5:0]	**U dynamic range expansion cos gain control bit [5:0]** This field contains the U gain value in dynamic range expansion process, its range is (-4 ~ 4)*32.

VDS_PROC 31 — REG DE, R/W

Bit	7	6	5	4	3	2	1	0
	VCOS_GAIN[5:0]						UCOS_GAIN[7:6]	

Bit	Name	Function
1-0	UCOS_GAIN[7:6]	**U dynamic range expansion gain control bit [7:6]** This field contains the U gain value in dynamic range expansion process, its range is (-4 ~ 4)*32.
7-2	VCOS_GAIN[5:0]	**V dynamic range expansion gain control bit [5:0]** This field contains the V gain value in dynamic range expansion process, its range is (-4 ~ 4)*32.

VDS_PROC 32 — REG DF, R/W

Bit	7	6	5	4	3	2	1	0
	Y_OFST [5:0]						VCOS_GAIN [7:6]	

Bit	Name	Function
1-0	VCOS_GAIN[7:6]	**V dynamic range expansion gain control bit [7:6]** This field contains the V gain value in dynamic range expansion process, its range is (-4 ~ 4)*32**.**
7-2	Y_OFST[5:0]	**Y dynamic range expansion offset control bit [5:0]** This field contains the Y offset value in dynamic range expansion process, its range is –128 ~ 127.

VDS_PROC 33 — REG E0, R/W

Bit	7	6	5	4	3	2	1	0
	U_OFST [5:0]						Y_OFST [7:6]	

Bit	Name	Function
1-0	Y_OFST[7:6]	**Y dynamic range expansion offset control bit [7:6]** This field contains the U offset value in dynamic range expansion process, its range is –128 ~ 127.
7-2	U_OFST[5:0]	**U dynamic range expansion offset control bit [5:0]** This field contains the U offset value in dynamic range expansion process, its range is –128 ~ 127.

VDS_PROC 34 — REG E1, R/W

Bit	7	6	5	4	3	2	1	0
	V_OFST [5:0]						U_OFST [7:6]	

Bit	Name	Function
1-0	U_OFST [7:6]	**U dynamic range expansion offset control bit [7:6]** This field contains the U offset value in dynamic range expansion process, its range is –128 ~ 127.
7-2	V_OFST [5:0]	**V dynamic range expansion offset control bit [5:0]** This field contains the V offset value in dynamic range expansion process., its range is –128 ~ 127.

VDS_PROC 35 — REG E2, R/W

Bit	7	6	5	4	3	2	1	0
	SYNC_LEV [5:0]						V_OFST [7:6]	

Bit	Name	Function
1-0	V_OFST [7:6]	**V dynamic range expansion offset control bit [7:6]** This field contains the V offset value in dynamic range expansion process, its range is –128 ~ 127.
7-2	SYNC_LEV [5:0]	**Sync level bit [5:0]** This field contains the composite sync level value, this value will add on Y, outside the composite sync interval. If the Y out is 1V, sync is 0.3V, then this value is (0.3/1)*1024=307, and the output sync's max voltage is 0.5V.

VDS_PROC 36 — REG E3, R/W

Bit	7	6	5	4	3	2	1	0
	D_SP[2:0]			DYN_BYPS	CONVT_BYPS	SYNC_LEV[8:6]		

Bit	Name	Function
2-0	SYNC_LEV[8:6]	**Sync level bit [8:6]** This field contains the composite sync level value, this value will add on Y, outside the composite sync interval.
3	CONVT_BYPS	**YUV to RGB color space conversion bypass control, active high** When this bit is 1, YUV data will bypass the YUV to RGB conversion, the output will still be YUV data. When this bit is 0, YUV data will do YUV to RGB conversion, the output will be RGB data.
4	DYN_BYPS	**Dynamic range expansion bypass control, active high** When this bit is 1, data will bypass the dynamic range expansion process.
7-5	D_SP[2:0]	**Line buffer write reset position control bit [2:0]** This field contains the write reset position of the line buffer, this position is also the write start position of the buffer.

VDS_PROC 37 — REG E4, R/W

Bit	7	6	5	4	3	2	1	0
	D_IREG_BYPS	D_SP [9:3]						

Bit	Name	Function
6-0	D_SP [9:3]	**Line buffer write reset position control bit [9:3]** This field contains the write reset position of the line buffer, this position is also the write start position of line buffer.
7	D_IREG_BYPS	**Line buffer output data delay bypass control, active high** When this bit is 0, output data (to line buffer) will triggered by falling edge clock, When this bit is 1, the output data will bypass this falling edge clock delay.

VDS_PROC 38 — REG E5, R/W

Bit	7	6	5	4	3	2	1	0
	PEAK_H_GAIN[4:0]					INT_BYPS	RAM_BYPS	D_DREG_BYPS

Bit	Name	Function
0	D_DREG_BYPS	**Line buffer input data delay bypass control, active high** When this bit is 0, input data (from line buffer) will triggered by falling edge clock, When this bit is 1, the input data will bypass this falling edge clock delay.
1	RAM_BYPS	**Line buffer one line delay data bypass, active high** When this bit is 1, data will bypass the line buffer.
2	INT_BYPS	**Tap6 interpolation filter bypass control, active high** When this bit is 1, data will bypass the tap6 2x interpolation filter.
7-3	PEAK_H_GAIN[4:0]	**Peaking high-pass signal gain control bit [4:0]** This field contains high-path signal gain control in peaking process, its range is (0 ~ 16)*16.

VDS_PROC 39 — REG E6, R/W

Bit	7	6	5	4	3	2	1	0
	OUT_DREG_BYPS	IN_DREG_BYPS	WEN_DELAY[1:0]		DITHER_BYPS	PEAK_H_GAIN[7:5]		

Bit	Name	Function
2-0	PEAK_H_GAIN[7:5]	**Peaking high-pass signal gain control bit [7:5]** This field contains high-path signal gain control in peaking process, its range is (0 ~ 16)*16.
3	DITHER_BYPS	**Dithering feedback bits number selection** When this bit is 1, data will be processed with 8 bits. When setting this bit 0, data will be processed with 6 bits internally.
5-4	WEN_DELAY[1:0]	**Compensation delay control bit [1:0] for horizontal write enable** This two-bit register defines the compensation delay of horizontal scale up write enable and phase.

WEN_DELAY[1]	WEN_DELAY[0]	Delay (VCLK)
0	0	1
0	1	2
1	0	3
1	1	4

6	IN_DREG_BYPS	**Input data bypass control, active high** When this bit is 0, input data will triggered by falling edge clock, When this bit is 1, the input data will bypass this falling edge clock delay.
7	OUT_DREG_BYPS	**Output data delay bypass control, active high** When this bit is 0, output data will triggered by falling edge clock, When this bit is 1, the output data will bypass this falling edge clock delay.

VDS_PROC 40 — REG E7, R/W

Bit	7	6	5	4	3	2	1	0
	DIS_HB_ST[7:0]							

Bit	Name	Function
7-0	DIS_HB_ST[7:0]	**Final display horizontal blanking start position control bit [7:0]** This field contains final display horizontal blanking start position control, this blanking is used to clean the output data in blanking.

VDS_PROC 41 — REG E8, R/W

Bit	7	6	5	4	3	2	1	0
	DIS_HB_SP[4:0]					DIS_HB_ST[10:8]		

Bit	Name	Function
2-0	DIS_HB_ST[10:8]	**Final display horizontal blanking start position control bit [10:8]** This field contains final display horizontal blanking start position control, this blanking is used to clean the output data in blanking.
7-3	DIS_HB_SP[4:0]	**Final display horizontal blanking stop position control bit [4:0]** This field contains final display horizontal blanking stop position control, this blanking is used to clean the output data in blanking.

VDS_PROC 42 — REG E9, R/W

Bit	7	6	5	4	3	2	1	0
	S_V2CLK_DLY[1:0]		DIS_HB_SP[10:5]					

Bit	Name	Function
5-0	DIS_HB_SP[10:5]	**Final display horizontal blanking stop position control bit [10:5]** This field contains final display horizontal blanking start position control, this blanking is used to clean the output data in blanking.
7-6	S_V2CLK_DLY[1:0]	**SVM delay be V2CLK control bit [1:0]** This field define the SVM delay from 1 to 4 V2CLKs

S_V2CLK_DLY[1]	S_V2CLK_DLY [0]	SVM delay
0	0	1
0	1	2
1	0	3
1	1	4

VDS_PROC 43 **REG EA, R/W**

Bit	7	6	5	4	3	2	1	0
	VSYNC_RST[10]	HSYNC_RST[10]	SBRST_GEN_EN	TEST_SEL[3:0]				TEST_EN

Bit	Name	Function
0	TEST_EN	**Test enable, active high** This bit is the test bus out enable bit, when this bit is 1, the test bus can output the internal status, and otherwise, the test bus is 0Xaaaa.
4-1	TEST_SEL[3:0]	**Test out select control bit [3:0]** This register is used to select internal status bus to test bus.
5	RESERVED	
6	HSYNC_RST[10]	**Internal Horizontal period control bit [10], Half of total pixels in field double mode.** This field contains horizontal total value minus 1. EX: Horizontal pixels is A, then HSYNC_RST[10:0] = A-1, in field double mode, HSYNC_RST[10:0] = (A/2 –1)
7	VSYNC_RST[10]	**Internal Vertical period control bit [10]** This field contains vertical total value minus 1.

test_sel[3]	test_sel[2]	test_sel[1]	test_sel[0]	bus select
0	0	0	0	td0
0	0	0	1	td1
0	0	1	0	td2
0	0	1	1	td3
0	1	0	0	td4
0	1	0	1	td5
0	1	1	0	td6
0	1	1	1	td7
1	0	0	0	td8
1	0	0	1	td9
1	0	1	0	td10
1	0	1	1	td11
1	1	0	0	td12
1	1	0	1	td13
1	1	1	0	td14
1	1	1	1	td15

VDS_PROC 44 — REG EB, R/W

Bit	7	6	5	4	3	2	1	0
	VS_SP[10]	VS_ST[10]	VB_SP[10]	VB_ST[10]	HS_SP[10]	HS_ST[10]	HB_SP[10]	HB_ST[10]

Bit	Name	Function
0	HB_ST[10]	**Horizontal blanking start position control bit [10]** This field is used to program horizontal blanking start position, this blanking is used to get data from memory.
1	HB_SP[10]	**Horizontal blanking stop position control bit [10]** This field is used to program horizontal blanking stop position, this blanking is used to get data from memory.
2	HS_ST[10]	**Horizontal sync start position control bit [10]** This field is used to program horizontal sync start position.
3	HS_SP[10]	**Horizontal sync stop position control bit [10]** This field is used to program horizontal sync stop position.
4	VB_ST[10]	**Vertical blanking start position control bit [10]** This field is used to program vertical blanking start position.
5	VB_SP[10]	**Vertical blanking stop position control bit [10]** This field is used to program vertical blanking stop position.
6	VS_ST[10]	**Vertical sync start position control bit [10]** This field is used to program vertical sync start position.
7	VS_SP[10]	**Vertical sync stop position control bit [10]** This field is used to program vertical sync stop position.

VDS_PROC 45 — REG EC, R/W

Bit	7	6	5	4	3	2	1	0
	GLB_NOISE[7:0]							

Bit	Name	Function
7-0	GLB_NOISE[7:0]	**Global noise threshold value control bit [7:0]** This field contains the global noise threshold value.

VDS_PROC 46 **REG ED, R/W**

Bit	7	6	5	4	3	2	1	0
	VPK_YGAIN[3:0]				reserved	GLB_NOISE[10:8]		

Bit	Name	Function
2-0	GLB_NOISE[10:8]	**Global noise threshold value control bit [10:8]** This field contains the global noise threshold value.
3	RESERVED	
7-4	VPK_YGAIN[3:0]	**Y vertical peaking gain control bit[3:0]** Y vertical peaking gain control on Y high-pass signal, the range is (0~4)x16.

VDS_PROC 47 **REG EE, R/W**

Bit	7	6	5	4	3	2	1	0
	FR_SELECT[0]	UV_STEP_CLIP[2:0]			VPK_CBYPS	VPK_YBYPS	VPK_YGAIN[5:4]	

Bit	Name	Function
1-0	VPK_YGAIN[5:4]	**Y vertical peaking gain control bit[5:4]** Y vertical peaking gain control on Y high-pass signal, the range is (0~4)x16.
2	VPK_YBYPS	**Y vertical peaking bypass contrl** When this bit is 1, Y don't do vertical peaking. When this bit is 0, and when vscale_byps (D2[4]) is 1, Y will do vertical peaking.
3	VPK_CBYPS	**C vertical peaking bypass contrl** When this bit is 1, C don't do vertical peaking. When this bit is 0, and when vscale_byps (D2[4]) is 1, C will do vertical peaking.
6-4	UV_STEP_CLIP[2:0]	**UV step response clip control bit [2:0]** This filed contains the clip control value of UV step response
7	FR_SELECT[0]	**Frame size select control bit [0]** FR_SELECT[2n+1:2n] is for frame n selection. 0 select VSYNC_RST; 1 select VSYNC_SIZE1; 2 select VSYNC_SIZE2.

VDS_PROC 48 **REG EF, R/W**

Bit	7	6	5	4	3	2	1	0
	FR_SELECT[8:6]			FR_SELECT[5:1]				

Bit	Name	Function
4-0	FR_SELECT[5:1] / R_H_LEV[4:0]	**Frame size select control bit [5:1] / Red high level control bit[4:0]** FR_SELECT[2n+1:2n] is for frame n selection. 0 select VSYNC_RST; 1 select VSYNC_SIZE1; 2 select VSYNC_SIZE2. Red component high level.
7-3	FR_SELECT[8:6] / R_L_LEV[2:0]	**Frame size select control bit [8:6] / Red low level control bit[2:0]** FR_SELECT[2n+1:2n] is for frame n selection. 0 select VSYNC_RST; 1 select VSYNC_SIZE1; 2 select VSYNC_SIZE2. Red component low level.

VDS_PROC 49 **REG F0, R/W**

Bit	7	6	5	4	3	2	1	0
	FR_SELECT[16:14]			FR_SELECT[13:9]				

Bit	Name	Function
4-0	FR_SELECT[13:9] / G_H_LEV[4:0]	**Frame size select control bit [13:9] / Green high level control bit[4:0]** FR_SELECT[2n+1:2n] is for frame n selection. 0 select VSYNC_RST; 1 select VSYNC_SIZE1; 2 select VSYNC_SIZE2. Green component high level.
7-5	FR_SELECT[6:13] /G_L_LEV[2:0]	**Frame size select control bit [13:9] / Green low level control bit[2:0]** FR_SELECT[2n+1:2n] is for frame n selection. 0 select VSYNC_RST; 1 select VSYNC_SIZE1; 2 select VSYNC_SIZE2. Green component low level.

VDS_PROC 50 **REG F1, R/W**

Bit	7	6	5	4	3	2	1	0
	FR_SELECT[24:22]			FR_SELECT[21:17]				

Bit	Name	Function
4-0	FR_SELECT[21:17] / B_H_LEV[4:0]	**Frame size select control bit [21:17] / Blue high level control bit[4:0]** FR_SELECT[2n+1:2n] is for frame n selection. 0 select VSYNC_RST; 1 select VSYNC_SIZE1; 2 select VSYNC_SIZE2. Blue component high level.
7-5	FR_SELECT[24:22] / B_L_LEV[2:0]	**Frame size select control bit [24:22] / Blue low level control bit[2:0]** FR_SELECT[2n+1:2n] is for frame n selection. 0 select VSYNC_RST; 1 select VSYNC_SIZE1; 2 select VSYNC_SIZE2. Blue component low level.

VDS_PROC 51 **REG F2, R/W**

Bit	7	6	5	4	3	2	1	0
	FRAME_NO[0]	FR_SELECT[31:25]						

Bit	Name	Function
6-0	FR_SELECT[31:25] / R_UP_GAIN[5:0] / R_LOW_GAIN[0]	**Frame size select control bit [31:25]** FR_SELECT[2n+1:2n] is for frame n selection. 0 select VSYNC_RST; 1 select VSYNC_SIZE1; 2 select VSYNC_SIZE2. Bit [5:0] is also used for red up gain Bit [6] is also used fro red low gain
7	FRAME_NO[0] / R_LOW_GAIN[1]	**Programmable repeat frame number control bit [0]** This field defines the repeated frame number, EX: if frame_no = 2, then the frame will repeat every 3 frame.

frame_no[3]	frame_no[2]	frame_no[1]	frame_no[0]	repeat num
0	0	0	0	1
0	0	0	1	2
0	0	1	0	3
0	0	1	1	4
0	1	0	0	5
0	1	0	1	6
0	1	1	0	7
0	1	1	1	8
1	0	0	0	9
1	0	0	1	10
1	0	1	0	11
1	0	1	1	12
1	1	0	0	13
1	1	0	1	14
1	1	1	0	15
1	1	1	1	16

This bit is also used for red lowgain.

VDS_PROC 52 **REG F3, R/W**

Bit	7	6	5	4	3	2	1	0
	VSYNC_SIZE1[4:0]					FRAME_NO[3:1]		

Bit	Name	Function
2-0	FRAME_NO[3:1] / R_LOW_GAIN[3:2] / G_UP_GAIN[0]	**Programmable repeat frame NO control bit [3:1]** This field define the repeated frame number, EX, if frame_no = 2, then the frame will repeat every 3 frame. Bit [1:0] is also used for red low gain. Bit [2] is also used for green up gain.
7-3	VSYNC_SIZE1[4:0] /G_UP_GAIN[5:1]	**Programmable vertical total size 1 control bit [4:0]** This field contains the vertical total line number minus 1. It can be the same as vsync_rst and vsync_size2, it also can different with them, and it can be used to define different frame size. Green up gain.

VDS_PROC 53 **REG F4, R/W**

Bit	7	6	5	4	3	2	1	0
	VSYNC_SIZE2[1:0]		VSYNC_SIZE1[10:5]					

Bit	Name	Function
5-0	VSYNC_SIZE1[10:5] / G_LOW_GAIN[3:0] / B_UP_GAIN[1:0]	**Programmable vertical total size 1 control bit [10:5]** This field contains the vertical total line number minus 1. It can be the same as vsync_rst and vsync_size2, it also can different with them, and it can be used to define different frame size. Bit [3:0] is also used for green down gain. Bit [5:4] is also used for blue up gain
7-6	VSYNC_SIZE2[1:0] / B_UP_GAIN[3:2[	**Programmable vertical total size 2 control bit [1:0]** This field contains the vertical total line number minus 1. It can be the same as vsync_rst and vsync_size1, it also can different with them, and it can be used to define different frame size. Blue up gain.

VDS_PROC 54 — REG F5, R/W

Bit	7	6	5	4	3	2	1	0
	VSYNC_SIZE2[9:2]							

Bit	Name	Function
7-0	VSYNC_SIZE2[9:2] / B_UP_GAIN[5:4] / B_LOW_GAIN[3:0]	**Programmable vertical total size 2 control bit [9:2]** This field contains the vertical total line number minus 1. It can be the same as vsync_rst and vsync_size1, it also can different with them, and it can be used to define different frame size. Bit [1:0] is also used for blue up gain. Bit [5:2] is also used for blue low gain.

VDS_PROC 55 — REG F6, R/W

Bit	7	6	5	4	3	2	1	0
	WLEV_BYPS	COLOR_ENH_EN	EN_FID_RST	FID_AA_DLY	FREERUN_FID	EN_FR_NUM_RST	FR_SEL_EN	VSYNC_SIZE2[10]

Bit	Name	Function
0	VSYNC_SIZE2[10]	**Programmable vertical total size 2 control bit [10]** This field contains the vertical total line number minus 1. It can be the same as vsync_rst and vsync_size1, it also can different with them, and it can be used to define different frame size.
1	FR_SEL_EN	**Enable the different frame size, active high** When this bit is 1, VDS_PROC can generate a sequence of different frame size.
2	EN_FR_NUM_RST	**Enable frame number reset, active high** When this bit is 1, frame number will be reset to 1 when frame lock is occur.
3	FREERUN_FID	**Enable internal free run field index generation, active high** When this bit is 1, the output field index is internal free run field, otherwise the output field index is based on input field index.
4	FID_AA_DLY	**Enable internal free run AABB field delay 1 frame, active high** When this bit is 1, the internal free run AABB field will delay 1 frame.
5	EN_FID_RST	**Enable internal free run field index reset, active high** When this bit is 1, internal free run field index will reset at every frame number is 0.
6	COLOR_ENH_EN	**Primary color enhancement enable, active high** When this bit is 1, primary color enhancement is enabled.
7	WLEV_BYPS	**White level expansion bypass control, active high** When this bit is 1, Y don't do white level expansion.

VDS_PROC 56 — REG F7, R/W

Bit	7	6	5	4	3	2	1	0
	BLEV_GAIN[7:0]							

Bit	Name	Function
7-0	BLEV_GAIN[7:0]	**Black level expansion gain control bit [7:0]** This field contains the gain control of black level expansion, its range is (0~16)*16.

VDS_PROC 57 — REG F8, RO

Bit	7	6	5	4	3	2	1	0
	HBOUT	VBOUT	HSOUT	VSOUT	FR_NUM[3:0]			

Bit	Name	Function
3-0	FR_NUM[3:0]	**Frame number** Internal frame number status. It counts every frame.
4	VSOUT	**Vertical sync output** Internal vertical sync period status. When it is 1, it indicates the vertical sync interval.
5	HSOUT	**Horizontal sync output** Internal horizontal sync period status. When it is 1, it indicates the horizontal sync interval.
6	VBOUT	**Vertical blanking output** Internal vertical blanking period status. When it is 1, it indicates the vertical blanking interval.
7	HBOUT	**Horizontal blanking output** Internal horizontal blanking period status. When it is 1, it indicates the horizontal blanking interval.

VDS_PROC 58 — REG F9, RO

Bit	7	6	5	4	3	2	1	0
	0				VERT_COUNT[2:0]			FIDOUT

Bit	Name	Function
0	FIDOUT	**Field flag output** Internal field ID status,. When it is 1, it indicates even field When it is 0, it indicates odd field.
3-1	VERT_COUNT[2:0]	**Vertical counter [2:0]** Internal vertical counter status, it counts every line.
7-4	0	

VDS_PROC 59 — REG FA, RO

Bit	7	6	5	4	3	2	1	0
	VERT_COUNT [10:3]							

Bit	Name	Function
7-0	VERT_COUNT[10:3]	**Vertical counter bit [10:3]** Internal vertical counter status, it counts every line.

VDS_PROC 60 — REG FB, R/W

Bit	7	6	5	4	3	2	1	0
	DS_MIG_OFFSET[6:0]							DS_MIG_EN

Bit	Name	Function
0	DS_MIG_EN	**Motion index generation user mode enable** When this bit is 1, the motion index generation will use mig_offset[3:0] as Motion index.
7-1	DS_MIG_OFFSET[6:0]	**Motion index offset control bit [6:0]** The offset control for motion index generation. When ds_mig_en is 1, ds_mig_offset[3:0] is user-defined motion index.

VDS_PROC 61 — REG FC, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED	DIF_LPF_BYPS	VT_C_BYPS	VT_Y_BYPS	DS_MI_GAIN[3:0]			

Bit	Name	Function
3-0	DS_MI_GAIN[3:0]	**Motion index generation gain control bit [3:0]** Motion index generation gain control, its range is (0~8)*2.
4	VT_Y_BYPS	**Y bypass the noise reduction process control** When this bit is 1, Y data will bypass the noise reduction process.
5	VT_C_BYPS	**C bypass the noise reduction process control** When this bit is 1, UV data will bypass the noise reduction process.
6	DIF_LPF_BYPS	**Bypass control of the tap5 low-pass filter used for Y difference between two frames.** When this bit is 1, Y difference data will bypass the tap5 low-pass filter
7	RESERVED	

VDS_PROC 62 — REG FD, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED	EN_GLB_STILL	RESERVED	ENABLE_NR	MI_THRESH[3:0]			

Bit	Name	Function
3-0	MI_THRESH[3:0]	**Noise reduction threshold value bit [3:0]** Noise-reduction threshold value. When MI is smaller than the threshold value, the noise reduction is enabled. Otherwise it is not.
4	ENABLE_NR	**Noise reduction enable** Noise-reduction enable bit, active high.
5	RESERVED	
6	EN_GLB_STILL	**Global still index enable, active high** This bit enable the global still signal.
7	RESERVED	

VDS_PROC 63 — REG FE, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED		GLB_STILL_MUNU	RESERVED	STILL_NR_GAIN[3:0]			

Bit	Name	Function
3-0	STILL_NR_GAIN[3:0]	**Motion index generation gain control bit [3:0] for still picture** When picture is still, this field contains the motion index generation gain, its range is (0~8)*2.
4	RESERVED	
5	GLB_STILL_MUNU	**Menu mode control for global still index** This bit is the user defined menu mode for global still signal, when it is 1, the global still signal is 1, the following is the detail.
7-6	RESERVED	

GLB_STILL_MENU	EN_GLB_STILL	Global still index
0	0	0
0	1	glb_still
1	x	1

VDS_PROC 64 — REG FF, R/W

Bit	7	6	5	4	3	2	1	0
	UV_BLK_VAL[7:0]							

Bit	Name	Function
7-0	UV_BLK_VAL	**UV blanking amplitude value control bit[7:0]** This filed indicates the amplitude value of UV in blanking interval.

VDS_PROC 65 — REG 7E_00/6C, R/W

Bit	7	6	5	4	3	2	1	0
	USIN_GAIN[7:0]							

Bit	Name	Function
7-0	USIN_GAIN[7:0]	**U dynamic range expansion sin gain control bit [5:0]** This field contains the U sin gain value in dynamic range expansion process, its range is (-4 ~ 4)*32.

VDS_PROC 66 — REG 7E_00/6D, R/W

Bit	7	6	5	4	3	2	1	0
	VSIN_GAIN[7:0]							

Bit	Name	Function
7-0	VSIN_GAIN[7:0]	**V dynamic range expansion sin gain control bit [5:0]** This field contains the V sin gain value in dynamic range expansion process, its range is (-4 ~ 4)*32.

VDS_PROC 67 — REG 7E_00/6E, R/W

Bit	7	6	5	4	3	2	1	0
	WLEV_GAIN[7:0]							

Bit	Name	Function
7-0	WLEV_GAIN[7:0]	**White level expansion gain control bit[7:0]** This field contains the gain control of white level expansion, its range is (0~16)*16.

VDS_PROC 68 — REG 7E_00/6F, R/W

Bit	7	6	5	4	3	2	1	0
	WHITE_LEV[7:0]							

Bit	Name	Function
7-0	WHITE_LEV[7:0]	**White level expansion level control bit[7:0]** This field defines the white level expansion threshold level value; data less than this level will have no white level expansion process.

OSD REGISTERS

OSD_TOP_00 **REG 7E_00/60, R/W**

Bit	7	6	5	4	3	2	1	0
	OSD_MENU	**OSD_EN**	**OSD_VZ[1:0]**		**OSD_HZ[2:0]**			**SRESET**

Bit	Name	Function
0	**SRESET]**	**Software reset for module , active high** When this bit is 1, it reset osd_top module
3-1	**OSD_HZ[2:0]**	**Osd horizontal zoom select**

OSD_HZ[2]	OSD_HZ[1]	OSD_HZ[0]	SIZE
0	0	0	Orignal size
0	0	1	2
0	1	0	3
0	1	1	4
1	0	0	5
1	0	1	6
1	1	0	7
1	1	1	8

Bit	Name	Function
5-4	**OSD_VZ[1:0]**	**Osd vertical zoom select**

OSD_VZ[1]	OSD_VZ[0]	SIZE
0	0	1
0	1	2
1	0	3
1	1	4

Bit	Name	Function
6	**OSD_EN**	**Osd display enable, active high** When this bit is 1, osd can display on screen.
7	**OSD_MENU**	Osd menu display enable, active high When this bit is 1, osd state will jump to menu display state.

OSD_TOP_01 **REG 7E_00/61, R/W**

Bit	7	6	5	4	3	2	1	0
	ICON_MOD[3:0]				MENU_ICON_SEL[3:0]			

Bit	Name	Function
3-0	MENU_ICON_SEL[3:0]	**Osd menu icons select**

SEL[3]	SEL[2]	SEL[1]	SEL[0]	Select icon
0	0	0	1	Brightness icon
0	0	1	0	Contrast icon
0	0	1	1	Hue icon
0	1	0	0	Sound icon
1	0	0	0	Up/down moving icon
1	0	0	1	Left/right moving icon
1	0	1	0	Vertical size icon
1	0	1	1	Horizontal size icon
others				Reserved , if SEL[3:0] = 4'h0, Nothing is selected

Bit	Name	Function
7-4	ICON_MOD[3:0]	**Osd icons modification select**

MOD[3]	MOD[2]	MOD[1]	MOD[0]	Select icon
0	0	0	1	Brightness icon
0	0	1	0	Contrast icon
0	0	1	1	Hue icon
0	1	0	0	Sound icon
1	0	0	0	Up/down moving icon
1	0	0	1	Left/right moving icon
1	0	1	0	Vertical size icon
1	0	1	1	Horizontal size icon
others				Reserved , if MOD[3:0] = 4'h0, Nothing is selected

OSD_TOP_02 **REG 7E_00/62, R/W**

Bit	7	6	5	4	3	2	1	0
	MENU_BAR_BDCOR		MENU_BAR_BGCOR			MENU_BAR_FGCOR		

Bit	Name	Function
2-0	MENU_BAR_FGCOR[2:0]	**Menu font or bar foreground color.** for bar and menu will not display on screen at the same time, so they are shared.
5-3	MENU_BAR_BGCOR[2:0]	**Menu font or bar background color.** for bar and menu will not display on screen at the same time, so they are shared.
7-6	MENU_BAR_BDCOR[1:0]	**Menu or bar border color.** It is the low 2 bits of menu or bar border color, for bar and menu will not display on screen at the same time, so they are shared.

OSD_REG_03 **REG 7E_00/63, R/W**

Bit	7	6	5	4	3	2	1	0
	CMD_STATUS	BAR_ICON_SEL_FGCOR			BAR_ICON_SEL_FGCOR			MENU_BAR_BDCOR[2]

Bit	Name	Function
0	MENU_BAR_BDCOR[2]	**Menu or bar border color.** It is the bit 2 of menu or bar border color.
3-1	BAR_ICON_SEL_FGCOR	**Selected icon or bar's icon foreground color.**
6-4	BAR_ICON_SEL_BGCOR	**Selected icon or bar's icon background color.**
7	CMD_STATUS	**Command finished status** When this bit is 1, it means CPU has finished command and hardware can execute the command, else hardware will do last operation. In order to avoid tearing, When you want to access OSD, pull this bit down first and pull up this bit when you finish programming osd responding registers.

OSD_REG_04 — REG 7E_00/64, R/W

Bit	7	6	5	4	3	2	1	0
	TEST_SEL				NEG_LAT_EN	MODE_SEL	RESERVED	ROW_EN

Bit	Name	Function
0	ROW_EN	**Menu display in row or column mode.** When 1, osd menu displays in row style, else in column style.
2	MODE_SEL	**YCbCr or RGB output.** Osd display in YCbCr or RGB format , when set to 1, display in YCbCr mode
3	NEG_LAT_EN	**V2clk latch osd data with negative enable.** When set to 1, V2CLK clock can latch osd data with negative edge
7-4	TEST_SEL	**Test logic output select.** TEST_SEL[0], test logic output enable, when set to 1, test logic can output. TEST_SEL[3:1] select 8 test logics to test bus.

test_sel[3]	test_sel[2]	test_sel[1]	bus select
0	0	0	td0
0	0	0	td1
0	0	1	td2
0	0	1	td3
0	1	0	td4
0	1	0	td5
0	1	1	td6
0	1	1	td7

OSD_REG_05 — REG 7E_00/65, R/W

Bit	7	6	5	4	3	2	1	0
	MENU_BAR_HORZ_START [7:0]							

Bit	Name	Function
7-0	MENU_BAR_HORZ_START [7:0]	**Menu or bar horizontal start address** The real address is { MENU_BAR_HORZ_START [7:0], 3'h0}.

OSD_REG_06 — REG 7E_00/66, R/W

Bit	7	6	5	4	3	2	1	0
	MENU_BAR_VIRT_START [7:0]							

Bit	Name	Function
7-0	MENU_BAR_ VIRT_START [7:0]	**Menu or bar vertical start address** The real address is { MENU_BAR_VIRT_START [7:0], 3'h0}.

OSD_REG_07 **REG 7E_00/67, R/W**

Bit	7	6	5	4	3	2	1	0
	BRA_TOTLENGTH [7:0]							

Bit	Name	Function
7-0	BRA_TOTLENGTH [7:0]	**BAR DISPLAY TOTAL LENGTH** Bar display on screen's total length, when horizontal zoom is 0.

OSD_REG_08 **REG 7E_00/68, R/W**

Bit	7	6	5	4	3	2	1	0
	BAR_VALUE [7:0]							

Bit	Name	Function
7-0	BAR_VALUE [7:0]	**Bar foreground color value.** The value of this register indicates the real value of icon, such as brightness's value is 8'hf0, and then this register is also programmed to 8'hf0.

ELECTRICAL CHARACTERISTICS

This chapter describes the electrical specifications for the TrueView 5715 TV Display Processor.

ABSOLUTE MAXIMUM RATINGS

Table 3: Absolute Maximum Ratings

Parameter	Symbol	Min	Max	Units	Conditions
3.3V Power Supply Voltage (Reference to Ground)	VCC*	-0.3	4.0	V	
Voltage on any input	V_I	-0.3	VCC+0.3	V	
Storage Temperature	T_S	-40	125	℃	
Operating Temprature	T_o	0	70	℃	

Warning: Stressing the device beyond the "Absolute maximum Ratings" may cause permanent damage.
* TrueView 5715 has four sets of power supplies (all 3.3V), VDD/GND for core, PVDD/PVSS for I/O PAD, PAVD/PAVS for PLL and DAVD/DAVS&DVSS for DAC.

DC CHARACTERISTICS

Table 4: DC Characteristics

Parameter	Symbol	Min	Typical	Max	Units	Conditions
Digital Inputs						
Input High Voltage	V_{IH}	0.8*VDD		VDD	V	
Input Low Voltage	V_{IL}	0		0.2*VDD	V	
Input Leakage Current	I_I			5	uA	
Input Capacitance	C_{IN}		8		pF	
Digital Outputs						
Output High Voltage	V_{OH}	PVDD-1.0			V	IOH=-1.0mA
		PVDD-0.5			V	IOH=-100uA
Output Low Voltage	V_{OL}			0.4	V	IOL=3.0mA
Power Requirements						
Power Supply Voltage (3.3V±5%)	VDD, PVDD, PAVD, DAVD	3.135	3.3	3.465	V	
3.3V Power Supply Current	I_{33d} (VDD&PVDD&PAVD)		280		mA	

AC CHARACTERISTICS

Figure 6: Video Input Port AC Timing

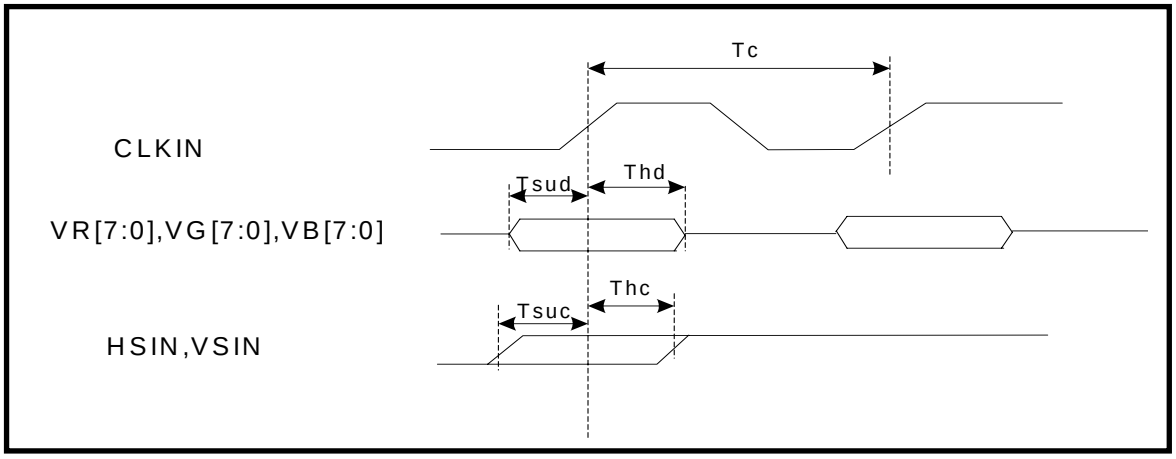


Table 5: Video Input Port AC Timing

Parameter	Symbol	Min	Typical	Max	Units	Conditions
CLKIN Frequency	1/Tc			80	MHz	
VR[7:0], VG[7:0], VB[7:0] Setup Time to CLKIN	Tsuc	5			ns	
VR[7:0], VG[7:0], VB[7:0] Hold Time to CLKIN	Thd	0			ns	
HSIN, VSIN Setup Time to CLKIN	Tsuc	5			ns	
HSIN, VSIN Hold Time to CLKIN	Thc	0			ns	

Figure 7: Video Output Port AC timing

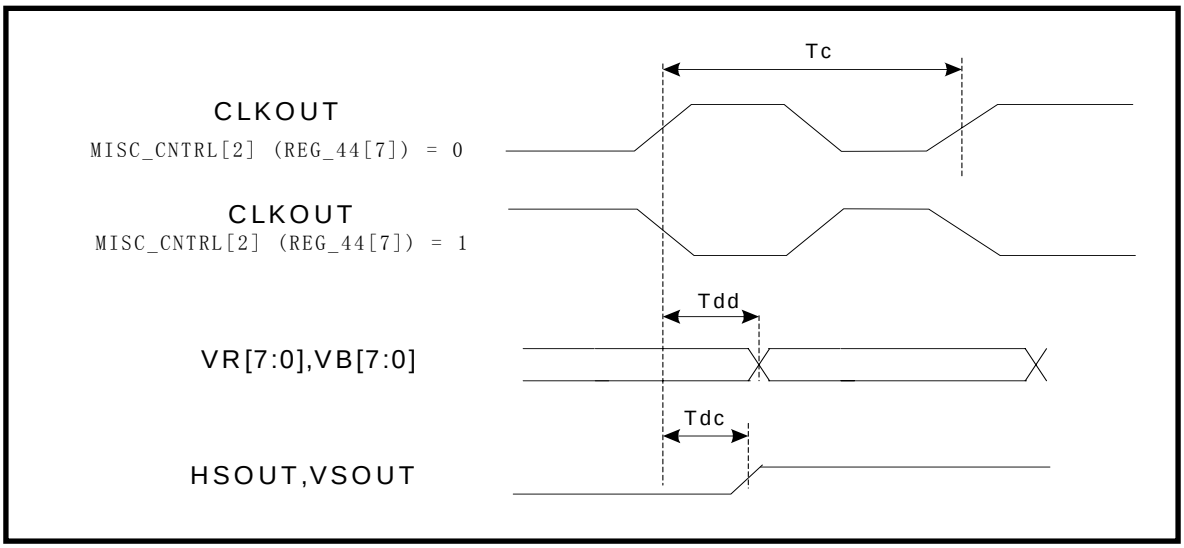


Table 6: Video Output Port AC timing

Parameter	Symbol	Min	Typical	Max	Units	Conditions
VCLKOUT Frequency	**1/Tc**			40.5	MHz	20pF load
VR[7:0], VB[7:0] Delay Time from CLKOUT	Tdd	0		5	ns	20pF load
HSOUT, VSOUT Delay Time from CLKIN	Tdc	0		5.5	ns	20pF load

Figure 8: Memory Interface AC Input timing

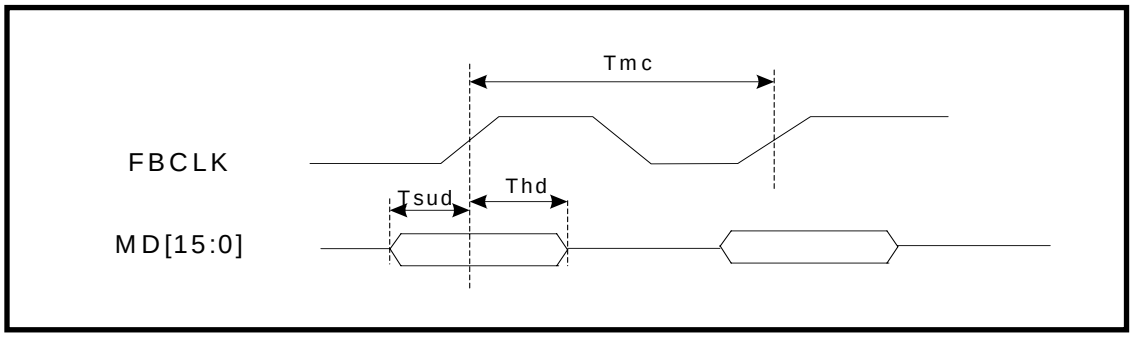


Table 7: Memory Interface AC Input timing

Parameter	Symbol	Min	Typical	Max	Units	Conditions
MD[15:0] Setup Time to FBCLK	Tsuc	1.8		2.6	ns	
MD[15:0] Hold Time to FBCLK	Thd	1.1		3.4	ns	
FBCLK Frequency	1/Tmc			129.6	MHz	

Figure 9: Memory Interface AC output timing

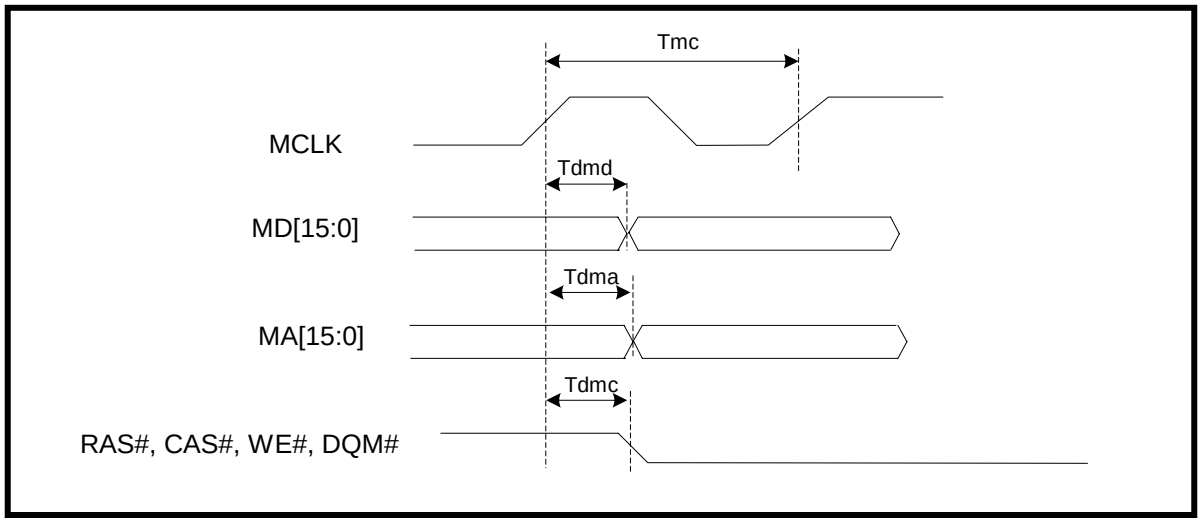
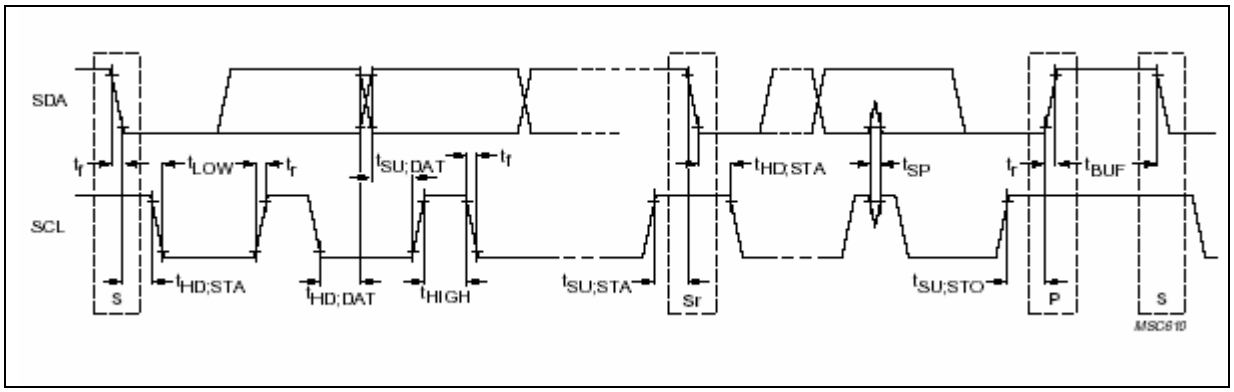


Table 8: Memory Interface AC output timing

Parameter	Symbol	Min	Typical	Max	Units	Conditions
MA[10:0] delay time from MCLK	Tdma	1.3		2.4	ns	20pF load
MD[15:0] delay time from MCLK	Tdmd	1.6		2.3	ns	20pF load
RAS#,CAS#,WE#,DQM# delay time from MCLK	Tdmc	1.0		2.0	ns	20pF load
MCLK Frequency	1/Tmc			129.6	MHz	20pF load

Figure 10: Definition of timing for F/S-mode devices on the I²C-bus



**** Figure 10 Referenced document:***

Philips Semiconductors, "The I²C-BUS Specification, version2.1 ---January 2000"

Page33: Fig.31: Definition of timing for F/S-mode devices on the I²C-bus.

Table 9: Characteristics of the SDA and SCL bus lines for F/S-mode I²C-bus devices

PARAMETER	SYMBOL	STANDARD-MODE		FAST-MODE		UNIT
		MIN.	MAX.	MIN.	MAX.	
SCL clock frequency	f_{SCL}	0	100	0	400	kHz
Hold time (repeated) START condition. After this period, the first clock pulse is generated	$t_{HD;STA}$	4.0	–	0.6	–	µs
LOW period of the SCL clock	t_{LOW}	4.7	–	1.3	–	µs
HIGH period of the SCL clock	t_{HIGH}	4.0	–	0.6	–	µs
Set-up time for a repeated START condition	$t_{SU;STA}$	4.7	–	0.6	–	µs
Data hold time: for CBUS compatible masters for I²C-bus devices	$t_{HD;DAT}$	 5.0 0(2)	 – 3.45(3)	 – 0(2)	 – 0.9(3)	 µs µs
Data set-up time	$t_{SU;DAT}$	250	–	100(4)	–	ns
Rise time of both SDA and SCL signals	t_r	–	1000	$20 + 0.1C_b$(5)	300	ns
Fall time of both SDA and SCL signals	t_f	–	300	$20 + 0.1C_b$(5)	300	ns
Set-up time for STOP condition	$t_{SU;STO}$	4.0	–	0.6	–	µs
Bus free time between a STOP and START condition	t_{BUF}	4.7	–	1.3	–	µs
Capacitive load for each bus line	C_b	–	400	–	400	pF
Noise margin at the LOW level for each connected device (including hysteresis)	V_{nL}	$0.1V_{DD}$	–	$0.1V_{DD}$	–	V
Noise margin at the HIGH level for each connected device (including hysteresis)	V_{nH}	$0.2V_{DD}$	–	$0.2V_{DD}$	–	V

Notes:

1. A device must internally provide a hold time of at least 300 ns for the SDA signal (referred to the V_{IHmin} of the SCL signal) to bridge the undefined region of the falling edge of SCL.
2. The maximum $t_{HD;DAT}$ has only to be met if the device does not stretch the LOW period (t_{LOW}) of the SCL signal.
3. A Fast-mode I²C-bus device can be used in a Standard-mode I²C-bus system, but the requirement $t_{SU;DAT} \geq 250$ ns must then be met. This will automatically be the case if the device does not stretch the LOW period of the SCL signal.If such a device does stretch the LOW period of the SCL signal, it must output the next data bit to the SDA line $t_{r\,max} + t_{SU;DAT} = 1000 + 250 = 1250$ ns (according to the Standard-mode I²C-bus specification) before the SCL line is released.
4. C_b = total capacitance of one bus line in pF.

**** Table 9 Referenced document:***

Philips Semiconductors, "The I²C-BUS Specification, version2.1 ---January 2000"

Page32: Table 5: Characteristics of the SDA and SCL bus lines for F/S-mode I²C-bus devices[1]

DAC

The RGBS (the DAC in the 5715) consists of four identical 8-bit DACs to provide red, green, blue color components and SVM. Each DAC can output a current from 0 to 255 units of current, where one unit of current (LSB) is defined based on the VESA video signal standard.

Functional Operating Range (VDD= 3.3V ± 5%, DAVD / DAVS = 3.3V ± 5%; T_o = 0C to + 70C)

DAC DC CHARACTERISTICS

Table 10: DAC DC Characteristics

Parameter	Min	Typical	Max	Unit	Notes
DAC Resolution/Channel			8	Bits	
The number of channel		4			
INL	-1.0		+1.0	LSB	
DNL	-1.0		+1.0	LSB	
Full Scale (gain) Error	-5.0		+10.0	%	Of Full Scale
DAC Full Scale Voltage	665	700	770	mV	1, 2
DAC current		140	150	mA	3
DAC – to DAC Matching			6	%	
LSB Current		73.2		uA	1
Monotonicity		Guaranteed			

NOTES:

1. It is VESA Video Level. The RIREF in the board is 220ohm. The output load is Double termination required with 75ohm (effective resistance 37.5ohm) and 10pf per channel.
2. For good linearity the full scale voltage should be less than 1.2v. The RIREF in the board is 125ohm. The output load is Double termination required with 75ohm (effective resistance 37.5ohm) and 10pf per channel.
3. It is the current from pin DAVD. For 4 channels full scale is on. The output is 1.2v.

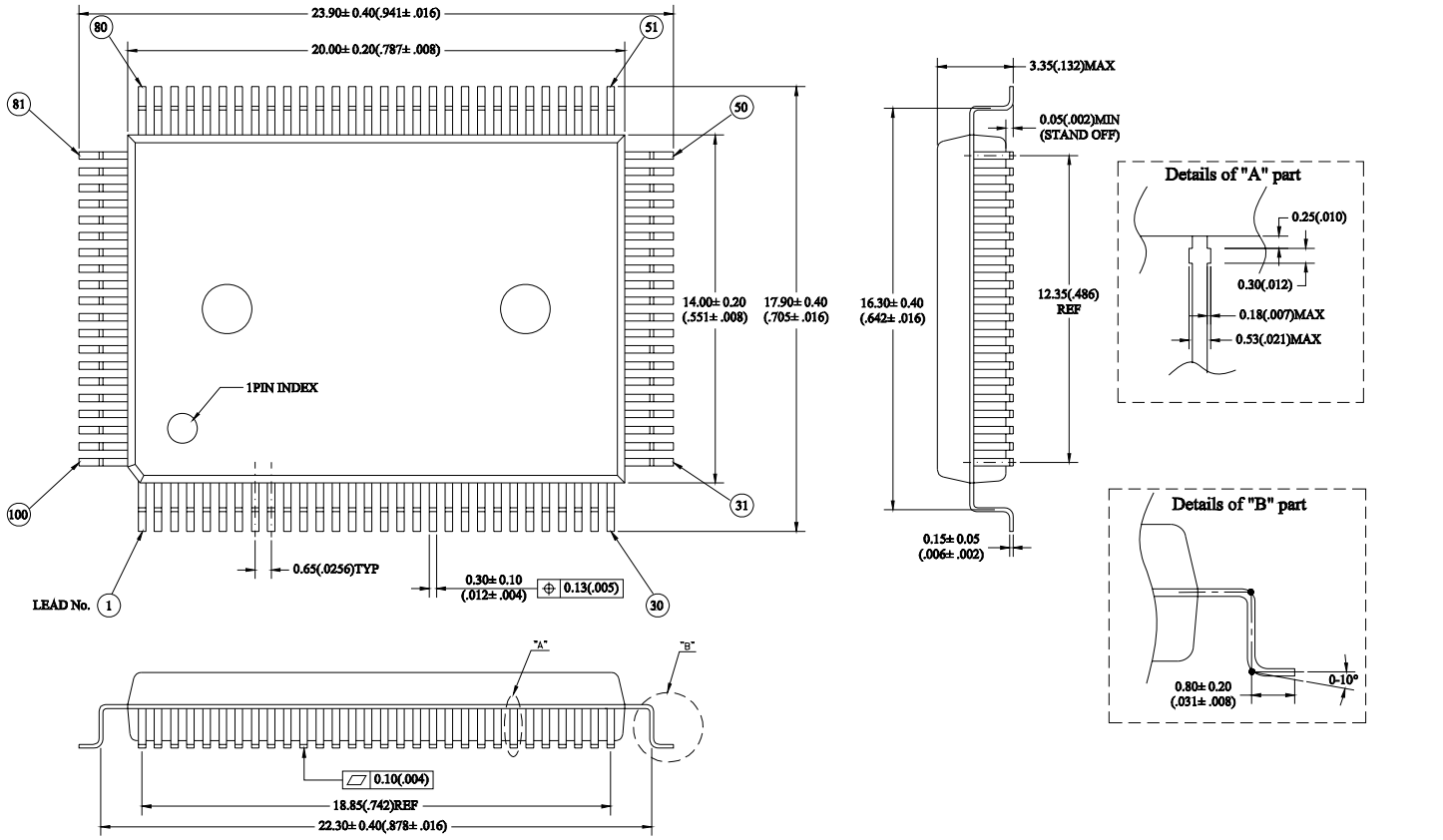
DAC AC CHARACTERISTICS

Table 11: DAC AC Characteristics

Parameter	Min	Typical	Max	Unit	Notes
Pixel Clock Rate			81	MHz	
RGB Video Output Rise Time (10-90% of full-scale)		2	4.5	ns	1
RGB Video Output Fall Time (10-90% of full-scale)		3	5	ns	1

NOTES:
1. As measured when the load is 37.5ohm, 10pf.

Figure 11: Package Dimensions



Contact Information

Tvia, Inc., USA

4001 Burton Drive
Santa Clara, CA 95054

Tel: 408-982-8588
Fax: 408-982-8591
Email: sales@tvia.com

Tvia, Inc., China
1F, No.3 Software Park Building,
No.669 Changjiang Rd.W, Hefei City,
Anhui, P.R.China 230088
Tel: 86-551-5318799
Fax: 86-551-5311620
Email: swang@tvia.com

Tvia, Inc., Shenzhen
D901, The Tenth Kejinan Road,
High-tech Zone,
Nan-Shan District, Shenzhen City,
P.R. China 518057
Phone : 86-755-26935800
Fax : 86-755-26935700
Email: swang@tvia.com

Tvia Inc., Beijing
310, Red Tower, SOHO New Town, No.88 Jianguo Road, Chaoyang District, Beijing, P.R.China 100022
Tel: 86-10-85895276
Fax: 86-10-85895245
Email: swang@tvia.com

Tvia, Inc., Japan & Asia Sales Office
10F, Odakyu Minami-Aoyama Bldg.
7-8-1 Minami-Aoyama
Minato-ku, Tokyo 107-0062
Tel: 813-5766-0676
Fax: 813-3499-3109
Email: mbuttrick@tvia.com