TrueView™5725

Registers Definition Rev1.1

August 2005

Revision History

Date	Version	By	Comments
07/28/2005	Rev 1.0	5725 team	Initial draft
07/28/2005	Rev 1.1	Zaken Chen	Chang register bit name to match txt file

Register Map

Segment / Address	0	1	2	3	4	5
00~0F	Status Register (Read Only) Chapter 00	Input Formatter Chapter 01	De-interlace Chapter 02	Video Processor Chapter 08	Memory Chapter 05	ADC Chapter 11
10~1F						
20~2F					Capture & Playback, Chapter 06	Sync Proc Chapter 12
30~3F		HD-bypass Chapter 03			Read FIFO, Write FIFO Chapter 07	
40~4F	Miscellaneous Chapter 04					
50~5F						
60~6F		Mode Detect Chapter 10		PIP Chapter 08		
70~7F						
80~8F						
90~9F	OSD Chapter 09					
A0~AF						
B0~BF						
C0~CF						
D0~DF						
E0~EF						

Note:

1. Address marked with ▒ is not existed in 5725.
2. All registers (except **chapter 01** status register is read only) have default value “0x00” after power up.
3. All registers require segment for access. Segment is defined in address F0.
   For example:
   S1_46 means F0 must be set to 1 before accessing 46
   S1_46=8D equal following operation:
   F0 = 01
   46 = 8D

Table of Contents

Chapter 00. STATUS REGISTERS

INPUT MODE STATUS 00 REG S0_00, RO

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

Bit	Name	Function
0	IF_STATUS_[0]	**Vertical stable indicator** When =1, means input vertical timing is stable
1	IF_STATUS_[1]	**Horizontal stable indicator** When =1, means input horizontal timing is table
2	IF_STATUS_[2]	**H & V stable indicator** When =1, means input H/V timing are both stable
3	IF_STATUS_[3]	**NTSC interlace indicator** When =1, means input is NTSC interlace (480i) source
4	IF_STATUS_[4]	**NTSC progressive indicator** When =1, means input is NTSC progressive (480P) source
5	IF_STATUS_[5]	**PAL interlace indicator** When =1, means input is PAL interlace (576i) source
6	IF_STATUS_[6]	**PAL progressive indicator** When =1, means input is PAL progressive (576P) source
7	IF_STATUS_[7]	**SD mode indicator** When =1, means input is SD mode (480i, 480P, 576i, 576P)

INPUT MODE STATUS 01 REG S0_01, RO

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

Bit	Name	Function
0	IF_STATUS_[8]	**VGA 60Hz mode** When =1, means input is VGA (640x480) 60Hz mode
1	IF_STATUS_[9]	**VGA 75Hz mode** When =1, means input is VGA (640x480) 75Hz mode
2	IF_STATUS_[10]	**VGA 85 Hz mode** When =1, means input is VGA (640x480) 85Hz mode
3	IF_STATUS_[11]	**VGA mode indicator** When =1, means input is VGA (640x480) source, include 60Hz/75Hz/85Hz
4	IF_STATUS_[12]	**SVGA 60Hz mode** When =1, means input is SVGA (800x600) 60Hz mode
5	IF_STATUS_[13]	**SVGA 75Hz mode** When =1, means input is SVGA (800x600) 75Hz mode
6	IF_STATUS_[14]	**SVGA 85Hz mode** When =1, means input is SVGA (800x600) 85Hz mode
7	IF_STATUS_[15]	**SVGA mode indicator** When =1, means input is SVGA (800x600) source, include 60Hz/75Hz/85Hz

INPUT MODE STATUS 02 REG S0_02, RO

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

Bit	Name	Function
0	IF_STATUS_[16]	**XGA 60Hz mode** When =1, means input is XGA (1024x768) 60Hz mode
1	IF_STATUS_[17]	**XGA 70Hz mode** When =1, means input is XGA (1024x768) 70Hz mode
2	IF_STATUS_[18]	**XGA 75Hz mode** When =1, means input is XGA (1024x768) 75Hz mode
3	IF_STATUS_[19]	**XGA 85Hz mode** When =1, means input is XGA (1024x768) 85Hz mode
4	IF_STATUS_[20]	**XGA mode indicator** When =1, means input is XGA (1024x768) source, include 60/70/75/85Hz
5	IF_STATUS_[21]	**SXGA 60Hz mode** When =1, means input is SXGA (1280x1024) 60Hz mode
6	IF_STATUS_[22]	**SXGA 75Hz mode** When =1, means input is SXGA (1280x1024) 75Hz mode
7	IF_STATUS_[23]	**SXGA 85Hz mode** When =1, means input is SXGA (1280x1024) 85Hz mode

INPUT MODE STATUS 03 REG S0_03, RO

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

Bit	Name	Function
0	IF_STATUS_[24]	**SXGA mode indicator** When =1, means input is SXGA (1280x1024) mode, include 60/75/85Hz
1	IF_STATUS_[25]	**Graphic mode indicator** When =1, means input is graphic mode input, include VGA/SVGA/XGA/SXGA
2	IF_STATUS_[26]	**HD720P 50Hz mode** When =1, means input is HD720P (1280x720) 50Hz mode
3	IF_STATUS_[27]	**HD720P 60Hz mode** When =1, means input is HD720P (1280x720) 60Hz mode
4	IF_STATUS_[28]	**HD720P mode indicator** When =1, means input is HD720P source, include 50Hz/60Hz
5	IF_STATUS_[29]	**HD2200_1125 interlace** When =1, means input is 2200x1125i mode
6	IF_STATUS_[30]	**HD2376_1250 interlace** When =1, means input is 2376x1250i mode
7	IF_STATUS_[31]	**HD2640_1125 interlace** When =1, means input is 2640x1125i mode

INPUT MODE STATUS 04 REG S0_04, RO

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

Bit	Name	Function
0	IF_STATUS_[32]	**HD1808i indicator** When =1, means input is HD1080i source, include 2200x1125i, 2376x1250i, 2640x1125i modes
1	IF_STATUS_[33]	**HD2200_1125P** When =1, means input is HD 2200x1125P mode
2	IF_STATUS_[34]	**HD2376_1250P** When =1, means input is HD 2376x1250P mode
3	IF_STATUS_[35]	**HD2640_1125P** When =1, means input is HD 2640x1125P mode
4	IF_STATUS_[36]	**HD 1080P indicator** When =1, means input is 1080P source, include 2200x1250P, 2376x1125P
5	IF_STATUS_[37]	**HD mode indicator** When =1, means input is HD source, include 720P, 1080i, 1080P
6	IF_STATUS_[38]	**Interlace video indicator** When =1, means input is interlace video source, include 480i, 576i, 1080i
7	IF_STATUS_[39]	**Progressive video indicator** When =1, means input is progressive video source, include 480P, 576P, 720P, 1080P modes

INPUT MODE STATUS 05 REG S0_05, RO

Bit	7	6	5	4	3	2	1	0
	RESERVED			IF_STATUS_[44:40]				

Bit	Name	Function
0	IF_STATUS_[40]	**User define mode** When =1, means input is the mode which match user define resolution
1	IF_STATUS_[41]	**No sync indicator** When =1, means input is not sync timing
2	IF_STATUS_[42]	**Horizontal unstable indicator** When =1, means input H sync is not stable
3	IF_STATUS_[43]	**Vertical unstable indicator** When =1, means input V sync is not stable
4	IF_STATUS_[44]	**Mode switch indicator** When =1, means input source switch the mode
7-5	RESERVED	**Reserved**

INPUT SIZE STATUS 00 REG S0_06, RO

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

Bit	Name	Function
7-0	IF_HPERIOD_[7:0]	**Input source H total measurement result** The value = input source H total pixels / 4

INPUT SIZE STATUS 01 REG S0_07, RO

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

Bit	Name	Function
0	IF_HPERIOD_[8]	**Input source H total measurement result** The value = input source H total pixels / 4
7-1	IF_VPERIOD_[6:0]	**Input source V total measurement result** The value = input source V total lines

INPUT SIZE STATUS 02 REG S0_08, RO

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

Bit	Name	Function
3-0	IF_VPERIOD_[10:7]	**Input source V total measurement result** The value = input source V total lines
7-4	RESERVED	**Reserved**

MISC STATUS 00 REG S0_09, RO

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

Bit	Name	Function
5-0	MISC_STATUS_[5:0]	**Reserved**
6	MISC_STATUS_[6]	**LOCK indicator from PLL648**
7	MISC_STATUS_[7]	**LOCK indicator from PLLAD**

MISC STATUS 01 REG S0_0A, RO

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

Bit	Name	Function
0	MISC_STATUS_[8]	**PIP enable signal in Vertical** When =1, means sub picture's vertical period in PIP mode
1	MISC_STATUS_[8]	**PIP enable signal in Horizontal** When =1, means sub picture's horizontal period in PIP mode
2	MISC_STATUS_[8]	**Reserved**
3	MISC_STATUS_[8]	**Reserved**
4	MISC_STATUS_[8]	**Display output Vertical Blank** When =1, means in display vertical blanking
5	MISC_STATUS_[8]	**Display output Horizontal Blank** When =1, means in display horizontal blanking
6	MISC_STATUS_[8]	**Display output Vertical Sync** When =1, means in display vertical sync (the output sync is high active)
7	MISC_STATUS_[8]	**Display output Horizontal Sync** When =1, means in display horizontal sync (the output sync is high active)

CHIP ID 00 REG S0_0B, RO

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

Bit	Name	Function
7-0	CHIP_ID_[7:0]	Foundry ID

CHIP ID 01 REG S0_0C, RO

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

Bit	Name	Function
7-0	CHIP_ID_[15:8]	Product ID

CHIP ID 02 REG S0_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 reversion ID

GPIO STATUS 00 REG S0_0E, RO

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

Bit	Name	Function
0	GPIO_STATUS_[0]	**GPIO bit0 status** GPIO bit0 (GPIO pin76) status
1	GPIO_STATUS_[1]	**GPIO bit1 status** GPIO bit1 (HALF pin77) status
2	GPIO_STATUS_[2]	**GPIO bit2 status** GPIO bit2 (SCLSA pin43) status
3	GPIO_STATUS_[3]	**GPIO bit3 status** GPIO bit3 (MBA pin107) status
4	GPIO_STATUS_[4]	**GPIO bit4 status** GPIO bit4 (MCS1 pin109) status
5	GPIO_STATUS_[5]	**GPIO bit5 status** GPIO bit5 (HBOUT pin6) status
6	GPIO_STATUS_[6]	**GPIO bit6 status** GPIO bit6 (VBOUT pin7) status
7	GPIO_STATUS_[7]	**GPIO bit7 status** GPIO bit7 (CLKOUT pin4) status

INTERRUPT STATUS 00 REG S0_0F, RO

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

Bit	Name	Function
0	INT_STATUS_[0]	**Interrupt status bit0, SOG unstable** When =1, means input SOG source is unstable
1	INT_STATUS_[1]	**Interrupt status bit1, SOG switch** When =1, means input SOG source switch the mode
2	INT_STATUS_[2]	**Interrupt status bit2, SOG stable** When =1, means input SOG source is stable
3	INT_STATUS_[3]	**Interrupt status bit3, mode switch** When =1, means input source switch the mode
4	INT_STATUS_[4]	**Interrupt status bit4, no sync** When =1, means input source is not H-sync input.
5	INT_STATUS_[5]	**Interrupt status bit5, H-sync status** When =1, means input H-sync status is changed between stable and unstable
6	INT_STATUS_[6]	**Interrupt status bi6, V-sync status** When =1, means input V-sync status is changed between stable and unstable
7	INT_STATUS_[7]	**Interrupt status bit7, H-sync status** When =1, means input H-sync status is changed between stable and unstable

VIDEO_PROC STATUS 00 REG S0_10, RO

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

Bit	Name	Function
3-0	VDS_STATUS_[3:0]	**Frame number**
4	VDS_STATUS_[4]	**Output Vertical Sync**
5	VDS_STATUS_[5]	**Output Horizontal Sync**
7-6	VDS_STATUS_[7:6]	**Reserved**

VIDEO_PROC STATUS 01 REG S0_11, RO

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

Bit	Name	Function
0	VDS_STATUS_[8]	**Field Index** When =0, in display top field When =1, in display bottom field
1	VDS_STATUS_[9]	**Composite Blanking** When =0, in display active period When =1, in display blanking period
3-2	VDS_STATUS_[11:10]	**Reserved**
7-4	VDS_STATUS_[15:12]	**Vertical counter bit [3:0]** Vertical counter value, indicate the line number in display

VIDEO_PROC STATUS 02 REG S0_12, RO

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

Bit	Name	Function
6-0	VDS_STATUS_[22:16]	**Vertical counter bit [10:4]** Vertical counter value, indicate the line number in display
7	VDS_STATUS_[23]	**Reserved**

MEM_FF STATUS 00 REG S0_13, RO

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

Bit	Name	Function
0	MEM_FF_STATUS_[0]	**WFF FIFO full indicator** When =1, means WFF FIFO is full
1	MEM_FF_STATUS_[1]	**WFF FIFO empty indicator** When =1, means WFF FIFO is empty
2	MEM_FF_STATUS_[2]	**RFF FIFO full indicator** When =1, means RFF FIFO is full
3	MEM_FF_STATUS_[3]	**RFF FIFO empty indicator** When =1, means RFF FIFO is empty
4	MEM_FF_STATUS_[4]	**Capture FIFO full indicator** When =1, means capture FIFO is full
5	MEM_FF_STATUS_[5]	**Capture FIFO empty indicator** When =1, means capture FIFO is empty
6	MEM_FF_STATUS_[6]	**Playback FIFO full indicator** When =1, means playback FIFO is full
7	MEM_FF_STATUS_[7]	**Playback FIFO empty indicator** When =1, means playback FIFO is empty

MEM_FF STATUS 00 REG S0_14, RO

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

Bit	Name	Function
0	MEM_FF_STATUS_[8]	**Memory control initial indicator** When =1, means external memory chip initial is finished
7-1	MEM_FF_STATUS_[15:9]	**Reserved**

DEINT STATUS 00 REG S0_15, RO

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

Bit	Name	Function
6-0	DEINT_STATUS_[6:0]	**Reserved**
7	DEINT_STATUS_[7]	**3:2 pull-down indicator** When =1, means de-interlace is in 3:2 pull-down mode

SYNC PROC STATUS 00 REG S0_16, RO

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

Bit	Name	Function
0	SYNC_PROC_STATUS_[0]	**HS polarity** When =0, means input H-sync is low active When =1, means input H-sync is high active
1	SYNC_PROC_STATUS_[1]	**HS active**
2	SYNC_PROC_STATUS_[2]	**VS polarity** When =0, means input V-sync is low active When =1, means input V-sync is high active
3	SYNC_PROC_STATUS_[3]	**VS active**
7-4	SYNC_PROC_STATUS_[7:4]	**Reserved**

SYNC PROC STATUS 01 REG S0_17, RO

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

Bit	Name	Function
7-0	SYNC_PROC_STATUS_[15:8]	**H total value** Input source H-total value

SYNC PROC STATUS 02 REG S0_18, RO

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

Bit	Name	Function
3-0	SYNC_PROC_STATUS_[19:16]	**H total value** Input source H-total value
7-4	SYNC_PROC_STATUS_[23:20]	**Reserved**

SYNC PROC STATUS 03 REG S0_19, RO

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

Bit	Name	Function
7-0	SYNC_PROC_STATUS_[31:24]	**H low pulse length value** Input H-sync low active pulse length (for H-sync polarity detection)

SYNC PROC STATUS 04 REG S0_1A, RO

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

Bit	Name	Function
3-0	SYNC_PROC_STATUS_[35:32]	**H low pulse length value** Input H-sync low active pulse length (for H-sync polarity detection)
7-4	SYNC_PROC_STATUS_[39:36]	**Reserved**

SYNC PROC STATUS 05 REG S0_1B, RO

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

Bit	Name	Function
7-0	SYNC_PROC_STATUS_[47:40]	**V total value** Input source V-total lines value

SYNC PROC STATUS 06 REG S0_1C, RO

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

Bit	Name	Function
2-0	SYNC_PROC_STATUS_[50:48]	**V total value** Input source V-total lines value
7-3	SYNC_PROC_STATUS_[55:51]	**Reserved**

SYNC PROC STATUS 07 REG S0_1D, RO

Bit	7	6	5	4	3	2	1	0
	RESERVED							

Bit	Name	Function
7-0	RESERVED	Reserved

SYNC PROC STATUS 08 REG S0_1E, RO

Bit	7	6	5	4	3	2	1	0
	RESERVED							

Bit	Name	Function
7-0	RESERVED	Reserved

TEST BUS STATUS 00 REG S0_1F, RO

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

Bit	Name	Function
7-0	TEST_BUS_[23:16]	Reserved

TEST FIFO STATUS 00 REG S0_20, RO

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

Bit	Name	Function
7-0	TEST_FF_STATUS_[7:0]	Reserved

TEST FIFO STATUS 01 REG S0_21, RO

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

Bit	Name	Function
7-0	TEST_FF_STATUS_[15:8]	Reserved

CRC RFF STATUS 00 REG S0_22, RO

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

Bit	Name	Function
7-0	CRC_REGOUT_RFF_[7:0]	Reserved

CRC PB STATUS 00 REG S0_23, RO

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

Bit	Name	Function
7-0	CRC_REGOUT_PB_[7:0]	Reserved

CRC RESULT STATUS 00 REG S0_24, RO

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

Bit	Name	Function
7-0	CRC_STATUS_[7:0]	Reserved

CRC RESULT STATUS 01 REG S0_25, RO

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

Bit	Name	Function
7-0	CRC_STATUS_[15:8]	Reserved

CRC RESULT STATUS 02 REG S0_26, RO

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

Bit	Name	Function
7-0	CRC_STATUS_[23:16]	Reserved

CRC RESULT STATUS 03 REG S0_27, RO

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

Bit	Name	Function
7-0	CRC_STATUS_[31:24]	Reserved

CRC RESULT STATUS 04 REG S0_28, RO

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

Bit	Name	Function
7-0	CRC_STATUS_[39:32]	Reserved

CRC RESULT STATUS 05 REG S0_29, RO

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

Bit	Name	Function
7-0	CRC_STATUS_[47:40]	Reserved

CRC RESULT STATUS 06 REG S0_2A, RO

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

Bit	Name	Function
7-0	CRC_STATUS_[55:48]	Reserved

CRC RESULT STATUS 07 REG S0_2B, RO

Bit	7	6	5	4	3	2	1	0
	CRC_STATUS_[63:56]							

Bit	Name	Function
7-0	CRC_STATUS_[63:56]	Reserved

CRC RESULT STATUS 08 REG S0_2C, RO

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

Bit	Name	Function
7-0	CRC_STATUS_[71:64]	Reserved

CRC RESULT STATUS 09 REG S0_2D, RO

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

Bit	Name	Function
7-0	CRC_STATUS_[79:72]	Reserved

TEST BUS STATUS 01 REG S0_2E, RO

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

Bit	Name	Function
7-0	TEST_BUS_[7:0]	Reserved

TEST BUS STATUS 02 REG S0_2F, RO

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

Bit	Name	Function
7-0	TEST_BUS_[15:8]	Reserved

Chapter 01. INPUT FORMATTER REGISTERS

INPUT_FORMATTER 00 REG S1_00, R/W

Bit	7	6	5	4	3	2	1	0
	IF_HS_FLIP	IF_PRGRSV_CNTRL	IF_VS_SEL	IF_SEL16BIT	IF_SEL_656	IF_UV_REVERT	IF_MATRIX_BYPS	IF_IN_DREG_BYPS

Bit	Name	Function
0	IF_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	IF_MATRIX_BYPS	**Rgb2yuv matrix bypass** If source is yuv24bit, bypass the rgb2yuv matrix. 0:source is 24bit RGB. Do rgb2yuv. 1: data bypass.
2	IF_UV_REVERT	**8bit to 16bit convert Y/UV flip control** If input is 8bit data, when it convert to 16bit, this bit control Y and UV order: 0: Keep the designed order 1: Flip the Y and UV order
3	IF_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.
4	IF_SEL16BIT	**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. {table: | | Sel_16bit | Sel_24bit | | 8bit 656/601 input | 0 | 0 | | 16bit 601 input | 1 | 0 | | 24bit yuv/rgb 601 input | * | 1 |}
5	IF_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.
6	IF_PRGRSV_CNTRL	**Select progressive data** Progressive mode. Choose the progressive data. 0: source is interlaced. 1: source is progressive.
7	IF_HS_FLIP	**Horizontal sync flip control** Control the horizontal sync output from CCIR process 0: keep the original horizontal sync. 1: flip horizontal sync.

INPUT_FORMATTER 01 REG S1_01, R/W

Bit	7	6	5	4	3	2	1	0
	IF_SEL24BIT	IF_Y_DELAY		IF_TAP6_BYPS	IF_V_DELAY	IF_U_DELAY	IF_UV_FLIP	IF_VS_FLIP

Bit	Name	Function
0	IF_VS_FLIP	**Vertical sync flip control** Control the vertical sync output from CCIR process 0: keep original vertical sync. 1: flip vertical sync.
1	IF_UV_FLIP	**YUV 422to444 UV flip control** Control the U and V order in yuv422to444 conversion. 0: keep original U and V order. 1: exchange the U and V order.
2	IF_U_DELAY	**U data select in YUV 422to444 conversion** Select original U data or 1-clock delayed U data, so that it can align with V data. 0: select original U data after dmux. 1: select 1-clock delayed U data after dmux.
3	IF_V_DELAY	**V data select in YUV 422to444 conversion** Select original V data or 1-clock delayed V data, so that it can align with U data. 0: select original V data after dmux. 1: select 1-clock delayed V data after dmux.
4	IF_TAP6_BYPS	**Tap6 interpolator bypass control in YUV 422to444 conversion** Select the data if pass the tap6 interpolator or not. 0: the data will pass the tap6 interpolator. 1: the data will not pass the tap6 interpolator
6-5	IF_Y_DELAY	**Y data pipes control in YUV422to444 conversion** Control the Y data pipe delay, so that it can align with U and V. IF_Y_DELAY / Y data delay pipes 00 / 1 01 / 2 10 / 3 11 / 4
7	IF_SEL24BIT	**Select 24bit data** If input source is 24bit data, choose the 24bit data path.

INPUT_FORMATTER 02 REG S1_02, R/W

Bit	7	6	5	4	3	2	1	0
	IF_HS_UV_SIGN2UNSIGN	IF_HS_Y_PDELAY		IF_HS_TAP11_BYPS	IF_HS_PSHIFT_BYPS	IF_HS_INT_LPF_BYPS	IF_HS_SEL_LPF	IF_SEL_WEN

Bit	Name	Function
0	IF_SEL_WEN	**Select the write enable for line double** If the input is HD source, this bit will be set to 1. 0: if the source is SD data. 1: if the source is HD data.
1	IF_HS_SEL_LPF	**Low pass filter or interpolator selection** The low pass filter and interpolator data path is combined together. 0: select interpolator data path. 1: select low pass filter data path.
2	IF_HS_INT_LPF_BYPS	**Combined INT and LPF data path bypass control** If the data can't do horizontal scaling-down, bypass the INT/LPF data path. 0: select the INT/LPF data path. 1: bypass the INT/LPF data path
3	IF_HS_PSHIFT_BYPS	**Phase adjustment bypass control** If the data can't do phase adjustment, this bit should be set to 1. 0: select phase adjustment data path. 1: bypass phase adjustment.
4	IF_HS_TAP11_BYPS	**Tap11 LPF bypass control in YUV444to422 conversion** Select the data if pass the tap11 LPF or not. 0: the data will pass the tap11 low pass filter. 1: the data will not pass the tap11 low pass filter
6-5	IF_HS_Y_PDELAY	**Y data pipes control in YUV444to422 conversion** Control the Y data pipe delay, so that it can align with UV. IF_HS_Y_DELAY / Y data delay pipes 00 / 1 01 / 2 10 / 3 11 / 4
7	IF_HS_UV_SIGN2UNSIGN	**UV data select** If UV is signed, select the unsigned UV data 0: select the original UV 1: select the UV after sign processing

INPUT_FORMATTER 03 REG S1_03, R/W

Bit	7	6	5	4	3	2	1	0
	IF_HS_RATE_SEG0							

Bit	Name	Function
7-0	IF_HS_RATE_SEG0	**Horizontal non-linear scaling-down 1st segment DDA increment [11:4] (total 12 bits)** The entire segment share the lowest 4bit, that is to say, the whole scale ration is hscale = {hscale0, hscale_low}. Assume the scaling ratio is n/m, then the value should be 4095x(m-n)/n

INPUT_FORMATTER 04 REG S1_04, R/W

Bit	7	6	5	4	3	2	1	0
	IF_HS_RATE_SEG1							

Bit	Name	Function
7-0	IF_HS_RATE_SEG1	**Horizontal non-linear scaling-down 2nd segment DDA increment [11:4] (total 12 bits)**

INPUT_FORMATTER 05 REG S1_05, R/W

Bit	7	6	5	4	3	2	1	0
	IF_HS_RATE_SEG2							

Bit	Name	Function
7-0	IF_HS_RATE_SEG2	**Horizontal non-linear scaling-down 3rd segment DDA increment [11:4] (total 12 bits)**

INPUT_FORMATTER 06 REG S1_06, R/W

Bit	7	6	5	4	3	2	1	0
	IF_HS_RATE_SEG3							

Bit	Name	Function
7-0	IF_HS_RATE_SEG3	**Horizontal non-linear scaling-down 4th segment DDA increment [11:4] (total 12 bits)**

INPUT_FORMATTER 07 REG S1_07, R/W

Bit	7	6	5	4	3	2	1	0
	IF_HS_RATE_SEG4							

Bit	Name	Function
7-0	IF_HS_RATE_SEG4	Horizontal non-linear scaling-down 5th segment DDA increment [11:4] (total 12 bits)

INPUT_FORMATTER 08 REG S1_08, R/W

Bit	7	6	5	4	3	2	1	0
	IF_HS_RATE_SEG5							

Bit	Name	Function
7-0	IF_HS_RATE_SEG5	Horizontal non-linear scaling-down 6th segment DDA increment [11:4] (total 12 bits)

INPUT_FORMATTER 09 REG S1_09, R/W

Bit	7	6	5	4	3	2	1	0
	IF_HS_RATE_SEG6							

Bit	Name	Function
7-0	IF_HS_RATE_SEG6	Horizontal non-linear scaling-down 7th segment DDA increment [11:4] (total 12 bits)

INPUT_FORMATTER 0A REG S1_0A, R/W

Bit	7	6	5	4	3	2	1	0
	IF_HS_RATE_SEG7							

Bit	Name	Function
7-0	IF_HS_RATE_SEG7	Horizontal non-linear scaling-down 8th segment DDA increment [11:4] (total 12 bits)

INPUT_FORMATTER 0B REG S1_0B, R/W

Bit	7	6	5	4	3	2	1	0
	IF_LD_SEL_PROV	IF_SEL_HSCALE	IF_HS_DEC_FACTOR		IF_HS_RATE_LOW			

Bit	Name	Function
3-0	IF_HS_RATE_LOW	**Horizontal non-linear scaling-down DDA increment shared lowest 4 bits [3:0] (total 12 bits)**
5-4	IF_HS_DEC_FACTOR	**Horizontal non-linear scaling-down factor select** If the scaling ratio is less than ½, use it and DDA to generate the we and phase 00: scaling-ratio is more than ½. 01: scaling-ratio is less than ½. 10: scaling-ratio is less than ¼.
6	IF_SEL_HSCALE	**Select the data path after horizontal scaling-down** If the data have do scaling-down, this bit should be open. 0: select the data and write enable from CCIR to line double. 1: select the scaling-down data and write enable to line double.
7	IF_LD_SEL_PROV	**Line double read reset select** If source is progressive data, choose the related progressive timing as read reset timing. 0: select read reset timing of interlace data. 1: select read reset timing of progressive data

INPUT_FORMATTER 0C REG S1_0C, R/W

Bit	7	6	5	4	3	2	1	0
	IF_INI_ST[2:0]			IF_LD_ST				IF_LD_RAM_BYPS

Bit	Name	Function
0	IF_LD_RAM_BYPS	**Line double bypass control** If the interlace data can't do line double, if the progressive data can't do scaling-down, line double FIFO should be bypass. 0: select interlace line double data from FIFO. 1: bypass line double FIFO.
4-1	IF_LD_ST	**Line double write reset generation start position** If the internal counter equals the defined value the write reset will be high pulse.
7-5	IF_INI_ST[2:0]	**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 0D REG S1_0D, R/W

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

Bit	Name	Function
7-0	IF_INI_ST [10:3]	**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 0E REG S1_0E, R/W

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

Bit	Name	Function
7-0	IF_HSYNC_RST [7:0]	**Total pixel number per line** Use to generate progressive timing if input is interlace data [7:0]

INPUT_FORMATTER 0F REG S1_0F, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED					IF_HSYNC_RST [10:8]		

Bit	Name	Function
2-0	IF_HSYNC_RST [10:8]	**Total pixel number per line** Use to generate progressive timing if input is interlace data [10:8]
7-3	RESERVED	

INPUT_FORMATTER 10 REG S1_10, R/W

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

Bit	Name	Function
7-0	IF_HB_ST [7:0]	**Horizontal blanking start position (set 0)** Horizontal blanking (set 0) start position [7:0].

INPUT_FORMATTER 11 REG S1_11, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED					IF_HB_ST [10:8]		

Bit	Name	Function
2-0	IF_HB_ST [10:8]	**Horizontal blanking start position (set 0)** Horizontal blanking (set 0) start position [10:8].
7-3	RESERVED	

INPUT_FORMATTER 12 REG S1_12, R/W

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

Bit	Name	Function
7-0	IF_HB_SP [7:0]	**Horizontal blanking stop position (set 0)** Horizontal blanking (set 0) stop position [7:0].

INPUT_FORMATTER 13 REG S1_13, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED					IF_HB_SP [10:8]		

Bit	Name	Function
2-0	IF_HB_SP [10:8]	**Horizontal blanking stop position (set 0)** Horizontal blanking (set 0) stop position [10:8].
7-3	RESERVED	

INPUT_FORMATTER 14 REG S1_14, R/W

Bit	7	6	5	4	3	2	1	0
	IF_HB_ST1 [7:0]							

Bit	Name	Function
7-0	IF_HB_ST1 [7:0]	**Horizontal blanking start position (set 1)** Horizontal blanking (set 1) start position [7:0].

INPUT_FORMATTER 15 REG S1_15, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED					IF_HB_ST1 [10:8]		

Bit	Name	Function
2-0	IF_HB_ST1 [10:8]	**Horizontal blanking start position (set 1)** Horizontal blanking (set 1) start position [10:8].
7-3	RESERVED	

INPUT_FORMATTER 16 REG S1_16, R/W

Bit	7	6	5	4	3	2	1	0
	IF_HB_SP1 [7:0]							

Bit	Name	Function
7-0	IF_HB_SP1 [7:0]	**Horizontal blanking stop position (set 1)** Horizontal blanking (set 1) stop position [7:0].

INPUT_FORMATTER 17 REG S1_17, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED					IF_HB_SP1 [10:8]		

Bit	Name	Function
2-0	IF_HB_SP1 [10:8]	**Horizontal blanking stop position (set 1)** Horizontal blanking (set 1) stop position [10:8].
7-3	RESERVED	

INPUT_FORMATTER 18 REG S1_18, R/W

Bit	7	6	5	4	3	2	1	0
	IF_HB_ST2 [7:0]							

Bit	Name	Function
7-0	IF_HB_ST2 [7:0]	**Horizontal blanking start position (set 2)** Horizontal blanking (set 2) start position [7:0].

INPUT_FORMATTER 19 REG S1_19, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED					IF_HB_ST2 [10:8]		

Bit	Name	Function
2-0	IF_HB_ST2 [10:8]	**Horizontal blanking start position (set 2)** Horizontal blanking (set 2) start position [10:8].
7-3	RESERVED	

INPUT_FORMATTER 1A REG S1_1A, R/W

Bit	7	6	5	4	3	2	1	0
	IF_HB_SP2 [7:0]							

Bit	Name	Function
7-0	IF_HB_SP2 [7:0]	**Horizontal blanking stop position (set 2)** Horizontal blanking (set 2) stop position [7:0].

INPUT_FORMATTER 1B REG S1_1B, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED					IF_HB_SP2 [10:8]		

Bit	Name	Function
2-0	IF_HB_SP2 [10:8]	**Horizontal blanking stop position (set 2)** Horizontal blanking (set 2) stop position [10:8].
7-3	RESERVED	

INPUT_FORMATTER 1C REG S1_1C, R/W

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

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

INPUT_FORMATTER 1D REG S1_1D, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED					IF_VB_ST [10:8]		

Bit	Name	Function
2-0	IF_VB_ST [10:8]	**Vertical blanking start position** Vertical blanking start position [10:8].
7-3	RESERVED	

INPUT_FORMATTER 1E REG S1_1E, R/W

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

Bit	Name	Function
7-0	IF_VB_SP [7:0]	**Vertical blanking stop position** Vertical blanking stop position [7:0].

INPUT_FORMATTER 1F REG S1_1F, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED					IF_VB_SP [10:8]		

Bit	Name	Function
2-0	IF_VB_SP [10:8]	**Vertical blanking stop position** Vertical blanking stop position [10:8].
7-3	RESERVED	

INPUT_FORMATTER 20 REG S1_20, R/W

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

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

INPUT_FORMATTER 21 REG S1_21, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED				IF_LINE_ST [11:8]			

Bit	Name	Function
3-0	IF_LINE_ST [11:8]	**Line signal start position** Progressive line start position.
7-4	RESERVED	

INPUT_FORMATTER 22 REG S1_22, R/W

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

Bit	Name	Function
7-0	IF_LINE_SP [7:0]	**Line signal stop position** Progressive line stop position.

INPUT_FORMATTER 23 REG S1_23, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED				IF_LINE_SP [11:8]			

Bit	Name	Function
3-0	IF_LINE_SP [11:8]	**Line signal stop position** Progressive line stop position.
7-4	RESERVED	

INPUT_FORMATTER 24 REG S1_24, R/W

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

Bit	Name	Function
7-0	IF_HBIN_ST [7:0]	**Horizontal blank for scale down start position** Horizontal blank for scale down line reset start position

INPUT_FORMATTER 25 REG S1_25, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED				IF_HBIN_ST [11:8]			

Bit	Name	Function
3-0	IF_HBIN_ST [11:8]	**Horizontal blank for scale down start position** Horizontal blank for scale down line reset start position
7-4	RESERVED	

INPUT_FORMATTER 26 REG S1_26, R/W

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

Bit	Name	Function
7-0	IF_HBIN_SP [7:0]	**Horizontal blank for scale down stop position** Horizontal blank for scale down line reset stop position

INPUT_FORMATTER 27 REG S1_27, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED				IF_HBIN_SP [11:8]			

Bit	Name	Function
3-0	IF_HBIN_SP [11:8]	**Horizontal blank for scale down stop position** Horizontal blank for scale down line reset stop position
7-4	RESERVED	

INPUT_FORMATTER 28 REG S1_28, R/W

Bit	7	6	5	4	3	2	1	0
	IF_TEST_SEL				IF_TEST_EN	IF_SEL_ADC_SYNC	IF_LD_WRST_SEL	RESERVED

Bit	Name	Function
0	RESERVED	
1	IF_LD_WRST_SEL	**Line double write reset select** Select hbin/line write reset 0: select line generated write reset 1: select hbin generated write reset
2	IF_SEL_ADC_SYNC	**ADC sync select** Select ADC sync to data path
3	IF_TEST_EN	**IF test bus control enable** Enable test signal.
7-4	IF_TEST_SEL	**Test signals select bits.** Select which signal to the test bus.

INPUT_FORMATTER 29 REG S1_29, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED						IF_AUTO_OFST_PRD	IF_AUTO_OFST_EN

Bit	Name	Function
0	IF_AUTO_OFST_EN	**Auto offset adjustment enable** 1: enable 0: disable
1	IF_AUTO_OFST_PRD	**Auto offset adjustment period control** 1: by frame 0: by line
7-2	RESERVED	

INPUT_FORMATTER 2A REG S1_2A, R/W

Bit	7	6	5	4	3	2	1	0
	IF_AUTO_OFST_V_RANGE				IF_AUTO_OFST_U_RANGE			

Bit	Name	Function
3-0	IF_AUTO_OFST_U_RANGE	U channel offset detection range
7-4	IF_AUTO_OFST_V_RANGE	V channel offset detection range

Chapter 02. DEINTERLACER REGISTERS

DEINTERLACER 00 REG S2_00, R/W

Bit	7	6	5	4	3	2	1	0
	DIAG_BOB_PLDY_RAM_BYPS	DIAG_BOB_MIN_CBYPS	DIAG_BOB_YTAP3_BYPS	DIAG_BOB_DET_BYPS		DIAG_BOB_WEAVE_BYPS	DIAG_BOB_COEF_SEL	DIAG_BOB_MIN_BYPS

Bit	Name	Function
0	DIAG_BOB_MIN_BYPS	**Diagonal Function Bypass Control** When set to 1, bypass diagonal min selection for Y. No diagonal detection, just vertically two pixels average.
1	DIAG_BOB_COEF_SEL	**Diagonal Bob Low pass Filter Coefficient Selection** Select coefficients for pixel difference low pass filter

DIAG_BOB_COEF_SEL	Internal Selected Coefficient
1	15/16
0	14/16

2	DIAG_BOB_WEAVE_BYPS	**Weave Function Bypass Control** When set to 1, weave function will bypass. Just repeat original data.
3	DIAG_BOB_DET_BYPS[0]	**Diagonal Bob Deinterlacer Angle Detect Bypass Control** When set to 1, bypass the detection of angle arctan (1/4).
4	DIAG_BOB_DET_BYPS[1]	**Diagonal Bob Deinterlacer Angle Detect Bypass Control** When set to 1, bypass the detection of angle arctan (1/6).
5	DIAG_BOB_YTAP3_BYPS	**Diagonal Bob Deinterlacer Y Tap3 Filter Bypass control** When set to 1, bypass the tap3 filter for Y.
6	DIAG_BOB_MIN_CBYPS	**Diagonal Bob Min Control For UV** When set to 1, bypass diagonal min select for UV. No diagonal detection, just vertically two pixels average.
7	DIAG_BOB_PLDY_RAM_BYPS	**Bypass Control For Pdelayer FIFO** When set to 1, bypass FIFO for pdelayer.

DEINTERLACER 01 REG S2_01, R/W

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

Bit	Name	Function
7-0	DIAG_BOB_PLDY_SP [7:0]	**The Distance Control of Pdelayer Reset [7:0]** In pdelayer, adjust the delay between read reset and write reset.

DEINTERLACER 02 REG S2_02, R/W

Bit	7	6	5	4	3	2	1	0
	MADPT_UV_VSCALE_BYPS	MADPT_Y_VSCALE_BYPS	MADPT_SEL_22	RESERVED				DIAG_BOB_PLDY_SP [8]

Bit	Name	Function
0	DIAG_BOB_PLDY_SP [8]	**The Distance Control of Pdelayer Reset [8]** In pdelayer, adjust the delay between read reset and write reset.
4-1	RESERVED	
5	MADPT_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.
6	MADPT_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
7	MADPT_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

DEINTERLACER 03 REG S2_03, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED			MADPT_NOISE_DET_RST		MADPT_NOISE_DET_SHIFT		MADPT_NOISE_DET_SEL

Bit	Name	Function
0	MADPT_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.
2-1	MADPT_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
4-3	MADPT_NOISE_DET_RST	**Noise detection time reset value** When set to 3, time counter reset at 1023. When set to 2, time counter reset at 511. When set to 1, time counter reset at 255. When set to 0, time counter reset at 127.
7-5	RESERVED	

DEINTERLACER 04 REG S2_04, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED	MADPT_NOISE_THRESHOLD_NOUT						

Bit	Name	Function
6-0	MADPT_NOISE_THRESHOLD_NOUT	**Auto noise detect threshold for NOUT** Threshold for NOUT signal.
7	RESERVED	

DEINTERLACER 05 REG S2_05, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED	MADPT_NOISE_THRESHOLD_VDS						

Bit	Name	Function
6-0	MADPT_NOISE_THRESHOLD_VDS	**Auto noise detect threshold for nout_vds_proc** Threshold for nout_vds_proc signal.
7	RESERVED	

DEINTERLACER 06 REG S2_06, R/W

Bit	7	6	5	4	3	2	1	0
	MADPT_GM_NOISE_VALUE							

Bit	Name	Function
3-0	MADPT_GM_NOISE_VALUE [3:0]	**Global noise low/global noise auto detect offset low** In global motion noise manual mode, global motion detection noise bit [3:0] In global motion noise auto-detect mode, global motion noise's offset bit [3:0]
7-4	MADPT_GM_NOISE_VALUE [7:4]	**Global noise high/global noise auto detect offset high** In global motion noise manual mode, global motion detection noise bit [7:4] In global motion noise auto-detect mode, global motion noise's offset bit [7:4]

DEINTERLACER 07 REG S2_07, R/W

Bit	7	6	5	4	3	2	1	0
	MADPT_STILL_NOISE_VALUE							

Bit	Name	Function
7-0	MADPT_STILL_NOISE_VALUE	**Global still control value** In manual mode, still-noise value bit. In auto-detect mode, still-noise's offset bit.

DEINTERLACER 08 REG S2_08, R/W

Bit	7	6	5	4	3	2	1	0
	MADPT_LESS_NOISE_VALUE							

Bit	Name	Function
7-0	MADPT_LESS_NOISE_VALUE	**Less-still noise value** User defined less still noise value.

DEINTERLACER 09 REG S2_09, R/W

Bit	7	6	5	4	3	2	1	0
	MADPT_STILL_NOISE_EST_GAIN				MADPT_NOISE_EST_GAIN			

Bit	Name	Function
3-0	MADPT_NOISE_EST_GAIN	**Global motion noise gain (in auto-detect mode)** Global motion noise gain in noise auto-detect mode
7-4	MADPT_STILL_NOISE_EST_GAIN	**Still-noise gain (in auto-detect mode)**

DEINTERLACER 0A REG S2_0A, R/W

Bit	7	6	5	4	3	2	1	0
	MADPT_Y_MI_DET_BYPS	RESERVED	MADPT_STILL_NOISE_EST_EN	MADPT_NOISE_EST_EN	RESERVED			

Bit	Name	Function
3-0	RESERVED	
4	MADPT_NOISE_EST_EN	**Global noise auto detection enable** When set to 1,global noise detection is in auto mode. When set to 0,global noise detection is in manual mode.
5	MADPT_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.
6	RESERVED	
7	MADPT_Y_MI_DET_BYPS	**Y motion index generation bypass** When set to 1, Y motion index generation is in manual mode

DEINTERLACER 0B REG S2_0B, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED	MADPT_Y_MI_OFFSET						

Bit	Name	Function
6-0	MADPT_Y_MI_OFFSET	**Y motion index offset** In auto mode, Y motion index's offset. In manual mode, Y motion index's user value.
7	RESERVED	

DEINTERLACER 0C

REG S2_0C, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED		MADPT_MI_1BIT_FRAME2_EN	MADPT_MI_1BIT_BYPS	MADPT_Y_MI_GAIN			

Bit	Name	Function
3-0	MADPT_Y_MI_GAIN	**Y motion index gain**
4	MADPT_MI_1BIT_BYPS	**Motion index feedback-bit bypass** When set to 1, motion index feedback-bit function will be bypass
5	MADPT_MI_1BIT_FRAME2_EN	**Enable Frame-two feedback-bit** When set to 1, enable frame-two feedback-bit.
7-6	RESERVED	

DEINTERLACER 0D

REG S2_0D, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED	MADPT_MI_THRESHOLD						

Bit	Name	Function
6-0	MADPT_MI_THRESHOLD	**Motion index feedback-bit generation's threshold bit**
7	RESERVED	

DEINTERLACER 0E

REG S2_0E, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED	MADPT_MI_THRESHOLD						

Bit	Name	Function
6-0	MADPT_MI_THRESHOLD	**Motion index fixed value**
7	RESERVED	

DEINTERLACER 0F REG S2_0F, R/W

Bit	7	6	5	4	3	2	1	0
	MADPT_STILL_LOCK				MADPT_STILL_UNLOCK		MADPT_STILL_ID	MADPT_STILL_DET_EN

Bit	Name	Function
0	MADPT_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.
1	MADPT_STILL_ID	**Still indicator defined by user (in manual mode only)** Still indicator defined by user, only useful in STILL_DET_EN =0.
3-2	MADPT_STILL_UNLOCK	**Still detection's auto unlock value** When unlock counter equals unlock value, "still" will go inactive.
7-4	MADPT_STILL_LOCK	**Still detection's auto lock value** When lock counter equals lock value, "still" will go active.

DEINTERLACER 10 REG S2_10, R/W

Bit	7	6	5	4	3	2	1	0
	MADPT_LESS_STILL_LOCK				MADPT_LESS_STILL_UNLOCK		MADPT_LESS_STILL_ID	MADPT_LESS_STILL_DET_EN

Bit	Name	Function
0	MADPT_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.
1	MADPT_LESS_STILL_ID	**Less still indicator defined by user (in manual mode only)** Less-still indicator defined by user, only useful in LESS_STILL_DET_EN =0.
3-2	MADPT_LESS_STILL_UNLOCK	**Less still detection's auto unlock value** When unlock counter equals unlock value, "less-still" will go inactive.
7-4	MADPT_LESS_STILL_LOCK	**Less still detection's auto lock value** When lock counter equals lock value, "less-still" will go active.

DEINTERLACER 11 REG S2_11, R/W

Bit	7	6	5	4	3	2	1	0
	MADPT_PULLDOWN32_OFFSET			MADPT_PULLDOWN32_ID	MADPT_EN_PULLDOWN32	RESERVED		

Bit	Name	Function
2-0	RESERVED	
3	MADPT_EN_PULLDOWN32	**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.
4	MADPT_PULLDOWN32_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
7-5	MADPT_PULLDOWN32_OFFSET	**3:2 pull-down sequence offset** 3:2 pull-down sequence offset

DEINTERLACER 12 REG S2_12, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED	MADPT_PULLDOWN32_LOCK_RST						

Bit	Name	Function
6-0	MADPT_PULLDOWN32_LOCK_RST	**3:2 pull-down auto lock value bit** When lock counter equals lock value, 3:2 pull-down is in active.
7	RESERVED	

DEINTERLACER 13 REG S2_13, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED	MADPT_PULLDOWN22_DET_CNTRL			RESERVED	MADPT_PULLDOWN22_OFFSET	MADPT_PULLDOWN22_ID	MADPT_EN_PULLDOWN22

Bit	Name	Function
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	MADPT_PULLDOWN22_OFFSET	**2:2 pull-down sequence offset** 2:2 pull-down sequence offset
3	RESERVED	
6-4	MADPT_PULLDOWN22_DET_CNTRL	**2:2 pull-down detection control bit** 2:2 pull-down accumulation result control
7	RESERVED	

DEINTERLACER 14 REG S2_14, R/W

Bit	7	6	5	4	3	2	1	0
	MADPT_PULLDOWN22_THRESHOLD [7:0]							

Bit	Name	Function
7-0	MADPT_PULLDOWN22_THRESHOLD [7:0]	**2:2 pull-down detection threshold bit [7:0]** 2:2 pull-down detection threshold bit [7:0]

DEINTERLACER 15 REG S2_15, R/W

Bit	7	6	5	4	3	2	1	0
	MADPT_PULLDOWN22_THRESHOLD [15:8]							

Bit	Name	Function
7-0	MADPT_PULLDOWN22_THRESHOLD [15:8]	**2:2 pull-down detection threshold bit [15:8]** 2:2 pull-down detection threshold bit [15:8]

DEINTERLACER 16 REG S2_16, R/W

Bit	7	6	5	4	3	2	1	0
	MAPDT_VT_SEL_PRGV	MADPT_VT_FILTER_CNTRL	MADPT_MO_ADP_UV_EN	MADPT_MO_ADP_Y_EN	RESERVED		MADPT_PULLDOWN22_THRESHOLD [17:16]	

Bit	Name	Function
1-0	MADPT_PULLDOWN22_THRESHOLD [17:16]	**2:2 pull-down detection threshold bit [17:16]** 2:2 pull-down detection threshold bit [17:16]
3-2	RESERVED	
4	MADPT_MO_ADP_Y_EN	**Enable pull-down in Y motion adaptive** When set to 1, enable pull-down for Y data motion adaptive.
5	MADPT_MO_ADP_UV_EN	**Enable pull-down in UV motion adaptive** When set to 1, enable pull-down for UV data motion adaptive.
6	MADPT_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.
7	MAPDT_VT_SEL_PRGV	**Select original data in progressive mode in VT filter** If the input is progressive mode or graphic mode, this bit must be set to 1.

DEINTERLACER 17 REG S2_17, R/W

Bit	7	6	5	4	3	2	1	0
	MADPT_UV_DELAY				MADPT_Y_DELAY			

Bit	Name	Function
3-0	MADPT_Y_DELAY	**Y delay pipe control**

MADPT_Y_DELAY	Y data delay pipes
0000	1
0001	2
0010	3
0011	4
0100	5
0101	6
0110	7
0111	8
1000	9
1001	10
1010	11
1011	12
1100	13
1101	14
1110	15
1111	16

Bit	Name	Function
7-4	MADPT_UV_DELAY	**UV delay pipe control**

MADPT_UV_DELAY	UV data delay pipes
0000	1
0001	2
0010	3
0011	4
0100	5
0101	6
0110	7
0111	8
1000	9
1001	10
1010	11
1011	12
1100	13
1101	14
1110	15
1111	16

DEINTERLACER 18 REG S2_18, R/W

Bit	7	6	5	4	3	2	1	0
	MADPT_HTAP_COEFF				MADPT_HTAP_BYPS	RESERVED	MADPT_DIVID_SEL	MADPT_DIVID_BYPS

Bit	Name	Function
0	MADPT_DIVID_BYPS	**Motion index divide bypass** When = 1, motion index no divide. When = 0, motion index will by divided by 2 or 4.
1	MADPT_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.
2	RESERVED	
3	MADPT_HTAP_BYPS	**Motion index horizontal filter bypass** When =1, motion index horizontal filter will be bypass
7-4	MADPT_HTAP_COEFF	**Motion index horizontal filter coefficient** Motion index horizontal filter coefficient.

DEINTERLACER 19 REG S2_19, R/W

Bit	7	6	5	4	3	2	1	0
	MADPT_VTAP2_COEFF				MADPT_VTAP2_ROUND_SEL	MADPT_VTAP2_BYPS	RESERVED	MADPT_BIT_STILL_EN

Bit	Name	Function
0	MADPT_BIT_STILL_EN	**Enable pixel base still** When set to 1, pixel base still function will enable.
1	RESERVED	
2	MADPT_VTAP2_BYPS	**Motion index vertical filter bypass** When = 1, motion index's vertical filter will be bypass.
3	MADPT_VTAP2_ROUND_SEL	**Motion index vertical filter round selection** When set to 1, the input data will be divided by 2.
7-4	MADPT_VTAP2_COEFF	**Motion index vertical filter coefficient**

DEINTERLACER 1A REG S2_1A, R/W

Bit	7	6	5	4	3	2	1	0
	MADPT_PIXEL_STILL_THRESHOLD_1							

Bit	Name	Function
7-0	MADPT_PIXEL_STILL_THRESHOLD_1	**Pixel base still threshold level one**

DEINTERLACER 1B REG S2_1B, R/W

Bit	7	6	5	4	3	2	1	0
	MADPT_PIXEL_STILL_THRESHOLD_2							

Bit	Name	Function
7-0	MADPT_PIXEL_STILL_THRESHOLD_2	Pixel base still threshold level two

DEINTERLACER 1C REG S2_1C, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED							

Bit	Name	Function
7-0	RESERVED	

DEINTERLACER 1D REG S2_1D, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED							

Bit	Name	Function
7-0	RESERVED	

DEINTERLACER 1E REG S2_1E, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED							

Bit	Name	Function
7-0	RESERVED	

DEINTERLACER 1F REG S2_1F, R/W

Bit	7	6	5	4	3	2	1	0
	MADPT_HFREQ_NOISE							

Bit	Name	Function
7-0	MADPT_HFREQ_NOISE	**High-frequency detection noise value** The noise value for high-frequency detection.

DEINTERLACER 20 REG S2_20, R/W

Bit	7	6	5	4	3	2	1	0
	MADPT_HFREQ_LOCK				RESERVED		MADPT_HFREQ_ID	MADPT_HFREQ_DET_EN

Bit	Name	Function
0	MADPT_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.
1	MADPT_HFREQ_ID	**High-frequency indicator by user (in manual mode)** High-frequency indicator by user, only useful in HFREQ_DET_EN =0
3-2	RESERVED	
7-4	MADPT_HFREQ_LOCK	**High-frequency auto lock value** When high-frequency lock counter equals lock value, high-frequency will be active

DEINTERLACER 21 REG S2_21, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED		MADPT_EN_NOUT_FOR_LESS_STILL	MADPT_EN_NOUT_FOR_STILL	RESERVED	MADPT_HFREQ_UNLOCK		

Bit	Name	Function
2-0	MADPT_HFREQ_UNLOCK	**High-frequency auto unlock value** When high-frequency unlock counter equals unlock value, high-frequency will be inactive
3	RESERVED	
4	MADPT_EN_NOUT_FOR_STILL	**Enable NOUT for still detection**
5	MADPT_EN_NOUT_FOR_LESS_STILL	**Enable NOUT for less-still detection**
7-6	RESERVED	

DEINTERLACER 22 REG S2_22, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED			MADPT_PD_SP				

Bit	Name	Function
4-0	MADPT_PD_SP	**Scaling down line buffer WRSTZ position adjustment bits** Adjust the position of write reset in vertical IIR filter line buffer, and phase adjustment line buffer.
7-5	RESERVED	

DEINTERLACER 23 REG S2_23, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED			MADPT_PD_ST				

Bit	Name	Function
4-0	MADPT_PD_ST	**Scaling down line buffer RRSTZ position adjustment** Adjust the position of read reset in vertical IIR filter line buffer, and phase adjustment line buffer.
7-5	RESERVED	

DEINTERLACER 24 REG S2_24, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED					MADPT_PD_RAM_BYPS	RESERVED	

Bit	Name	Function
1-0	RESERVED	
2	MADPT_PD_RAM_BYPS	**Bypass scaling down's line buffer** When set to 1, scaling down's line buffer will be bypass.
7-3	RESERVED	

DEINTERLACER 25 REG S2_25, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED							

Bit	Name	Function
7-0	RESERVED	

DEINTERLACER 26 REG S2_26, R/W

Bit	7	6	5	4	3	2	1	0
	MADPT_VIIR_ROUND_SEL	MADPT_VIIR_BYPS	RESERVED					

Bit	Name	Function
5-0	RESERVED	
6	MADPT_VIIR_BYPS	**Bypass V-IIR filter in vertical scaling down** When set to 1, V-IIR filter in vertical scaling down will be bypass.
7	MADPT_VIIR_ROUND_SEL	**V-IIR filter in vertical scaling down round selection** When set to 1, the input data will be divided by 2.

DEINTERLACER 27 REG S2_27, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED	MADPT_VIIR_COEF						

Bit	Name	Function
6-0	MADPT_VIIR_COEF	**V-IIR filter coefficient**
7	RESERVED	

DEINTERLACER 28 REG S2_28, R/W

Bit	7	6	5	4	3	2	1	0
	MADPT_VSCALE_RATE_LOW				RESERVED			

Bit	Name	Function
3-0	RESERVED	
7-4	MADPT_VSCALE_RATE_LOW	**Vertical non-linear scale down DDA increment shared low 4-bit** All the segment DDA increment share low 4bit

DEINTERLACER 29 REG S2_29, R/W

Bit	7	6	5	4	3	2	1	0
	MADPT_VSCALE_RATE_SEG0							

Bit	Name	Function
7-0	MADPT_VSCALE_RATE_SEG0	**Vertical non-linear scale down 1st segment DDA increment value** The actual DDA increment is vscale={vscale0, vscale_low}. Assume the scale factor is n/m, then vscale= 4095x(m-n)/n.

DEINTERLACER 2A REG S2_2A, R/W

Bit	7	6	5	4	3	2	1	0
	MADPT_VSCALE_RATE_SEG1							

Bit	Name	Function
7-0	MADPT_VSCALE_RATE_SEG1	**Vertical non-linear scale down 2nd segment DDA increment value** The actual DDA increment is vscale={vscale1, vscale_low}.

DEINTERLACER 2B REG S2_2B, R/W

Bit	7	6	5	4	3	2	1	0
	MADPT_VSCALE_RATE_SEG2							

Bit	Name	Function
7-0	MADPT_VSCALE_RATE_SEG2	**Vertical non-linear scale down 3rd segment DDA increment value** The actual DDA increment is vscale={vscale2, vscale_low}.

DEINTERLACER 2C REG S2_2C, R/W

Bit	7	6	5	4	3	2	1	0
	MADPT_VSCALE_RATE_SEG3							

Bit	Name	Function
7-0	MADPT_VSCALE_RATE_SEG3	**Vertical non-linear scale down 4th segment DDA increment value** The actual DDA increment is vscale={vscale3, vscale_low}.

DEINTERLACER 2D REG S2_2D, R/W

Bit	7	6	5	4	3	2	1	0
	MADPT_VSCALE_RATE_SEG4							

Bit	Name	Function
7-0	MADPT_VSCALE_RATE_SEG4	**Vertical non-linear scale down 5th segment DDA increment value** The actual DDA increment is vscale={vscale4, vscale_low}.

DEINTERLACER 2E REG S2_2E, R/W

Bit	7	6	5	4	3	2	1	0
	MADPT_VSCALE_RATE_SEG5							

Bit	Name	Function
7-0	MADPT_VSCALE_RATE_SEG5	**Vertical non-linear scale down 6th segment DDA increment value** The actual DDA increment is vscale={vscale5, vscale_low}.

DEINTERLACER 2F REG S2_2F, R/W

Bit	7	6	5	4	3	2	1	0
	MADPT_VSCALE_RATE_SEG6							

Bit	Name	Function
7-0	MADPT_VSCALE_RATE_SEG6	**Vertical non-linear scale down 7th segment DDA increment value** The actual DDA increment is vscale={vscale6, vscale_low}.

DEINTERLACER 30 REG S2_30, R/W

Bit	7	6	5	4	3	2	1	0
	MADPT_VSCALE_RATE_SEG7							

Bit	Name	Function
7-0	MADPT_VSCALE_RATE_SEG7	**Vertical non-linear scale down 8th segment DDA increment value** The actual DDA increment is vscale={vscale7, vscale_low}.

DEINTERLACER 31 REG S2_31, R/W

Bit	7	6	5	4	3	2	1	0
	MADPT_TEST_SEL				MADPT_TEST_EN	MADPT_SEL_PHASE_INI	MADPT_VSCALE_DEC_FACTOR	

Bit	Name	Function
1-0	MADPT_VSCALE_DEC_FACTOR	**Vertical non-linear scaling-down factor select** If the scaling ratio is less than ½, use it and DDA to generate the we and phase 00: scaling-ratio is more than ½. 01: scaling-ratio is less than ½. 10: scaling-ratio is less than ¼.
2	MADPT_SEL_PHASE_INI	**Vertical scaling down initial phase selection**
3	MADPT_TEST_EN	**Test bus output enable** Internal hardware debugging use only.
7-4	MADPT_TEST_SEL	**Test bus select** Internal hardware debugging use only.

DEINTERLACER 32 REG S2_32, R/W

Bit	7	6	5	4	3	2	1	0
	MADPT_NRD_SEL	MADPT_Y_VTAP_CNTRL			MADPT_Y_HTAP_CNTRL			

Bit	Name	Function
3-0	MADPT_Y_HTAP_CNTRL	**Y horizontal filter control for background reduction** Y_HTAP_CNTRL[3:0] could bypass four tap3 FIR filter.
6-4	MADPT_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	MADPT_NRD_SEL	**Background reduction selection control** Only set it to 1 in huge noise condition

DEINTERLACER 33 REG S2_33, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED	MADPT_M_VTAP_CNTRL			MADPT_M_HTAP_CNTRL			

Bit	Name	Function
3-0	MADPT_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	MADPT_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	RESERVED	

DEINTERLACER 34 REG S2_34, R/W

Bit	7	6	5	4	3	2	1	0
	MADPT_UV_WOUT			MADPT_UV_WOUT_BYPS	MADPT_Y_WOUT			MADPT_Y_WOUT_BYPS

Bit	Name	Function
0	MADPT_Y_WOUT_BYPS	**Bypass Y WOUT**
3-1	MADPT_Y_WOUT	**Coefficient for Y noise reduction**
4	MADPT_UV_WOUT_BYPS	**Bypass UV WOUT**
7-5	MADPT_UV_WOUT	**Coefficient for UV noise reduction**

DEINTERLACER 35 REG S2_35, R/W

Bit	7	6	5	4	3	2	1	0
	MADPT_CMP_USER_ID	MADPT_CMP_EN	MADPT_UVDLY_PD_BYPS	MADPT_NRD_VIIR_PD_BYPS	MADPT_DD0_SEL	MADPT_NRD_OUT_SEL	MADPT_UV_NRD_ENABLE	MADPT_Y_NRD_ENABLE

Bit	Name	Function
0	MADPT_Y_NRD_ENABLE	**Enable background noise reduction in Y domain** When set to 1, enable background noise reduction in Y domain.
1	MADPT_UV_NRD_ENABLE	**Enable background noise reduction in UV domain** When set to 1, enable background noise reduction in UV domain.
2	MADPT_NRD_OUT_SEL	**NRD output selection** Only set it to 1 in huge noise condition.
3	MADPT_DD0_SEL	**DD0 select control** Set it to 1 when background noise reduction enable Set it to 0 when background noise reduction disable
4	MADPT_NRD_VIIR_PD_BYPS	**Bypass NRD VIIR line buffer**
5	MADPT_UVDLY_PD_BYPS	**Bypass UV delay line buffer**
6	MADPT_CMP_EN	**Motion compare enable** When set to 1, enable motion compare When set to 0, motion compare is in manual mode
7	MADPT_CMP_USER_ID	**Motion compare result defined by user (in manual mode)** Motion compare result defined by user when CMP_EN = 0

DEINTERLACER 36 REG S2_36, R/W

Bit	7	6	5	4	3	2	1	0
	MADPT_CMP_LOW_THRESHOLD							

Bit	Name	Function
7-0	MADPT_CMP_LOW_THRESHOLD	**Motion compare low level threshold**

DEINTERLACER 37 REG S2_37, R/W

Bit	7	6	5	4	3	2	1	0
	MADPT_CMP_HIGH_THRESHOLD							

Bit	Name	Function
7-0	MADPT_CMP_HIGH_THRESHOLD	**Motion compare high level threshold**

DEINTERLACER 38 REG S2_38, R/W

Bit	7	6	5	4	3	2	1	0
	MADPT_NRD_VIIR_PD_ST				MADPT_NRD_VIIR_PD_SP			

Bit	Name	Function
3-0	MADPT_NRD_VIIR_PD_SP	NRD line buffer WRSTZ position adjustment
7-4	MADPT_NRD_VIIR_PD_ST	NRD line buffer RRSTZ position adjustment

DEINTERLACER 39 REG S2_39, R/W

Bit	7	6	5	4	3	2	1	0
	MADPT_UVDLY_PD_ST				MADPT_UVDLY_PD_SP			

Bit	Name	Function
3-0	MADPT_UVDLY_PD_SP	UV delay line buffer WRSTZ position adjustment
7-4	MADPT_UVDLY_PD_ST	UV delay line buffer RRSTZ position adjustment

DEINTERLACER 3A REG S2_3A, R/W

Bit	7	6	5	4	3	2	1	0
	MADPT_UV_MI_DET_BYPS	MADPT_MI_1BIT_DLY		MADPT_EN_STILL_FOR_PULLDWN	MADPT_EN_STILL_FOR_NRD	MADPT_EN_NOUT_FOR_NRD	MADPT_EN_PULLDWN_FOR_NRD	MADPT_EN_UV_DEINT

Bit	Name	Function
0	MADPT_EN_UV_DEINT	**Enable UV deinterlacer** When set to 1, enable UV deinterlacer.
1	MADPT_EN_PULLDWN_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.
2	MADPT_EN_NOUT_FOR_NRD	**Enable NOUT for background noise reduction**
3	MADPT_EN_STILL_FOR_NRD	**Enable still for background noise reduction**
4	MADPT_EN_STILL_FOR_PULLDWN	**Enable STILL to reset pull-down detection** When set to 1, still will be used to reset 3:2/2:2 pull-down detection.
6-5	MADPT_MI_1BIT_DLY	**Delay pipe control for motion index feedback-bit**
7	MADPT_UV_MI_DET_BYPS	**UV motion index generation bypass** When set to 1, UV motion index generation is in manual mode.

MADPT_MI_1BIT_DELAY	MI feedback-bit delay pipes
00	1
01	2
10	3
11	4

DEINTERLACER 3B REG S2_3B, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED	MADPT_UV_MI_OFFSET						

Bit	Name	Function
6-0	MADPT_UV_MI_OFFSET	**UV motion index offset** In auto mode, UV motion index offset In manual mode, UV motion index user defined value
7	RESERVED	

DEINTERLACER 3C REG S2_3C, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED	MADPT_MI_DELAY			MADPT_UV_MI_GAIN			

Bit	Name	Function
3-0	MADPT_UV_MI_GAIN	**UV motion index gain** UV motion index gain.
6-4	MADPT_MI_DELAY	**Motion index delay control** Control motion index (both Y and UV's) delay pipes, so that the motion index can align with corresponding data.
7	RESERVED	

MADPT_MI_DELAY	Motion index delay pipes
000	1 pipe
001	2 pipe
010	3 pipes
011	4 pipes
100	5 pipes
101	6 pipes
110	7 pipes
111	8 pipes

Chapter 03. HD_BYPS REGISTERS

HD_BYPS 00 REG S1_30, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED				HD_SEL_BLK_IN	HD_DYN_BYPS	HD_MATRIX_BYPS	HD_IN_DREG_BYPS

Bit	Name	Function
0	HD_IN_DREG_BYPS	**Input retiming bypass** Use the falling or rising edge of clock to get the input data. 0: Clock input data on the falling edge 1: Clock input date on the rising edge
1	HD_MATRIX_BYPS	**YUV2RGB conversion bypass control** Available only when input source is YUV source 0: YUV2RGB convert 1: bypass YUV2RGB function
2	HD_DYN_BYPS	**Dynamic range bypass control** If the input is YUV data, it must do dynamic range. 0: input is YUV data, do dynamic range . 1: input is RGB data, bypass dynamic range
3	HD_SEL_BLK_IN	**Blank select** Choose the input blank or generated blank use sync. 0: choose the blank that sync generated. 1: choose the input blank, if the input is DVI data.
7-4	RESERVED	

HD_BYPS 01 REG S1_31, R/W

Bit	7	6	5	4	3	2	1	0
	HD_Y_GAIN							

Bit	Name	Function
7-0	HD_Y_GAIN	**Dynamic range Y gain value** The gain value of Y dynamic range.

HD_BYPS 02 REG S1_32, R/W

Bit	7	6	5	4	3	2	1	0
	HD_Y_OFFSET							

Bit	Name	Function
7-0	HD_Y_OFFSET	**Dynamic range Y offset value** The offset value of Y dynamic range.

HD_BYPS 03 REG S1_33, R/W

Bit	7	6	5	4	3	2	1	0
	HD_U_GAIN							

Bit	Name	Function
7-0	HD_U_GAIN	**Dynamic range U gain value** The gain value of U dynamic range.

HD_BYPS 04 REG S1_34, R/W

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

Bit	Name	Function
7-0	HD_U_OFFSET [7:0]	**Dynamic range U offset value** The offset value of U dynamic range.

HD_BYPS 05 REG S1_35, R/W

Bit	7	6	5	4	3	2	1	0
	HD_V_GAIN							

Bit	Name	Function
7-0	HD_V_GAIN	**Dynamic range V gain value** The gain value of V dynamic range.

HD_BYPS 06 REG S1_36, R/W

Bit	7	6	5	4	3	2	1	0
	HD_V_OFFSET							

Bit	Name	Function
7-0	HD_V_OFFSET	**Dynamic range V offset value** The offset value of V dynamic range.

HD_BYPS 07 REG S1_37, R/W

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

Bit	Name	Function
7-0	HD_HSYNC_RST [7:0]	**Horizontal reset value** Horizontal counter reset value [7:0].

HD_BYPS 08 REG S1_38, R/W

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

Bit	Name	Function
2-0	HD_HSYNC_RST [10:8]	**Horizontal reset value** Horizontal counter reset value [10:8].
7-3	RESERVED	

HD_BYPS 09 REG S1_39, R/W

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

Bit	Name	Function
7-0	HD_INI_ST [7:0]	**Horizontal reset pulse start position** Vertical counter write enable, adjust the distance between hblank and vblank.

HD_BYPS 0A REG S1_3A, R/W

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

Bit	Name	Function
2-0	HD_INI_ST [10:8]	**Horizontal reset pulse start position** Vertical counter write enable, adjust the distance between hblank and vblank.
7-3	RESERVED	

HD_BYPS 0B REG S1_3B, R/W

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

Bit	Name	Function
7-0	HD_HB_ST [7:0]	**Horizontal blank start position** Generate horizontal blank to select programmed data.

HD_BYPS 0C REG S1_3C, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED				HD_HB_ST [11:8]			

Bit	Name	Function
3-0	HD_HB_ST [11:8]	**Horizontal blank start position** Generate horizontal blank to select programmed data.
7-4	RESERVED	

HD_BYPS 0D REG S1_3D, R/W

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

Bit	Name	Function
7-0	HD_HB_SP [7:0]	**Horizontal blank stop position** Generate horizontal blank to select programmed data.

HD_BYPS 0E REG S1_3E, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED				HD_HB_SP [11:8]			

Bit	Name	Function
3-0	HD_HB_SP [11:8]	**Horizontal blank stop position** Generate horizontal blank to select programmed data.
7-4	RESERVED	

HD_BYPS 0F REG S1_3F, R/W

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

Bit	Name	Function
7-0	HD_HS_ST [7:0]	**Horizontal sync start position** Output sync to DAC start position

HD_BYPS 10 REG S1_40, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED				HD_HS_ST [11:8]			

Bit	Name	Function
3-0	HD_HS_ST [11:8]	**Horizontal sync start position** Output sync to DAC start position
7-4	RESERVED	

HD_BYPS 11 REG S1_41, R/W

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

Bit	Name	Function
7-0	HD_HS_SP [7:0]	**Horizontal sync stop position** Output sync to DAC stop position

HD_BYPS 12 REG S1_42, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED				HD_HS_SP [11:8]			

Bit	Name	Function
3-0	HD_HS_SP [11:8]	**Horizontal sync stop position** Output sync to DAC stop position
7-4	RESERVED	

HD_BYPS 13 REG S1_43, R/W

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

Bit	Name	Function
7-0	HD_VB_ST [7:0]	**Vertical blank start position** Generate blank to select program data in blank

HD_BYPS 14 REG S1_44, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED				HD_VB_ST [11:8]			

Bit	Name	Function
3-0	HD_VB_ST [11:8]	**Vertical blank start position** Generate blank to select program data in blank
7-4	RESERVED	

HD_BYPS 15 REG S1_45, R/W

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

Bit	Name	Function
7-0	HD_VB_SP [7:0]	**Vertical blank stop position** Generate blank to select program data in blank

HD_BYPS 16 REG S1_46, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED				HD_VB_SP [11:8]			

Bit	Name	Function
3-0	HD_VB_SP [11:8]	**Vertical blank stop position** Generate blank to select program data in blank
7-4	RESERVED	

HD_BYPS 17 REG S1_47, R/W

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

Bit	Name	Function
7-0	HD_VS_ST [7:0]	**Vertical sync start position** Output vertical sync to DAC start position

HD_BYPS 18 REG S1_48, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED				HD_VS_ST [11:8]			

Bit	Name	Function
3-0	HD_VS_ST [11:8]	**Vertical sync start position** Output vertical sync to DAC start position
7-4	RESERVED	

HD_BYPS 19 REG S1_49, R/W

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

Bit	Name	Function
7-0	HD_VS_SP [7:0]	**Vertical sync stop position** Output vertical sync to DAC stop position

HD_BYPS 1A REG S1_4A, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED				HD_VS_SP [11:8]			

Bit	Name	Function
3-0	HD_VS_SP [11:8]	**Vertical sync stop position** Output vertical sync to DAC stop position
7-4	RESERVED	

HD_BYPS 1B REG S1_4B, R/W

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

Bit	Name	Function
7-0	HD_EXT_VB_ST [7:0]	**DVI mode vertical blank start position** Output vertical blank to DAC for DIV mode start position

HD_BYPS 1C REG S1_4C, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED				HD_EXT_VB_ST [11:8]			

Bit	Name	Function
3-0	HD_EXT_VB_ST [11:8]	**DVI mode vertical blank start position** Output vertical blank to DAC for DIV mode start position
7-4	RESERVED	

HD_BYPS 1D REG S1_4D, R/W

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

Bit	Name	Function
7-0	HD_EXT_VB_SP [7:0]	**DVI mode vertical blank stop position** Output vertical blank to DAC for DIV mode stop position

HD_BYPS 1E REG S1_4E, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED				HD_EXT_VB_SP [11:8]			

Bit	Name	Function
3-0	HD_EXT_VB_SP [11:8]	**DVI mode vertical blank stop position** Output vertical blank to DAC for DIV mode stop position
7-4	RESERVED	

HD_BYPS 1F REG S1_4F, R/W

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

Bit	Name	Function
7-0	HD_EXT_HB_ST [7:0]	**DVI mode horizontal blank start position** Output horizontal blank to DAC for DIV mode start position

HD_BYPS 20 REG S1_50, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED				HD_EXT_HB_ST [11:8]			

Bit	Name	Function
3-0	HD_EXT_HB_ST [11:8]	**DVI mode horizontal blank start position** Output horizontal blank to DAC for DIV mode start position
7-4	RESERVED	

HD_BYPS 21 REG S1_51, R/W

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

Bit	Name	Function
7-0	HD_EXT_HB_SP [7:0]	**DVI mode horizontal blank start position** Output horizontal blank to DAC for DIV mode stop position

HD_BYPS 22 REG S1_52, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED				HD_EXT_HB_SP [11:8]			

Bit	Name	Function
3-0	HD_EXT_HB_SP [11:8]	**DVI mode horizontal blank start position** Output horizontal blank to DAC for DIV mode stop position
7-4	RESERVED	

HD_BYPS 23 REG S1_53, R/W

Bit	7	6	5	4	3	2	1	0
	HD_BLK_GY_DATA							

Bit	Name	Function
7-0	HD_BLK_GY_DATA	**Programmed GY data in horizontal blank** Force the blank of GY data to the defined programmed data

HD_BYPS 24 REG S1_54, R/W

Bit	7	6	5	4	3	2	1	0
	HD_BLK_BU_DATA							

Bit	Name	Function
7-0	HD_BLK_BU_DATA	**Programmed BU data in horizontal blank** Force the blank of BU data to the defined programmed data

HD_BYPS 25 REG S1_55, R/W

Bit	7	6	5	4	3	2	1	0
	HD_BLK_RV_DATA							

Bit	Name	Function
7-0	HD_BLK_RV_DATA	**Programmed RV data in horizontal blank** Force the blank of BU data to the defined programmed data

Chapter 04. MISCELLANEOUS REGISTERS

PLL648 CONTROL 00 REG S0_40, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED	PLL_MS			PLL_ADS	PLL_IS	PLL_DIVBY2Z	PLL_CKIS

Bit	Name	Function
0	PLL_CKIS	**CKIS, PLL source clock selection** When = 0, PLL use OSC clock When = 1, PLL use input clock
1	PLL_DIVBY2Z	**DIVBY2Z, PLL source clock divide bypass** When = 0, PLL source clock divide by two When = 1, PLL source clock bypass divide
2	PLL_IS	**IS, ICLK source selection** When = 0, ICLK use PLL clock When = 1, ICLK use input clock
3	PLL_ADS	**ADS, input clock selection** When = 0, input clock is from PCLKIN(pin40) When = 1, input clock is from ADC
6-4	PLL_MS	**MS[2:0], memory clock control** When = 000, memory clock =108MHz When = 001, memory clock = 81MHz When = 010, memory clock from FBCLK (pin110) When = 011, memory clock = 162MHz When = 100, memory clock = 144MHz When = 101, memory clock = 185MHz When = 110, memory clock = 216MHz When = 111, memory clock = 129.6Mhz
7	RESERVED	**Reserved**

PLL648 CONTROL 01 REG S0_41, R/W

Bit	7	6	5	4	3	2	1	0
	PLL_4XV	PLL_2XV	PLL_VS4		PLL_VS2		PLL_VS	

Bit	Name	Function
1-0	PLL_VS	**VS[1:0]** Display clock tuning register
3-2	PLL_VS2	**VS2[1:0]** Display clock tuning register
5-4	PLL_VS4	**VS4[1:0]** Display clock tuning register
6	PLL_2XV	**2XV** Display clock tuning register
7	PLL_4XV	**4XV** Display clock tuning register

display clock freq (MHz)			resgiter setting				
VCLK	V2CLK	V4CLK	PLL_4XV	PLL_2XV	PLL_VS4	PLL_VS2	PLL_VS
27	54	108	1	x	00	00	00
27	54	54	0	1	00	01	00
27	27	27	0	0	00	01	01
32.4	64.8	129.6	1	x	01	00	00
32.4	64.8	64.8	0	1	01	01	00
32.4	32.4	32.4	0	0	01	01	01
40.5	81	162	1	x	10	00	00
40.5	81	81	0	1	10	01	00
40.5	40.5	40.5	0	0	10	01	01
54	108	108	1	x	00	01	00
54	54	54	0	1	00	01	01
64.8	129.6	129.6	1	x	01	01	00
64.8	64.8	64.8	0	1	01	01	01
81	162	162	1	x	10	01	00
81	81	81	0	1	10	01	01
108	108	108	1	x	00	01	01
129.6	129.6	129.6	1	x	01	01	01
162	162	162	1	x	10	01	01
FBCLK	FBCLK	FBCLK	1	x	11	01	01
PCLKIN	PCLKIN	PCLKIN	0	1	11	01	01
from ADC	from ADC	from ADC	0	0	11	01	01

Note: FBCLK is pin110, PCLKIN is pin40

PLL648 CONTROL 02 REG S0_42, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED							

Bit	Name	Function
7-0	RESERVED	**Reserved**

PLL648 CONTROL 03 REG S0_43, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED		PLL_VCORST	PLL_LEN	PLL_S		PLL_R	

Bit	Name	Function
1-0	PLL_R	**R[1:0]** Skew control for testing
3-2	PLL_S	**S[1:0]** Skew control for testing
4	PLL_LEN	**LEN** Lock Enable
5	PLL_VCORST	**VCORST** VCO control voltage reset bit When =1, reset VCO control voltage
7-6	RESERVED	**Reserved**

DAC CONTROL 00 REG S0_44, R/W

Bit	7	6	5	4	3	2	1	0
	DAC_RGBS_BPD	DAC_RGBS_G1EN	DAC_RGBS_G0ENZ	DAC_RGBS_GPD	DAC_RGBS_R1EN	DAC_RGBS_R0ENZ	DAC_RGBS_RPD	DAC_RGBS_PWDNZ

Bit	Name	Function
0	DAC_RGBS_PWDNZ	**DAC enable** When = 0, DAC (R,G,B,S) in power down mode When = 1, DAC (R,G,B,S) is enable
1	DAC_RGBS_RPD	**RPD, RDAC power down control** When = 0, RDAC work normally When = 1, RDAC is in power down mode
2	DAC_RGBS_R0ENZ	**R0ENZ, DAC min output bypass** When = 0, RDAC output Min voltage When = 1, RDAC output follow input R data
3	DAC_RGBS_R1EN	**R1EN, RDAC max output control** When = 0, RDAC output follow input R data When = 1, RDAC output Max voltage
4	DAC_RGBS_GPD	**GPD, GDAC power down control** When = 0, GDAC work normally When = 1, GDAC is in power down mode
5	DAC_RGBS_G0ENZ	**G0ENZ, GDAC min output bypass** When = 0, GDAC output Min voltage When = 1, GDAC output follow input G data
6	DAC_RGBS_G1EN	**G1EN, GDAC max output control** When = 0, GDAC output follow input G data When = 1, GDAC output Max voltage
7	DAC_RGBS_BPD	**BPD, BDAC power down control** When = 0, BDAC work normally When = 1, BDAC is in power down mode

DAC CONTROL 01 REG S0_45, R/W

Bit	7	6	5	4	3	2	1	0
	CKT_FF_CNTRL		RESERVED	DAC_RGBS_S1EN	DAC_RGBS_S0ENZ	DAC_RGBS_SPD	DAC_RGBS_B1EN	DAC_RGBS_B0ENZ

Bit	Name	Function
0	DAC_RGBS_B0ENZ	**B0ENZ, BDAC min output bypass** When = 0, BDAC output Min voltage When = 1, BDAC output follow input B data
1	DAC_RGBS_B1EN	**B1EN, BDAC max output control** When = 0, BDAC output follow input B data When = 1, BDAC output Max voltage
2	DAC_RGBS_SPD	**SPD, SDAC power down control** When = 0, GDAC work normally When = 1, GDAC is in power down mode
3	DAC_RGBS_S0ENZ	**S0ENZ, SDAC min output bypass** When = 0, SDAC output Min voltage When = 1, SDAC output follow input S data
4	DAC_RGBS_S1EN	**S1EN, SDAC max output control** When = 0, SDAC output follow input S data When = 1, SDAC output Max voltage
5	RESERVED	**Reserved**
7-6	CKT_FF_CNTRL	**CKT used to control FIFO**

RESET CONTROL 00 REG S0_46, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED	SFTRST_VDS_RSTZ	SFTRST_OSD_RSTZ	SFTRST_FIFO_RSTZ	SFTRST_MEM_RSTZ	SFTRST_MEM_FF_RSTZ	SFTRST_DEINT_RSTZ	SFTRST_IF_RSTZ

Bit	Name	Function
0	SFTRST_IF_RSTZ	**Input formatter reset control** When = 0, input formatter is in reset status When = 1, input formatter work normally
1	SFTRST_DEINT_RSTZ	**Deint_madpt3 reset control** When = 0, deint_madpt3 is in reset status When = 1, deint_madpt3 work normally
2	SFTRST_MEM_FF_RSTZ	**Mem_ff (wff/rff/playback/capture) reset control** When = 0, mem_ff is in reset status When = 1, mem_ff work normally
3	SFTRST_MEM_RSTZ	**Mem controller reset control** When = 0, mem controller is in reset status When = 1, mem controller work normally
4	SFTRST_FIFO_RSTZ	**FIFO reset control** When = 0, all FIFO (FF64/FF512) is in reset status When = 1, all FIFO work normally
5	SFTRST_OSD_RSTZ	**OSD reset control** When = 0, OSD generator is in reset status When = 1, OSD generator work normally
6	SFTRST_VDS_RSTZ	**Vds_proc reset control** When = 0, vds_proc is in reset status When = 1, vds_proc work normally
7	RESERVED	**Reserved**

RESET CONTROL 01 REG S0_47, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED			SFTRST_INT_RSTZ	SFTRST_HDBYPS_RSTZ	SFTRST_SYNC_RSTZ	SFTRST_MODE_RSTZ	SFTRST_DEC_RSTZ

Bit	Name	Function
0	SFTRST_DEC_RSTZ	**Decimation reset control** When = 0, decimation is in reset status When = 1, decimation work normally
1	SFTRST_MODE_RSTZ	**Mode detection reset control** When = 0, mode detection is in reset status When = 1, mode detection work normally
2	SFTRST_SYNC_RSTZ	**Sync procesor reset control** When = 0, sync processor is in reset status When = 1, sync processor work normally
3	SFTRST_HDBYPS_RSTZ	**HD bypass channel reset control** When = 0, HD bypass is in reset status When = 1, HD bypasswork normally
4	SFTRST_INT_RSTZ	**Interrupt generator reset control** When = 0, interrupt generator is in reset status When = 1, interrupt generator work normally
7-5	RESERVED	**Reserved**

PAD CONTROL 00 REG S0_48, R/W

	7	6	5	4	3	2	1	0
Bit	PAD_SYNC2_IN_ENZ	PAD_SYNC1_IN_ENZ	PAD_GIN_ENZ	PAD_GOUT_EN	PAD_RIN_ENZ	PAD_ROUT_EN	PAD_BIN_ENZ	PAD_BOUT_EN

Bit	Name	Function
0	PAD_BOUT_EN	**VB_[7:0] output control** When = 0, disable VB_[7:0] (test_out_[7:0]) output When = 1, enable VB_[7:0] (test_out_[7:0]) output
1	PAD_BIN_ENZ	**VB_[7:0] input control** When = 0, enable VB_[7:0] input When = 1, disable VB_[7:0] input
2	PAD_ROUT_EN	**VR_[7:0] output control** When = 0, disable VR_[7:0] (test_out_[15:8]) output When = 1, enable VR_[7:0] (test_out_[15:8]) output
3	PAD_RIN_ENZ	**VR_[7:0] input control** When = 0, enable VR_[7:0] input When = 1, disable VR_[7:0] input
4	PAD_GOUT_EN	**VG_[7:0] output control** When = 0, disable VG_[7:0] (test_out_[23:16]) output When = 1, enable VG_[7:0] (test_out_[23:16]) output
5	PAD_GIN_ENZ	**VG_[7:0] input control** When = 0, enable VG_[7:0] input When = 1, disable VG_[7:0] input
6	PAD_SYNC1_IN_ENZ	**H/V sync1 input control** When = 0, enable H/V sync1 input filter When = 1, disable H/V sync1 input filter
7	PAD_SYNC2_IN_ENZ	**H/V sync2 input control** When = 0, enable H/V sync2 input filter When = 1, disable H/V sync2 input filter

PAD CONTROL 01 REG S0_49, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED	PAD_PLUP_ENZ	PAD_PLDN_ENZ	PAD_TRI_ENZ	PAD_BLK_OUT_ENZ	PAD_SYNC_OUT_ENZ	PAD_CKOUT_ENZ	PAD_CKIN_ENZ

Bit	Name	Function
0	PAD_CKIN_ENZ	**PCLKIN control** When = 0, PCLKIN input enable When = 1, PCLKIN input disable
1	PAD_CKOUT_ENZ	**CLKOUT control** When = 0, CLKOUT output enable When = 1, CLKOUT output disable
2	PAD_SYNC_OUT_ENZ	**HSOUT/VSOUT control** When = 0, HSOUT/VSOUT output enable When = 1, HSOUT/VSOUT output disable
3	PAD_BLK_OUT_ENZ	**HBOUT/VBOUT control** When = 0, HBOUT/VBOUT output enable When = 1, HBOUT/VBOUT output disable
4	PAD_TRI_ENZ	**Tri-state gate control** When = 0, enable output pad's tri-state gate When = 1, disable output pad's tri-state gate
5	PAD_PLDN_ENZ	**Pull-down control** When = 0, enable pad's pull-down transistor When = 1, disable pad's pull-down transistor
6	PAD_PLUP_ENZ	**Pull-up control** When = 0, enable pad's pull-up transistor When = 1, disable pad's pull-up transistor
7	RESERVED	**Reserved**

PAD CONTROL 02 REG S0_4A, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED				PAD_XTOUT_TTL	PAD_OSC_CNTRL		

Bit	Name	Function
2-0	PAD_OSC_CNTRL	**OSC pad C2/C1/C0 control** OSC pad C2/C1/C0 control
3	PAD_XTOUT_TTL	**OSC pad output control** When = 0, enable OSC pad output by schmitt When = 1, enable OSC pad output by TTL
7-4	RESERVED	**Reserved**

DAC_MUX CONTROL 00 REG S0_4B, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED					DAC_RGBS_ADC2DAC	DAC_RGBS_BYPS2DAC	DAC_RGBS_BYPS_IREG

Bit	Name	Function
0	DAC_RGBS_BYPS_IREG	**DAC input DFF control** When = 0, DAC input DFF is falling edge D-flipflop When = 1, bypass falling edge D-flipflop
1	DAC_RGBS_BYPS2DAC	**HD bypass to DAC control** When = 0, disable HD bypass channel to DAC When = 1, enable HD bypass channel to DAC directly
2	DAC_RGBS_ADC2DAC	**ADC to DAC control** When = 0, disable ADC (with decimation) to DAC When = 1, enable ADC (with decimation) to DAC directly
7-3	RESERVED	**Reserved**

TEST BUS CONTROL 00 REG S0_4D, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED		TEST_BUS_EN	TEST_BUS_SEL				

Bit	Name	Function
4-0	TEST_BUS_SEL	**Test bus selection**
5	TEST_BUS_EN	**Test bus enable** When = 0, disable test bus output When = 1, test bus output to VR_[7:0], VB_[7:0] (test_out_[15:0])
7-6	RESERVED	**Reserved**

DIG_OUT CONTROL 00 REG S0_4E, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED						DIGOUT_ADC2PAD	DIGOUT_BYPS2PAD

Bit	Name	Function
0	DIGOUT_BYPS2PAD	**HD bypass channel to digital output control** When = 0, disable HD bypass to digital output When = 1, enable HD bypass to digital output (VG_[7:0], VR_[7:0], VB_[7:0])
1	DIGOUT_ADC2PAD	**ADC to digital output control** When = 0, disable ADC to digital output When = 1, enable ADC (with decimation) to digital output (VG, VR, VB)
7-2	RESERVED	**Reserved**

CLK/SYNC CONTROL 00 REG S0_4F, R/W

Bit	7	6	5	4	3	2	1	0
	OUT_SYNC_SEL		OUT_SYNC_CNTRL	OUT_CLK_EN	OUT_CLK_MUX		OUT_CLK_PHASE_CNTRL	DAC_RGBS_V4CLK_INVT

Bit	Name	Function
0	DAC_RGBS_V4CLK_INVT	**V4CLK invert control** When = 0, V4CLK to DAC directly When = 1, V4CLK will invert before go to DAC
1	OUT_CLK_PHASE_CNTRL	**CLKOUT invert control** When = 0, CLKOUT output no invert When = 1, CLKOUT will invert before output
3-2	OUT_CLK_EN	**CLKOUT selection control** When = 00, CLKOUT = V4CLK When = 01, CLKOUT = V2CLK When = 10, CLKOUT = VCLK When = 11, CLKOUT = ADC output clock
4	CLKOUT_EN	**CLKOUT enable control** When = 0, disable CLKOUT to PAD When = 1, enable CLKOUT to PAD
5	OUT_SYNC_CNTRL	**H/V sync output enable** When = 0, disable H/V sync output to PAD When = 1, enable H/V sync output to PAD
7-6	OUT_SYNC_SEL	**H/V sync output selection control** When = 00, H/V sync output are from vds_proc When = 01, H/V sync output are from HD bypass When = 10, H/V sync output are from sync processor When = 11, reserved

BLANK CONTROL 00 REG S0_50, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED		IN_BLANK_IREG_BYPS	IN_BLANK_SEL	RESERVED		OUT_BLANK_SEL_1	OUT_BLANK_SEL_0

Bit	Name	Function
0	OUT_BLANK_SEL_0	**HBOUT/VBUT selection control** When = 0, VBOUT output Vertical Blank When = 1, VBOUT output composite Display Enable
1	OUT_BLANK_SEL_1	**HBOUT/VBOUT selection control** When = 0, HBOUT/VBOUT is from vds_proc When = 1, HBOUT/VBOUT is from HD bypass
3-2	RESERVED	**Reserved**
4	IN_BLANK_SEL	**Input blank selection** When = 0, disable input composite Display Enable When = 1, enable input composite Display Enable
5	IN_BLANK_IREG_BYPS	**Input blank IREG bypas** When = 0, input composite Display Enable latched by falling edge DFF When = 1, bypass falling edge DFF
7-6	RESERVED	**Reserved**

GPIO CONTROL 00 REG S0_52, R/W

	7	6	5	4	3	2	1	0
Bit	GPIO_SEL_7	GPIO_SEL_6	GPIO_SEL_5	GPIO_SEL_4	GPIO_SEL_3	GPIO_SEL_2	GPIO_SEL_1	GPIO_SEL_0

Bit	Name	Function
0	GPIO_SEL_0	**GPIO bit0 selection** When = 0, GPIO (pin76) is used as INTZ output When = 1, GPIO (pin76) is used as GPIO bit0
1	GPIO_SEL_1	**GPIO bit1 selection** When = 0, HALF (pin77) is used as half tone input When = 1, HALF (pin77) is used as GPIO bit1
2	GPIO_SEL_2	**GPIO bit2 selection** When = 0, SCLSA (pin43) is used as two wire serial bus slave address selection When = 1, SCLSA (pin43) is used as GPIO bit2
3	GPIO_SEL_3	**GPIO bit3 selection** When = 0, MBA (pin107) is used as external memory BA When = 1, MBA (pin107) is used as GPIO bit3
4	GPIO_SEL_4	**GPIO bit4 selection** When = 0, MCS1 (pin109) is used as external memory CS1 When = 1, MCS1 (pin109) is used as GPIO bit4
5	GPIO_SEL_5	**GPIO bit5 selection** When = 0, HBOUT (pin6) is used as H-blank output When = 1, HBOUT (pin6) is used as GPIO bit5
6	GPIO_SEL_6	**GPIO bit6 selection** When = 0, VBOUT (pin7) is used as V-blank output When = 1, VBOUT (pin7) is used as GPIO bit6
7	GPIO_SEL_7	**GPIO bit7 selection** When = 0, CLKOUT (pin4) is used as clock output When = 1, CLKOUT (pin4) is used as GPIO bit7

GPIO CONTROL 01 REG S0_53, R/W

Bit	7	6	5	4	3	2	1	0
	GPIO_EN_7	GPIO_EN_6	GPIO_EN_5	GPIO_EN_4	GPIO_EN_3	GPIO_EN_2	GPIO_EN_1	GPIO_EN_0

Bit	Name	Function
0	GPIO_EN_0	**GPIO bit0 output enable** When = 0, GPIO bit0 output disable When = 1, GPIO bit0 output enable
1	GPIO_EN_1	**GPIO bit1 output enable** When = 0, GPIO bit1 output disable When = 1, GPIO bit1 output enable
2	GPIO_EN_2	**GPIO bit2 output enable** When = 0, GPIO bit2 output disable When = 1, GPIO bit2 output enable
3	GPIO_EN_3	**GPIO bit3 output enable** When = 0, GPIO bit3 output disable When = 1, GPIO bit3 output enable
4	GPIO_EN_4	**GPIO bit4 output enable** When = 0, GPIO bit4 output disable When = 1, GPIO bit4 output enable
5	GPIO_EN_5	**GPIO bit5 output enable** When = 0, GPIO bit5 output disable When = 1, GPIO bit5 output enable
6	GPIO_EN_6	**GPIO bit6 output enable** When = 0, GPIO bit6 output disable When = 1, GPIO bit6 output enable
7	GPIO_EN_7	**GPIO bit7 output enable** When = 0, GPIO bit7 output disable When = 1, GPIO bit7 output enable

GPIO CONTROL 02 REG S0_54, R/W

Bit	7	6	5	4	3	2	1	0
	GPIO_VAL_7	GPIO_VAL_6	GPIO_VAL_5	GPIO_VAL_4	GPIO_VAL_3	GPIO_VAL_2	GPIO_VAL_1	GPIO_VAL_0

Bit	Name	Function
0	GPIO_VAL_0	GPIO bit0 output value
1	GPIO_VAL_1	GPIO bit1 output value
2	GPIO_VAL_2	GPIO bit2 output value
3	GPIO_VAL_3	GPIO bit3 output value
4	GPIO_VAL_4	GPIO bit4 output value
5	GPIO_VAL_5	GPIO bit5 output value
6	GPIO_VAL_6	GPIO bit6 output value
7	GPIO_VAL_7	GPIO bit7 output value

INVT_RING CONTROL 00 REG S0_57, R/W

Bit	7	6	5	4	3	2	1	0
	INVT_RING_EN	RESERVED						

Bit	Name	Function
6-0	RESERVED	Reserved
7	INVT_RING_EN	Enable invert ring When = 0, disable invert ring When = 1, enable invert ring for processing test

INTERRUPT CONTROL 00 REG S0_58, R/W

Bit	7	6	5	4	3	2	1	0
	INT_RST7	INT_RST6	INT_RS5	INT_RST4	INT_RST3	INT_RST2	INT_RST1	INT_RST0

Bit	Name	Function
0	INT_RST_0	**Interrupt bit0 reset control** When = 1, interrupt bit0 status will be reset to zero
1	INT_RST_1	**Interrupt bit1 reset control** When = 1, interrupt bit1 status will be reset to zero
2	INT_RST_2	**Interrupt bit2 reset control** When = 1, interrupt bit2 status will be reset to zero
3	INT_RST_3	**Interrupt bit3 reset control** When = 1, interrupt bit3 status will be reset to zero
4	INT_RST_4	**Interrupt bit4 reset control** When = 1, interrupt bit4 status will be reset to zero
5	INT_RST_5	**Interrupt bit5 reset control** When = 1, interrupt bit5 status will be reset to zero
6	INT_RST_6	**Interrupt bit6 reset control** When = 1, interrupt bit6 status will be reset to zero
7	INT_RST_7	**Interrupt bit7 reset control** When = 1, interrupt bit7 status will be reset to zero

INTERRUPT CONTROL 01 REG S0_59, R/W

Bit	7	6	5	4	3	2	1	0
	INT_ENABLE7	INT_ENABLE6	INT_ENABLE5	INT_ENABLE4	INT_ENABLE3	INT_ENABLE2	INT_ENABLE1	INT_ENABLE0

Bit	Name	Function
0	INT_ENABLE0	**Interrupt bit0 enable** When = 1, enable interrupt bit0 generator
1	INT_ENABLE1	**Interrupt bit1 enable** When = 1, enable interrupt bit1 generator
2	INT_ENABLE2	**Interrupt bit2 enable** When = 1, enable interrupt bit2 generator
3	INT_ENABLE3	**Interrupt bit3 enable** When = 1, enable interrupt bit3 generator
4	INT_ENABLE4	**Interrupt bit4 enable** When = 1, enable interrupt bit4 generator
5	INT_ENABLE5	**Interrupt bit5 enable** When = 1, enable interrupt bit5 generator
6	INT_ENABLE6	**Interrupt bit6 enable** When = 1, enable interrupt bit6 generator
7	INT_ENABLE7	**Interrupt bit7 enable** When = 1, enable interrupt bit7 generator

Chapter 05. MEMORY REGISTERS

MEMORY CONTROLLER 00 — REG S4_00, R/W

Bit	7	6	5	4	3	2	1	0
	MEM_INI_REG							

Bit	Name	Function
1-0	MEM_INI_REG [1:0]	**SDRAM Idle Period Control and IDLE Done Select**: (default 0) MEM_INI_REG [1:0] / #of VS (vertical syn) 0 0 — 2 0 1 — 3 1 0 — 1 1 1 — R_MSOFTH
2	MEM_INI_REG [2]	**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.
3	MEM_INI_REG [3]	**Reserved**
4	MEM_INI_REG [4]	**SDRAM Reset Signal:** When this bit is 1, will generate 5-mmclk pulse, and reset memory controller timing, data pipe and state machine;
5	MEM_INI_REG [5]	**Reserved**
6	MEM_INI_REG [6]	**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	MEM_INI_REG [7]	**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 S4_01, R/W

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

Bit	Name	Function
7-0	MEM_MODE_REG [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 S4_02, R/W

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

Bit	Name	Function
3-0	MEM_MODE_REG [11:8]	**SDRAM Mode Information High 4 bits** Reserved for future usage.
7-4	MEM_MODE_REG [15:12]	**Reserved**

MEMORY CONTROLLER 03 REG S4_03, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED		MEM_INI_REF_CYC	MEM_MODE_CYC	MEM_MODE_CS1	MEM_MODE_CS0	MEM_MODE_BA1	MEM_MODE_BA0

Bit	Name	Function
0	MEM_MODE_BA0	**Bank0 Select Value In Load Mode Register Cycle :** This register 's aim is compatible with more sdram chips
1	MEM_MODE_BA1	**Bank1 Select Value In Load Mode Register Cycle :** This register 's aim is compatible with more sdram chips
2	MEM_MODE_CS0	**Chip0 Select Value in Load Mode Register Cycle :** This register 's aim is compatible with more sdram chips
3	MEM_MODE_CS1	**Chip1 Select Value in Load Mode Register Cycle :** **This register 's aim is compatible with more sdram chips**
4	MEM_MODE_CYC	**Mode Cycle Period Select** When this bit is 1, then mode cycle for memory initialization will be 3 clocks, otherwise be 2 clocks ;
5	MEM_INI_REF_CYC	**Initial Cycle Refresh Period Clock Number Select:** 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.**
7-6	RESERVED	**RESERVED**

MEMORY CONTROLLER 04 REG S4_04, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED	MEM_RD_LAT_PIP			RESERVED	MEM_FK_RD_DLY		

Bit	Name	Function
2-0	MEM_FK_RD_DLY	**SDRAM Rising Edge Clock Delay for Latching Read Data: (default set 3'b 000); With DLY8LV and NCASDLY**

MEM_FK_RD_DLY [2:0]			#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

Bit	Name	Function
3	RESERVED	**RESERVED**
6-4	MEM_RD_LAT_PIP	**SDRAM Latch Signal Generate Timing For Memory Data Read Cycle:** latency =2 Set 3'b000 ;latency =3 set 3'b011 ;

MEM_RD_LAT_PIP [2:0]			#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

Bit	Name	Function
7	RESERVED	**RESERVED**

MEMORY CONTROLLER 05 REG S4_05, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED		MEM_PCHG_CYCLE		RESERVED		MEM_ACT_CYCLE	

Bit	Name	Function
1-0	MEM_ACT_CYCLE	**Number of Memory Clock For SDRAM Active Cycle:** MEM_ACT_CYCLE [1:0] / # of Mclk 0 0 / 2 0 1 / 3 1 0 / 4 1 1 / 5
3-2	RESERVED	**RESERVED**
5-4	MEM_PCHG_CYCLE	**Number of Memory Clock For SDRAM Precharge Cycle:** **MEM_PCHG_CYCLE [1:0] / # of Mclk** 0 0 / 2 0 1 / 3 1 0 / 4 1 1 / 5
7-6	RESERVED	**RESERVED**

MEMORY CONTROLLER 06 REG S4_06, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED	MEM_REF_CYCLE			RESERVED	MEM_REF_RATE		

<table>
<tr><th>Bit</th><th>Name</th><th>Function</th></tr>
<tr><td>2-0</td><td>MEM_REF_RATE</td><td>For VGA Mode of Refresh Cycle:
<table>
<tr><th colspan="3">MEM_REF_RATE [2:0]</th><th>#of refresh</th></tr>
<tr><td>0</td><td>0</td><td>0</td><td>3</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>5</td></tr>
<tr><td>0</td><td>1</td><td>X</td><td>1</td></tr>
<tr><td>1</td><td>0</td><td>X</td><td>2</td></tr>
<tr><td>1</td><td>1</td><td>X</td><td>4</td></tr>
</table></td></tr>
<tr><td>3</td><td>RESERVED</td><td>RESERVED</td></tr>
<tr><td>6-4</td><td>MEM_REF_CYCLE</td><td>Number of Memory Clock For SDRAM Refresh Cycle:
<table>
<tr><th colspan="3">MEM_REF_CYCLE [2:0]</th><th># Of Mclk</th></tr>
<tr><td>0</td><td>0</td><td>0</td><td>6</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>7</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>8</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>9</td></tr>
<tr><td>1</td><td>X</td><td>X</td><td>10</td></tr>
</table></td></tr>
<tr><td>7</td><td>RESERVED</td><td>RESERVED</td></tr>
</table>

MEMORY CONTROLLER 07 REG S4_07, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED	MEM_TRAS_SEL			RESERVED	MEM_TWR_SEL		

Bit	Name	Function
2-0	MEM_TWR_SEL	**TWR Period Select (Number of Memory Clock Inserted from Last Write Cycle to Precharge)** MEM_TWR_SEL [2:0] / #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
3	RESERVED	**RESERVED**
6-4	MEM_TRAS_SEL	**TRAS Timing (from active cycle to precharge cycle)** MEM_TRAS_SEL [2:0] / # OF MCLK 0 0 0 — 3 0 0 1 — 4 0 1 0 — 5 0 1 1 — 6 1 X X — 7
7	RESERVED	**RESERVED**

MEMORY CONTROLLER 08 REG S4_08, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED		MEM_W2R_SEL_CYC		RESERVED		MEM_R2W_NOP_CYC	

Bit	Name	Function
1-0	MEM_R2W_NOP_CYC	**Number of Dummy Clock For SDRAM Read to Write Cycle:** MEM_R2W_NOP_CYC [1:0] / #OF MCLK 0 0 / 0 0 1 / 1 1 0 / 2 1 1 / 3
3-2	RESERVED	**RESERVED**
5-4	MEM_W2R_SEL_CYC	**Memory Write to Read Dummy Clock Cycle Insertion:** MEM_W2R_SEL_CYC [1:0] / #OF MCLK 0 0 / 0 0 1 / 1 1 0 / 2 1 1 / 3
7-6	RESERVED	**RESERVED**

MEMORY CONTROLLER 09 REG S4_09, R/W

Bit	7 6	5 4	3 2	1 0
	MEM_CS1_SEL	MEM_CS0_SEL	MEM_BK1_SEL	MEM_BK0_SEL

Bit	Name	Function
1-0	MEM_BK0_SEL	**Bank Select Address Mux:** MEM_BK0_SEL [1:0] / # OF ADDRESS BIT 0 0 / ADR 19 0 1 / ADR 20 1 0 / ADR 21 1 1 / NOP
3-2	MEM_BK1_SEL	**Bank Select Address Mux:** MEM_BK1_SEL [1:0] / # OF ADDRESS BIT 0 0 / ADR 19 0 1 / ADR 20 1 0 / ADR 21 1 1 / NOP
5-4	MEM_CS0_SEL	**Bank Select Address Mux:** MEM_CS0_SEL [1:0] / # OF ADDRESS BIT 0 0 / ADR 19 0 1 / ADR 20 1 0 / ADR 21 1 1 / NOP
7-6	MEM_CS1_SEL	**Bank Select Address Mux:** MEM_CS1_SEL [1:0] / # OF ADDRESS BIT 0 0 / ADR 19 0 1 / ADR 20 1 0 / ADR 21 1 1 / NOP

MEMORY CONTROLLER 10 REG S4_0A, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED		MEM_ROW_ST_SEL		RESERVED		MEM_COL_ST_SEL	

Bit	Name	Function
0	MEM_COL_ST_SEL [0]	**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	MEM_COL_ST_SEL [1]	**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
3-2	RESERVED	**RESERVED**
4	MEM_ROW_ST_SEL [0]	**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.
5	MEM_ROW_ST_SEL [1]	**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.
7-6	RESERVED	**RESERVED**

MEMORY CONTROLLER 11 REG S4_0B, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED	MEM_ADR_REG						

Bit	Name	Function
0	MEM_ADR_REG [0]	**Memory Row Address Precharge enable for Address 8** When this bit is 1,address 8 changed will do precharge; When this bit is 0,address 8 changed will not do precharge.
1	MEM_ADR_REG [1]	**Memory Row Address Precharge enable for Address 9** When this bit is 1,address 9 changed will do precharge; When this bit is 0,address 9 changed will not do precharge.
2	MEM_ADR_REG [2]	**Memory Row Address Precharge enable for Address 10** When this bit is 1,address 10 changed will do precharge; When this bit is 0,address 10 changed will not do precharge.
3	MEM_ADR_REG [3]	**Memory Row Address Precharge enable for Address18** When this bit is 1,address 20 changed will do precharge; When this bit is 0,address 20 changed will not do precharge.
4	MEM_ADR_REG [4]	**Memory Row Address Precharge enable for Address 19** When this bit is 1,address 19 changed will do precharge; When this bit is 0,address 19 changed will not do precharge.
5	MEM_ADR_REG [5]	**Memory Row Address Precharge enable for Address 20** When this bit is 1,address 20 changed will do precharge; When this bit is 0,address 20 changed will not do precharge.
6	MEM_ADR_REG [6]	**Memory Row Address Precharge enable for Address 21** When this bit is 1,address 21 changed will do precharge; When this bit is 0,address 21 changed will not do precharge.
7	RESERVED	**RESERVED**

MEMORY CONTROLLER 12 REG S4_0C, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED				MEM_COL_ADR_VLD			

Bit	Name	Function
0	MEM_COL_ADR_VLD [0]	**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	MEM_COL_ADR_VLD [1]	**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	MEM_COL_ADR_VLD [2]	**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	MEM_COL_ADR_VLD [3]	**Memory Column Address Enable For Address bit 11 (default value 0)** For others SDRAM 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.
7-4	RESERVED	**RESERVED**

MEMORY CONTROLLER 13 REG S4_0D, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED	MEM_CS1_BA1_SEL	MEM_CS0_BA0_SEL	MEM_BA_ADR11_SEL	RESERVED	MEM_SPECIAL_PIN		

Bit	Name	Function
0	MEM_SPECIAL_PIN [0]	**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.
1	MEM_SPECIAL_PIN [1]	**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.
2	MEM_SPECIAL_PIN [2]	**Special Pin10 For Precharge: default value 0** 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.
3	RESERVED	**RESERVED**
4	MEM_BA_ADR11_SEL	**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;
5	MEM_CS0_BA0_SEL	**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;
6	MEM_CS1_BA1_SEL	**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;
7	RESERVED	**RESERVED**

MEMORY CONTROLLER 14 REG S4_0E, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED						MEM_CMD_PIPE	

Bit	Name	Function
0	MEM_CMD_PIPE [0]	**SDRAM WE Command Pipe Select:** When it is at 0,WE signal pass through a pipe, or it will bypass a pipe;
1	MEM_CMD_PIPE [1]	**SDRAM CAS Command Pipe Select:** When it is at 0,CAS signal pass through a pipe, or it will bypass a pipe;
2	MEM_CMD_PIPE [2]	**SDRAM RAS Command Pipe Select:** When it is at 0, RAS signal pass through a pipe, or it will bypass a pipe;
7-3	RESERVED	**RESERVED**

MEMORY CONTROLLER 15 REG S4_0F, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED						MEM_FST_REG	

Bit	Name	FUNCTION
0	MEM_FST_REG [0]	**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,
1	MEM_FST_REG [1]	**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
2	MEM_FST_REG [2]	**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
3	MEM_FST_REG [3]	**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
7-4	RESERVED	**RESERVED**

MEMORY CONTROLLER 16 REG S4_10, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED			MEM_MISC_REG				

Bit	Name	FUNCTION
0	MEM_MISC_REG [0]	**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	MEM_MISC_REG [1]	**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	MEM_MISC_REG [2]	**SDRAM SAFE REAE/WRITE OPERATION** Don't care, default value 0, When this bit sets 1, will make state machine calculate up during read/write operation. When this bit sets 0, will make state machine hold during read/write operation
3	MEM_MISC_REG [3]	**Add No Operation For Precharge Cycle** Don't care, default value 0; When this bit sets 0, will add NOP for precharge cycle ; When this bit sets 1,will no add NOP for precharge cycle.
4	MEM_MISC_REG [4]	**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;
7-5	RESERVED	**RESERVED**

MEMORY CONTROLLER 17 REG S4_11, R/W

Bit	7	6	5	4	3	2	1	0
	MEM_FBK_PATH_SEL	RESERVED		MEM_FBK_INV_PATH_SEL	RESERVED	MEM_FBK_CS2_SEL	MEM_FBK_SEL_MCLK	MEM_FBK_CLK_SEL

Bit	Name	Function
0	MEM_FBK_CLK_SEL	**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 ;
1	MEM_FBK_SEL_MCLK	**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.
2	MEM_FBK_CS2_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.
3	RESERVED	**RESERVED**
4	MEM_FBK_INV_PATH_SEL	**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.
6-5	RESERVED	**RESERVED**
7	MEM_FBK_PATH_SEL	**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.

MEMORY CONTROLLER 18 REG S4_12, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED					MEM_FBK_CLK_DLYCELL_SEL	MEM_CLK_DLYCELL_SEL	MEM_INTER_DLYCELL_SEL

Bit	Name	Function
0	MEM_INTER_DLYCELL_SEL	**Select SDRAM Delay Cell:** This register is control the delay of data/address/command When it is at 0, select bypass delay cell, when it is at 1, select DLY8LV cell.
1	MEM_CLK_DLYCELL_SEL	**Select SDRAM Delay Cell:** This register is only control the delay of clock send to PAD When it is at 0, select bypass delay cell, when it is at 1, select DLY8LV cell.
2	MEM_FBK_CLK_DLYCELL_SEL	**Select SDRAM Delay Cell:** This register is only control the delay of feed back clock. When it is at 0, select bypass delay cell, when it is at 1, select DLY8LV cell.
7-3	RESERVED	**RESERVED**

MEMORY CONTROLLER 19 REG S4_13, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED					MEM_FBK_CLK_INVERT	MEM_RD_DATA_CLK_INVERT	MEM_PAD_CLK_INVERT

Bit	Name	Function
0	MEM_PAD_CLK_INVERT	**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	MEM_RD_DATA_CLK_INVERT	**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.
2	MEM_FBK_CLK_INVERT	**Control feedback clock register** When this bit is at 1, will invert feedback clock; When it's at 0, will bypass feedback clock;
7-3	RESERVED	**RESERVED**

MEMORY CONTROLLER 20 REG S4_14, R/W

Bit	7	6	5	4	3	2	1	0
	MEM_MBUS32OR16_SEL	RESERVED		MEM_WRITE_CYCL_CTL	RESERVED	MEM_NEW_FUNC_CTL		

Bit	Name	Function
0	MEM_NEW_FUNC_CTL [0]	**LATCH READ DATA ADDED A PIPE** When this register sets 1, latch signal will add a pipe; When this register sets 0, no change.
1	MEM_NEW_FUNC_CTL [1]	**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.
2	MEM_NEW_FUNC_CTL [2]	**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.
3	RESERVED	**RESERVED**
4	MEM_WRITE_CYCL_CTL	**Control Read cycle to Write cycle** When this bit sets 1, read cycle hold will enter write cycle directly. When this bit sets 0, will not enter write cycle directly. This bit register is for save bandwidth, reduce read to write nop.
6-5	RESERVED	**RESERVED**
7	MEM_MBUS32OR16_SEL	**Memory Bus 32-bit to 16-bit transfer** When this bit sets 1, memory bus is 32-bit. When this bit set2 0, memory bus is 16-bit.

MEMORY CONTROLLER 21 REG S4_15, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED					MEM_REQ_WFF_CAP	MEM_REQ_PB_RFF_CAP	MEM_REQ_PBH_RFFH

Bit	Name	FUNCTION
0	MEM_REQ_PBH_RFFH	**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	MEM_REQ_PB_RFF_CAP	**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	MEM_REQ_WFF_CAP	**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
7-3	RESERVED	**RESERVED**

MEMORY CONTROLLER 22 REG S4_16, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED					MEM_TEST_SEL		

Bit	Name	FUNCTION
2-0	MEM_TEST_SEL	**Test Logic Controll** Select four groups test signals (internal hardware debug use only)
7-3	RESERVED	**RESERVED**

MEMORY CONTROLLER 23 REG S4_17, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED		MEM_WOEZ_SEL_DLYCELL	MEM_WOEZ_PIP	RESERVED	MEM_WOEZ_DLY		

Bit	Name	FUNCTION
2-0	MEM_WOEZ_DLY	**Data TRI_STATE Enable Delay Control Bits: with DLY8LV and NCASDLY**
3	RESERVED	**RESERVED**
4	MEM_WOEZ_PIP	**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 the other registers.**
5	MEM_WOEZ_SEL_DLYCELL	**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 **.**
7-6	RESERVED	**RESERVED**

MEM_WOEZ_DLY [2:0]			#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 24 REG S4_18, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED		MEM_WR_DATA_PIP		RESERVED	MEM_DATA_DLY_REG		

Bit	Name	Function
2-0	MEM_DATA_DLY_REG	**Data Delay Control Bits: with DLY8LV** MEM_DATA_DLY_REG [2:0] / #OF NS 0 0 0 — 0.00 0 0 1 — 0.25 0 1 0 — 0.50 0 1 1 — 0.75 1 0 0 — 1.00 1 0 1 — 1.50 1 1 0 — 2.00 1 1 1 — 3.00
3	RESERVED	**RESERVED**
5-4	MEM_WR_DATA_PIP	**Memory Write Data (Rising Edge) Pipe Select: default value 2'b00; (In 5705,only 2'b00)** MEM_WR_DATA_PIP [1:0] / # OF PIPE 0 0 — 0 0 1 — 1
7-6	RESERVED	**RESERVED**

MEMORY CONTROLLER 25 REG S4_19, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED	MEM_CAS_DLY_REG			RESERVED	MEM_RAS_DLY_REG		

Bit	Name	Function
2-0	MEM_RAS_DLY_REG	RAS Delay Control bits: default value 3'b000;with DLY8LV

MEM_RAS_DLY_REG [2:0]			#OF NS
0	0	0	0.00
0	0	1	0.25
0	1	0	0.50
0	1	1	0.75
1	0	0	1.00
1	0	1	1.50
1	1	0	2.00
1	1	1	3.00

Bit	Name	Function
3	RESERVED	RESERVED
6-4	MEM_CAS_DLY_REG	CAS Delay Control bits: default value 3'b000; with DLY8LV

MEM_CAS_DLY_REG [2:0]			#OF NS
0	0	0	0.00
0	0	1	0.25
0	1	0	0.50
0	1	1	0.75
1	0	0	1.00
1	0	1	1.50
1	1	0	2.00
1	1	1	3.00

Bit	Name	Function
7	RESERVED	RESERVED

MEMORY CONTROLLER 26 REG S4_1A, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED	MEM_DQM_DLY_REG			RESERVED	MEM_WE_DLY_REG		

Bit	Name	Function
2-0	MEM_WE_DLY_REG	**WE Delay Control bits High 2 bits: with DLY8LV** **MEM_WE_DLY_REG [2:0] / #OF NS** 0 0 0 — 0.00 0 0 1 — 0.25 0 1 0 — 0.50 0 1 1 — 0.75 1 0 0 — 1.00 1 0 1 — 1.50 1 1 0 — 2.00 1 1 1 — 3.00
3	RESERVED	**RESERVED**
6-4	MEM_DQM_DLY_REG	**DQM Delay Control bits: with DLY8LV** MEM_DQM_DLY_REG [2:0] / #OF NS 0 0 0 — 0.00 0 0 1 — 0.25 0 1 0 — 0.50 0 1 1 — 0.75 1 0 0 — 1.00 1 0 1 — 1.50 1 1 0 — 2.00 1 1 1 — 3.00
7	RESERVED	**RESERVED**

MEMORY CONTROLLER 27 REG S4_1B, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED	MEM_CLK_DLY_REG			RESERVED	MEM_ADR_DLY_REG		

Bit	Name	Function
2-0	MEM_ADR_DLY_REG	**Address Delay Control bits: with DLY8LV** MEM_ADR_DLY_REG [2:0] / #OF NS 0 0 0 — 0.00 0 0 1 — 0.25 0 1 0 — 0.50 0 1 1 — 0.75 1 0 0 — 1.00 1 0 1 — 1.50 1 1 0 — 2.00 1 1 1 — 3.00
3	RESERVED	**RESERVED**
6-4	MEM_CLK_DLY_REG	**Clk of Rising Edge Delay Control bits: with DLY8LV** **MEM_CLK_DLY_REG [2:0]** / **#OF NS** 0 0 0 — 0.00 0 0 1 — 0.20 0 1 0 — 0.50 0 1 1 — 0.75 1 0 0 — 1.00 1 0 1 — 1.50 1 1 0 — 2.00 1 1 1 — 3.00
7	RESERVED	**RESERVED**

MEMORY CONTROLLER 28 REG S4_1C, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED	MEM_CS1_DLY_REG			RESERVED	MEM_CS0_DLY_REG		

Bit	Name	Function
2-0	MEM_CS0_DLY_REG	Chip Select 0 Delay Control Low 2bits: with DLY8LV

MEM_CS0_DLY_REG [2:0]			#OF NS
0	0	0	0.00
0	0	1	0.25
0	1	0	0.50
0	1	1	0.75
1	0	0	1.00
1	0	1	1.50
1	1	0	2.00
1	1	1	3.00

Bit	Name	Function
3	RESERVED	RESERVED
6-4	MEM_CS1_DLY_REG	Chip Select 1 Delay Control Low 2bits: with DLY8LV

MEM_CS1_DLY_REG [2:0]			#OF NS
0	0	0	0.00
0	0	1	0.25
0	1	0	0.50
0	1	1	0.75
1	0	0	1.00
1	0	1	1.50
1	1	0	2.00
1	1	1	3.00

Bit	Name	Function
7	RESERVED	RESERVED

MEMORY CONTROLLER 29 REG S4_1D, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED	MEM_BA1_DLY_REG			RESERVED	MEM_BA0_DLY_REG		

Bit	Name	Function
2-0	MEM_BA0_DLY_REG	**Bank0 Delay Control bits: with DLY8LV** MEM_BA0_DLY_REG [2:0] / #OF NS 0 0 0 — 0.00 0 0 1 — 0.25 0 1 0 — 0.50 0 1 1 — 0.75 1 0 0 — 1.00 1 0 1 — 1.50 1 1 0 — 2.00 1 1 1 — 3.00
3	RESERVED	**RESERVED**
6-4	MEM_BA1_DLY_REG	**Bank1 Delay Control bits: with DLY8LV** MEM_BA1_DLY_REG [2:0] / #OF NS 0 0 0 — 0.00 0 0 1 — 0.25 0 1 0 — 0.50 0 1 1 — 0.75 1 0 0 — 1.00 1 0 1 — 1.50 1 1 0 — 2.00 1 1 1 — 3.00
7	RESERVED	**RESERVED**

Chapter 06. CAPTURE & PLAYBACK REGISTERS

CAPTURE 00 REG S4_20, R/W

Bit	7	6	5	4	3	2	1	0
	CAP_NR_STATUS_OFFSET			RESERVED		CAP_CNTRL_TST		

Bit	Name	Function
2-0	CAP_CNTRL_TST	**Capture Test logic control:**
		Bit [2:0]: select capture internal test bus.
4-3	RESERVED	**RESERVED**
7:5	CAP_NR_STATUS_OFFSET	**Capture Noise Reduction Frame Status Offset**
		For NTSC and PAL, Noise Reduction will save 4 or 6 frame data, for Play back read which frame at first, set different value, will read different frame data firstly, default 0.

CAPTURE 01 REG S4_21, R/W

Bit	7	6	5	4	3	2	1	0
	CAP_ADR_ADD_2	CAP_VRST_FFRST_EN	CAP_SAFE_GUARD_EN	RESERVED	CAP_DOUBLE_BUFFER	CAP_BUF_STA_INV	CAP_FF_HALF_REQ	CAPTURE_ENABLE

Bit	Name	Function
0	CAPTURE_ENABLE	**Enable capture** When it's set 1, capture will be turn on. When it's set 0, capture will be turn off.
1	CAP_FF_HALF_REQ	**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_BUF_STA_INV	**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_DOUBLE_BUFFER	**Enable double buffer** When set to 1, enable double buffer. When set to 0, disable double buffer.
4	RESERVED	**RESERVED**
5	CAP_SAFE_GUARD_EN	**Enable safe guard function** When set to 1, turn on safe guard function. When set to 0, turn off safe guard function.
6	CAP_VRST_FFRST_EN	**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_ADR_ADD_2	**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 S4_22, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED				CAP_REQ_FREEZ	CAP_LAST_POP_CTL	CAP_STATUS_SEL	CAP_REQ_OVER

Bit	Name	Function
0	CAP_REQ_OVER	**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
1	CAP_STATUS_SEL	**Capture FIFO half status select** When set to 1, request generated when capture FIFO is half. When set to 0, request generated when capture FIFO is delm's value.
2	CAP_LAST_POP_CTL	**Capture POP data control** When set to 1, horizontal or vertical load start address will check if there is pop When set to 0, horizontal or vertical load start address will not check.
3	CAP_REQ_FREEZ	**Capture Request Freeze** When set to 1, capture FIFO will pause the FIFO write and read . When set to 0, capture FIFO will operate normally.
7-4	RESERVED	**RESERVED**

CAPTURE 03 REG S4_23, R/W

Bit	7	6	5	4	3	2	1	0
	CAP_FF_STATUS							

Bit	Name	Function
7-0	CAP_FF_STATUS	**Capture FIFO status**
		When cap_cntrl_[17] set 1'b1, this register will be valid, this value will less than 64.

CAPTURE 04 REG S4_24, R/W

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

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

CAPTURE 05 REG S4_25, R/W

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

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

CAPTURE 06 REG S4_26, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED			CAP_SAFE_GAURD_A [20:16]				

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

CAPTURE 07 REG S4_27, R/W

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

Bit	Name	Function
7-0	CAP_SAFE_GUARD_B [7:0]	**Safe Guard Address For Buffer B:** Safe guard address B [7:0]; Mapping to 32bits width data bus field.

CAPTURE 08 REG S4_28, R/W

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

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

CAPTURE 09 REG S4_29, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED			CAP_SAFE_GUARD_B [20:16]				

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

PLAY BACK 00 REG S4_2B, R/W

Bit	7	6	5	4	3	2	1	0
	PB_ENABLE	PB_2FRAME_EXCHG	PB_DB_BUFFER_EN	PB_DB_FIELD_EN	PB_BYPASS	PB_REQ_SEL		PB_CUT_REFRESH

Bit	Name	Function
0	PB_CUT_REFRESH	**Disable refresh request generation** When set to 1, disable refresh request generation. When set to 0, enable refresh request generation.
2-1	PB_REQ_SEL	**Enable playback request mode** PB_REQ_SEL / PBHREQ / PBLREQ 00 / 0 / Low request 01 / 0 / High request 10 / Low request / 0 11 / High request / Low request
3	PB_BYPASS	**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_DB_FIELD_EN	**Enable double field display** When set to 1, enable double field display. When set to 0, disable double field display.
5	PB_DB_BUFFER_EN	**Enable double buffer** When set to 1, enable double buffer. When set to 0, disable double buffer.
6	PB_2FRAME_EXCHG	**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_ENABLE	**Enable Playback** When it's set 1, play back will be on work, or will not work.

PLAY BACK 01 REG S4_2C, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED		PB_MAST_FLAG_REG					

Bit	Name	Function
5-0	PB_MAST_FLAG_REG	**Master line flag [5:0]** Playback FIFO policy master value: This field will define FIFO high request timing.
7-6	PB_CNTRL_[23:22]	**RESERVED** .

PLAY BACK 02 REG S4_2D, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED		PB_GENERAL_FLAG_REG					

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

PLAY BACK 03 REG S4_2E, R/W

Bit	7	6	5	4	3	2	1	0
	PB_DOUBLE_REFRESH_EN	RESERVED					PB_UP_DOW_RBUF_SEL	PB_UP_DOW_RBUF_INV

Bit	Name	Function
0	PB_UP_DOW_RBUF_INV	**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.
1	PB_UP_DOW_RBUF_SEL	**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.
6-2	RESERVED	**RESERVED**
7	PB_DOUBLE_REFRESH_EN	**Refresh Double** When set to 1, refresh request will at the rising and falling edge of hbout. When set to 0, refresh will be only at the rising edge of hbout.

PLAY BACK 04 REG S4_2F, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED				PB_TST_REG			

Bit	Name	Function
3-0	PB_TST_REG	**PlayBack Test Logic** To select playback test bus, total 8 groups can be selected.
7-4	RESERVED	**RESERVED**

CAPTURE AND PLAYBACK SHARED 00 REG S4_30, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED				PB_CAP_NOISE_CMD			

Bit	Name	Function
3-0	PB_CAP_NOISE_CMD	**Capture Noise Reduction Command** 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 [4] = 1 In playback**
7-4	RESERVED	**RESERVED**

CAPTURE AND PLAYBACK SHARED 01 REG S4_31, R/W

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

Bit	Name	Function
7-0	PB_CAP_BUF_STA_ADDR_A [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 02 REG S4_32, R/W

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

Bit	Name	Function
7-0	PB_CAP_BUF_STA_ADDR_A [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 03 — REG S4_33, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED			PB_CAP_BUF_STA_ADDR_A [20:16]				

Bit	Name	Function
4-0	PB_CAP_BUF_STA_ADDR_A [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	RESERVED	**RESERVED**

CAPTURE AND PLAYBACK SHARED 04 — REG S4_34, R/W

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

Bit	Name	Function
7-0	PB_CAP_BUF_STA_ADDR_B [7:0]	**Buffer B START address [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.

CAPTURE AND PLAYBACK SHARED 05 — REG S4_35, R/W

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

Bit	Name	Function
7-0	PB_CAP_BUF_STA_ADDR_B [15:8]	**Buffer B START address [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.

CAPTURE AND PLAYBACK SHARED 06 — REG S4_36, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED			PB_CAP_BUF_STA_ADDR_B [20:16]				

Bit	Name	Function
4-0	PB_CAP_BUF_STA_ADDR_B [20:16]	**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	RESERVED	**RESERVED**

CAPTURE AND PLAYBACK SHARED 07 — REG S4_37, R/W

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

Bit	Name	Function
7-0	PB_CAP_OFFSET [7:0]	**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 08 — REG S4_38, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED						PB_CAP_OFFSET [9:8]	

Bit	Name	Function
1-0	PB_CAP_OFFSET [9:8]	**Capture and Play Back Offset [9:8]:** Offset [9:8] will determine next line start address, mapping to 64 bits width data bus field.
7-2	RESERVED	**RESERVED**

CAPTURE AND PLAYBACK SHARED 09 — REG S4_39, R/W

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

Bit	Name	Function
7-0	PB_FETCH_NUM [7:0]	**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 10 — REG S4_3A, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED						PB_FETCH_NUM [9:8]	

Bit	Name	Function
1-0	PB_FETCH_NUM [9:8]	**Capture and Play Back Fetch Number [9:8]:** Fetch number [9:8] will determine to fetch the number of pixels from memory, Mapping to 64bits width data bus field.
7-2	RESERVED	**RESERVED**

CAPTURE AND PLAYBACK SHARED 11 — REG S4_3B, R/W

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

Bit	Name	Function
7-0	PB_CAP_BUF_STA_ADDR_C [7:0]	**Capture and playback Buffer C Start Address [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.

CAPTURE AND PLAYBACK SHARED 12 — REG S4_3C, R/W

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

Bit	Name	Function
7-0	PB_CAP_BUF_STA_ADDR_C [15:8]	**Capture and Play Back Buffer C Start Address [15:8]** 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.

CAPTURE AND PLAYBACK SHARED 13 REG S4_3D, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED			PB_CAP_BUF_STA_ADDR_C [20:16]				

Bit	Name	Function
4-0	PB_CAP_BUF_STA_ADDR_C [20:16]	**Capture and Play Back Buffer C Start Address [20:16]** 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.
7-5	RESERVED	**RESERVED**

CAPTURE AND PLAYBACK SHARED 14 REG S4_3E, R/W

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

Bit	Name	Function
7-0	PB_CAP_BUF_STA_ADDR_D [7:0]	**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 15 REG S4_3F, R/W

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

Bit	Name	Function
7-0	PB_CAP_BUF_STA_ADDR_D [15:8]	**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 16 REG S4_40, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED			PB_CAP_BUF_STA_ADDR_D [20:16]				

Bit	Name	Function
4-0	PB_CAP_BUF_STA_ADDR_D [20:16]	**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.
7-5	RESERVED	**RESERVED**

Chapter 07. WRITE & READ FIFO REGISTERS

WRITE FIFO 00 REG S4_41, R/W

Bit	7	6	5	4	3	2	1	0
	WFF_TST_REG							

Bit	Name	Function
7-0	WFF_TST_REG	**WRITE FIFO Test logic control:**
		BIT[7:0] : SELECT CAPTURE INTERNAL TEST BUS.

WRITE FIFO 01 REG S4_42, R/W

Bit	7	6	5	4	3	2	1	0
	WFF_FF_STATUS_SEL	WFF_REQ_OVER	WFF_ADR_ADD_2	WFF_VRST_FF_RST	WFF_SAFE_GUARD	WFF_FF_STA_INV	WFF_FF_HALF_REQ	WFF_ENABLE

Bit	Name	Function
0	WFF_ENABLE	**Enable write FIFO**
		When it's set 1, write FIFO will be turn on. When it's set 0, write FIFO will be turn off.
1	WFF_FF_HALF_REQ	**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_FF_STA_INV	**Write FIFO status invert**
		When set to 1, write FIFO status invert. When set to 0, write FIFO status don't change.
3	WFF_SAFE_GUARD	**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_VRST_FF_RST	**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_ADR_ADD_2	**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_REQ_OVER	**WRITE FIFO Horizontal Request End**
		When this bit set 1, the final write FIFO request of one line is in the horizontal blank rising edge, set 0 write FIFO request will free run
7	WFF_FF_STATUS_SEL	**WRITE FIFO HALF STATUS SELECT**
		When set to 1, request generated when FIFO is half. When set to 0, request generated when c FIFO is delm's value**.**

WRITE FIFO 02 REG S4_43, R/W

Bit	7	6	5	4	3	2	1	0
	WFF_FF_STATUS							

Bit	Name	Function
7-0	WFF_FF_STATUS	**Write FIFO status** When wff_cntrl_[15] set 1'b1, this register will be valid, this value will less than 64.

WRITE FIFO 03 REG S4_44, R/W

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

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

WRITE FIFO 04 REG S4_45, R/W

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

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

WRITE FIFO 05 REG S4_46, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED			WFF_SAFE_GUARD_A [20:16]				

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

WRITE FIFO 06 REG S4_47, R/W

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

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

WRITE FIFO 07 REG S4_48, R/W

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

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

WRITE FIFO 08 REG S4_49, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED			WFF_SAFE_GUARD_B [20:16]				

Bit	Name	Function
4-0	WFF_SAFE_GUARD_B [20:16]	**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	**RESERVED** .

WRITE FIFO 09 REG S4_4A, R/W

Bit	7	6	5	4	3	2	1	0
	WFF_LAST_POP_CTL	RESERVED		WFF_LINE_FLIP	RESERVED			WFF_YUV_DEINTERLACE

Bit	Name	Function
0	WFF_YUV_DEINTERLACE	**WRITE FIFO YUV DE-INTERLACE** When set 1, write FIFO will write one field YUV, set 0, will write one frame Y.
3-1	RESERVED	**RESERVED**
4	WFF_LINE_FLIP	**WRITE FIFO LINE INVERT:** When set 1, line id will be inverted; When set 0, line id will be normal.
6-5	RESERVED	RESERVED
7	WFF_LAST_POP_CTL	**WRITE FIFO POP data control** When set to 1, horizontal or vertical load start address will check if there is pop When set to 0, horizontal or vertical load start address will not check.

WRITE FIFO 10 REG S4_4B, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED	WFF_VB_DELAY			RESERVED	WFF_HB_DELAY		

Bit	Name	Function
2-0	WFF_HB_DELAY	**Write FIFO H-Timing Programmable Delay:**
3	RESERVED	**RESERVED**
6-4	WFF_VB_DELAY	**Write FIFO V-Timing Programmable Delay:**
7	RESERVED	**RESERVED**

READ FIFO 00 REG S4_4D, R/W

Bit	7	6	5	4	3	2	1	0
	RFF_ENABLE	RFF_REQ_SEL		RFF_ADR_ADD_2	RFF_NEW_PAGE			

Bit	Name	Function
3-0	RFF_NEW_PAGE	**Read buffer page select from 1 to 16**

RFF_NEW_PAGE	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

Bit	Name	Function
4	RFF_ADR_ADD_2	**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_REQ_SEL	**Enable read FIFO request mode**

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

Bit	Name	Function
7	RFF_ENABLE	**Enable Read FIFO** When set 1, read FIFO will be turned on; When set 0, read FIFO will be turned off.

READ FIFO 01 REG S4_4E, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED		RFF_MASTER_FLAG					

Bit	Name	Function
5-0	RFF_MASTER_FLAG	**Master line flag [5:0]** Read FIFO policy master value: This field will define FIFO high request timing.
7-6	RESERVED	**RESERVED**

READ FIFO 02 REG S4_4F, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED		RFF_GENERAL_FLAG					

Bit	Name	Function
5-0	RFF_GENERAL_FLAG	**General line flag [5:0]** Read FIFO policy master value: This field will define FIFO low request timing.
7-6	RESERVED	**RESERVED**

READ FIFO 03 REG S4_50, R/W

Bit	7	6	5	4	3	2	1	0
	RFF_LREQ_CUT	RFF_YUV_DEINTERLACE	RFF_LINE_FLIP	RESERVED	RFF_TST_REG			

Bit	Name	Function
3-0	RFF_TST_REG	**General Test Logic [3:0]** Read FIFO test bus select.
4	RESERVED	**RESERVED**
5	RFF_LINE_FLIP	**Line ID Invert** When set 1, line ID will be inverted; When set 0, line ID will be normal.
6	RFF_YUV_DEINTERLACE	**Read FIFO YUV De-interlace** When set 1, Read FIFO will read Frame 2 YUV data in line = 1, line =0, read Frame 1 YUV data. When set 0, Read FIFO will read Frame 2 Y data in line = 1, line =0 , read Frame 1 Y data.
7	RFF_LREQ_CUT	**READ FIFO LOW REQUEST CUT ENABLE** Cut the read FIFO low request, only output high request to memory

READ FIFO AND WRITE FIFO SHARED 00 REG S4_51, R/W

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

Bit	Name	Function
7-0	RFF_WFF_STA_ADDR_A [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 S4_52, R/W

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

Bit	Name	Function
7-0	RFF_WFF_STA_ADDR_A [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 S4_53, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED			RFF_WFF_STA_ADDR_A [20:16]				

Bit	Name	Function
4-0	RFF_WFF_STA_ADDR_A [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.
7-5	RESERVED	**RESERVED**

READ FIFO AND WRITE FIFO SHARED 03 REG S4_54, R/W

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

Bit	Name	Function
7-0	RFF_WFF_STA_ADDR_B [7:0]	**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 S4_55, R/W

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

Bit	Name	Function
7-0	RFF_WFF_STA_ADDR_B [15:8]	**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 S4_56, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED			RFF_WFF_STA_ADDR_B [20:16]				

Bit	Name	Function
4-0	RFF_WFF_STA_ADDR_B [20:16]	**Read FIFO AND Write FIFO START Address [20:16]** Start address buffer B [20:16], Mapping to 32 bits width data bus field.
7-5	RESERVED	**RESERVED**

READ FIFO AND WRITE FIFO SHARED 06 REG S4_57, R/W

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

Bit	Name	Function
7-0	RFF_WFF_OFFSET [7:0]	**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 S4_58, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED						RFF_WFF_OFFSET [9:8]	

Bit	Name	Function
1-0	RFF_WFF_OFFSET [9:8]	**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.
7-2	RESERVED	**RESERVED**

READ FIFO AND WRITE FIFO SHARED 08 REG S4_59, R/W

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

Bit	Name	Function
7-0	RFF_FETCH_NUM [7:0]	**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.

READ FIFO AND WRITE FIFO SHARED 09 REG S4_5A, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED						RFF_FETCH_NUM [9:8]	

Bit	Name	Function
1-0	RFF_FETCH_NUM [9:8]	**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.
7-2	RESERVED	**RESERVED**

READ FIFO AND WRITE FIFO SHARED 10 REG S4_5B, R/W

Bit	7	6	5	4	3	2	1	0
	MEM_FF_TOP_FF_SEL	RESERVED						

Bit	Name	Function
6-0	RESERVED	**RESERVED**
7	MEM_FF_TOP_FF_SEL	**All FIFO Status Output Enable** When set 1, all FIFO status output, can read FIFO status through test bus; When set 0, not FIFO status output.

Chapter 08. VIDEO PROCESSOR REGISTERS

VDS_PROC 00 REG S3_00, R/W

Bit	7	6	5	4	3	2	1	0
	VDS_SRESET	VDS_HALF_EN	VDS_VSCALE_BYPS	VDS_HSCALE_BYPS	VDS_FIELD_FLIP	VDS_DFIELD_EN	VDS_FIELDAB_EN	VDS_SYNC_EN

Bit	Name	Function
0	VDS_SYNC_EN	**External sync enable, active high** This bit enable sync lock mode. vds_flock_en (1A[4]) / vds_sync_en / VDS timing: 0 / 0 / Free run 0 / 1 / Sync lock 1 / X / Frame lock
1	VDS_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	VDS_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	VDS_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.
4	VDS_HSCALE_BYPS	**Horizontal scale up bypass control, active high** When this bit is 1, data will bypass horizontal scale up process.
5	VDS_VSCALE_BYPS	**Vertical scale up bypass control, active high** When this bit is 1, data will bypass vertical scale up process.
6	VDS_HALF_EN	**Horizontal scale up bypass control, active high**
7	VDS_SRESET	**Horizontal scale up bypass control, active high** When this bit is 1, it reset the VDS_PROC internal module ds_video_enhance,

VDS_PROC 01 REG S3_01, R/W

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

Bit	Name	Function
7-0	VDS_HSYNC_RST [7:0]	**Internal Horizontal period control bit[7:0], 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 02 REG S3_02, R/W

Bit	7	6	5	4	3	2	1	0
	VDS_VSYNC_RST [3:0]				VDS_HSYNC_RST [11:8]			

Bit	Name	Function
3-0	VDS_HSYNC_RST [11:8]	**Internal Horizontal period control bit[7:0], 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)
7-4	VDS_VSYNC_RST [3:0]	**Internal Vertical period control bit[3:0]** This field contains vertical total value minus 1.

VDS_PROC 03 REG S3_03, R/W

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

Bit	Name	Function
6-0	VDS_VSYNC_RST [10:4]	**Internal Vertical period control bit[10:4]** This field contains vertical total value minus 1.
7	RESERVED	

VDS_PROC 04 REG S3_04, R/W

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

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

VDS_PROC 05 REG S3_05, R/W

Bit	7	6	5	4	3	2	1	0
	VDS_HB_SP [3:0]				VDS_HB_ST [11:8]			

Bit	Name	Function
3-0	VDS_HB_ST [3:0]	**Horizontal blanking stop position control bit[11:8]** This field is used to program horizontal blanking start position, this blanking is used to get data from memory.
7-4	VDS_HB_SP [3:0]	**Horizontal blanking stop position control bit[3:0]** This field is used to program horizontal blanking stop position, this blanking is used to get data from memory.

VDS_PROC 06 REG S3_06, R/W

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

Bit	Name	Function
7-0	VDS_HB_SP [11:4]	**Horizontal blanking stop position control bit[3:0]** This field is used to program horizontal blanking stop position, this blanking is used to get data from memory.

VDS_PROC 07 REG S3_07, R/W

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

Bit	Name	Function
7-0	VDS_VB_ST [7:0]	**Vertical blanking start position control bit[7:0]** This field is used to program vertical blanking start position.

VDS_PROC 08 REG S3_08, R/W

Bit	7	6	5	4	3	2	1	0
	VDS_VB_SP [3:0]				RESERVED	VDS_VB_ST [10:8]		

Bit	Name	Function
2-0	VDS_VB_ST [10:8]	**Vertical blanking start position control bit[10:8]** This field is used to program vertical blanking start position.
3	RESERVED	
7-4	VDS_VB_SP [3:0]	**Vertical blanking stop position control bit[3:0]** This field is used to program vertical blanking stop position.

VDS_PROC 09 REG S3_09, R/W

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

Bit	Name	Function
6-0	VDS_VB_SP [10:4]	**Vertical blanking stop position control bit[10:4]** This field is used to program vertical blanking stop position.
7	RESERVED	

VDS_PROC 10 REG S3_0A, R/W

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

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

VDS_PROC 11 REG S3_0B, R/W

Bit	7	6	5	4	3	2	1	0
	VDS_HS_SP [3:0]				VDS_HS_ST [11:8]			

Bit	Name	Function
3-0	VDS_HS_ST [11:8]	**Horizontal sync start position control bit [7:0]** This field is used to program horizontal sync start position.
7-4	VDS_HS_SP [3:0]	**Horizontal sync stop position control bit [3:0]** This field is used to program horizontal sync stop position.

VDS_PROC 12 REG S3_0C, R/W

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

Bit	Name	Function
7-0	VDS_HS_SP [11:4]	**Horizontal sync stop position control bit [11:4]** This field is used to program horizontal sync stop position.

VDS_PROC 13 REG S3_0D, R/W

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

Bit	Name	Function
7-0	VDS_VS_ST [7:0]	**Vertical sync start position control bit [7:0]** This field is used to program vertical sync start position.

VDS_PROC 14 REG S3_0E, R/W

Bit	7	6	5	4	3	2	1	0
	VDS_VS_SP [3:0]				RESERVED	VDS_VS_ST [10:8]		

Bit	Name	Function
2-0	VDS_VS_ST [10:8]	**Vertical sync start position control bit [10:8]** This field is used to program vertical sync start position.
3	RESERVED	
7-4	VDS_VS_SP [3:0]	**Vertical sync stop position control bit [3:0]** This field is used to program vertical sync stop position.

VDS_PROC 15 REG S3_0F, R/W

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

Bit	Name	Function
6-0	VDS_VS_SP [10:4]	**Vertical sync stop position control bit [10:4]** This field is used to program vertical sync stop position.
7	RESERVED	

VDS_PROC 16 REG S3_10, R/W

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

Bit	Name	Function
7-0	VDS_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 17 REG S3_11, R/W

Bit	7	6	5	4	3	2	1	0
	VDS_DIS_HB_SP [3:0]				VDS_DIS_HB_ST [11:8]			

Bit	Name	Function
3-0	VDS_DIS_HB_ST [11:8]	**Final display horizontal blanking start position control bit [11:8]** This field contains final display horizontal blanking start position control, this blanking is used to clean the output data in blanking.
7-4	VDS_DIS_HB_SP [3:0]	**Final display horizontal blanking stop position control bit [3:0]** This field contains final display horizontal blanking stop position control, this blanking is used to clean the output data in blanking.

VDS_PROC 18 REG S3_12, R/W

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

Bit	Name	Function
7-0	VDS_DIS_HB_SP [11:4]	**Final display horizontal blanking stop position control bit [11:4]** This field contains final display horizontal blanking stop position control, this blanking is used to clean the output data in blanking.

VDS_PROC 19 REG S3_13, R/W

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

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

VDS_PROC 20 REG S3_14, R/W

Bit	7	6	5	4	3	2	1	0
	VDS_DIS_VB_SP [3:0]				RESERVED	VDS_DIS_VB_ST [10:8]		

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

VDS_PROC 21 REG S3_15, R/W

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

Bit	Name	Function
6-0	VDS_DIS_VB_SP [10:4]	**Final display vertical blanking stop position control bit [10:4]** This field contains final display vertical blanking stop position control, this blanking is used to clean the output data in blanking.
7	RESERVED	

VDS_PROC 22 REG S3_16, R/W

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

Bit	Name	Function
7-0	VDS_HSCALE [7:0]	**Horizontal scaling coefficient bit [7:0]** 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

VDS_PROC 23 REG S3_17, R/W

Bit	7	6	5	4	3	2	1	0
	VDS_VSCALE [3:0]				RESERVED		VDS_HSCALE [9:8]	

Bit	Name	Function
1-0	VDS_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
3-2	RESERVED	
7-4	VDS_VSCALE[3:0]	**Vertical scaling up coefficient bit [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

VDS_PROC 24 REG S3_18, R/W

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

Bit	Name	Function
6-0	VDS_VSCALE[9:4]	**Vertical scaling up coefficient bit [9:4]** 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
7	RESERVED	

VDS_PROC 25 REG S3_19, R/W

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

Bit	Name	Function
7-0	VDS_FRAME_RST [7:0]	**Frame reset period control bit [7:0]** 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 26 REG S3_1A, R/W

Bit	7	6	5	4	3	2	1	0
	VDS_FID_RST	VDS_FID_AA_DLY	VDS_FREERUN_FID	VDS_FLOCK_EN	RESERVED		VDS_FRAME_RST [9:8]	

Bit	Name	Function
1-0	VDS_FRAME_RST [9:8]	**Frame reset period control bit [9:8]** 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)
3-2	RESERVED	
4	VDS_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.
5	VDS_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.
6	VDS_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.
7	VDS_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.

VDS_PROC 27 REG S3_1B, R/W

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

Bit	Name	Function
7-0	VDS_FR_SELECT [7:0]	**Frame size select control bit [7:0]** FR_SELECT[2n+1:2n] is for frame n selection. 0 select VSYNC_RST; 1 select VSYNC_SIZE1; 2 select VSYNC_SIZE2.

VDS_PROC 28 REG S3_1C, R/W

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

Bit	Name	Function
7-0	VDS_FR_SELECT [15:8]	**Frame size select control bit [15:8]** FR_SELECT[2n+1:2n] is for frame n selection. 0 select VSYNC_RST; 1 select VSYNC_SIZE1; 2 select VSYNC_SIZE2.

VDS_PROC 29 REG S3_1D, R/W

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

Bit	Name	Function
7-0	VDS_FR_SELECT [23:16]	**Frame size select control bit [23:16]** FR_SELECT[2n+1:2n] is for frame n selection. 0 select VSYNC_RST; 1 select VSYNC_SIZE1; 2 select VSYNC_SIZE2.

VDS_PROC 30 REG S3_1E, R/W

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

Bit	Name	Function
7-0	VDS_FR_SELECT [31:24]	**Frame size select control bit [31:24]** FR_SELECT[2n+1:2n] is for frame n selection. 0 select VSYNC_RST; 1 select VSYNC_SIZE1; 2 select VSYNC_SIZE2.

VDS_PROC 31 REG S3_1F, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED		VDS_EN_FR_NUM_RST	VDS_DIF_FR_SEL_EN	VDS_FRAME_NO [3:0]			

Bit	Name	Function
3-0	VDS_FRAME_NO [3:0]	**Programmable repeat frame number control bit [3:0]** This field defines the repeated frame number, EX: if frame_no = 2, then the frame will repeat every 3 frame.
4	VDS_DIF_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.
5	VDS_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.
7-6	RESERVED	

VDS_FRAME_NO [3:0]				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

VDS_PROC 32 REG S3_20, R/W

Bit	7	6	5	4	3	2	1	0
	VDS_VSYN_SIZE1 [7:0]							

Bit	Name	Function
7-0	VDS_VSYN_SIZE1 [7:0]	**Programmable vertical total size 1 control bit [7:0]** 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.

VDS_PROC 33 REG S3_21, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED					VDS_VSYN_SIZE1 [10:8]		

Bit	Name	Function
2-0	VDS_VSYN_SIZE1 [10:8]	**Programmable vertical total size 1 control bit [10:8]** 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.
7-3	RESERVED	

VDS_PROC 34 REG S3_22, R/W

Bit	7	6	5	4	3	2	1	0
	VDS_VSYN_SIZE2 [7:0]							

Bit	Name	Function
7-0	VDS_VSYN_SIZE2 [7:0]	**Programmable vertical total size 2 control bit [7:0]** 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.

VDS_PROC 35 REG S3_23, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED					VDS_VSYN_SIZE2 [10:8]		

Bit	Name	Function
2-0	VDS_VSYN_SIZE2 [10:8]	**Programmable vertical total size 2 control bit [10:8]** 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.
7-4	RESERVED	

VDS_PROC 36 REG S3_24, R/W

Bit	7	6	5	4	3	2	1	0
	VDS_WEN_DELAY		VDS_Y_DELAY		VDS_TAP6_BYPS	VDS_V_DELAY	VDS_U_DELAY	VDS_UV_FLIP

Bit	Name	Function
0	VDS_UV_FLIP	**422 to 444 conversion UV flip control** This bit is used to flip UV, when this bit is 1, UV position will be flipped.
1	VDS_U_DELAY	**UV 422 to 444 conversion U delay** When this bit is 1, U will delay 1 clock, otherwise, no delay for internal pipe.
2	VDS_V_DELAY	**UV 422 to 444 conversion V delay** When this bit is 1, V will delay 1 clock, otherwise, no delay for internal pipe.
3	VDS_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
5-4	VDS_Y_DELAY	**Y compensation 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.
7-6	VDS_WEN_DELAY	**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.

VDS_Y_DELAY [1:0]		Y delay
0	0	1
0	1	2
1	0	3
1	1	4

VDS_WEN_DELAY [1:0]		Delay (VCLK)
0	0	1
0	1	2
1	0	3
1	1	4

VDS_PROC 37 REG S3_25, R/W

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

Bit	Name	Function
7-0	VDS_D_SP [7:0]	**Line buffer write reset position control bit [7:0]** This field contains the write reset position of the line buffer, this position is also the write start position of the buffer.

VDS_PROC 38 REG S3_26, R/W

Bit	7	6	5	4	3	2	1	0
	VDS_BLEV_AUTO_EN	VDS_D_RAM_BYPS	RESERVED				VDS_D_SP [9:8]	

Bit	Name	Function
1-0	VDS_D_SP [9:8]	**Line buffer write reset position control bit [7:0]** This field contains the write reset position of the line buffer, this position is also the write start position of the buffer.
5-2	RESERVED	
6	VDS_D_RAM_BYPS	**Line buffer one line delay data bypass, active high** When this bit is 1, data will bypass the line buffer.
7	VDS_BLEV_AUTO_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.

VDS_PROC 39 REG S3_27, R/W

Bit	7	6	5	4	3	2	1	0
	VDS_USER_MAX				VDS_USER_MIN			

Bit	Name	Function
3-0	VDS_USER_MIN	**Programmable minimum value control bit [3:0]** This field is the user defined min value for black level expansion, the actual min value in use is 2*blev_det_min+1.
7-4	VDS_USER_MAX	**Programmable maximum value control bit [3:0]** This field is the user defined max value for black level expansion, the actual min value in use is 16*blev_det_max+15.

VDS_PROC 40 REG S3_28, R/W

Bit	7	6	5	4	3	2	1	0
	VDS_BLEV_LEVEL							

Bit	Name	Function
7-0	VDS_BLEV_LEVEL	**Black level expansion level control bit [7:0]** This field defines the black level expansion threshold level value, data larger than this level will have no black level expansion process.

VDS_PROC 41 REG S3_29, R/W

Bit	7	6	5	4	3	2	1	0
	VDS_BLEV_GAIN							

Bit	Name	Function
7-0	VDS_BLEV_GAIN	**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 42 REG S3_2A, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED		VDS_STEP_DLY_CNTRL		RESERVED			VDS_BLEV_BYPS

Bit	Name	Function
0	VDS_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.
3-1	RESERVED	
5-4	VDS_STEP_DLY_CNTRL	**UV step response data select control bit [1:0]** VDS_STEP_DLY_CNTRL [1:0] / 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.
7-6	RESERVED	

VDS_PROC 43 REG S3_2B, R/W

Bit	7	6	5	4	3	2	1	0
	VDS_UV_STEP_BYPS	VDS_STEP_CLIP			VDS_STEP_GAIN			

Bit	Name	Function
3-0	VDS_STEP_GAIN	**UV 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.
6-4	VDS_STEP_CLIP	**UV step response clip control bit [2:0]** This filed contains the clip control value of UV step response
7	VDS_UV_STEP_BYPS	**UV step response bypass control, active high** When this bit is 1, UV data will don't do step response

VDS_PROC 44 REG S3_2C, R/W

Bit	7	6	5	4	3	2	1	0
	VDS_SK_U_CENTER							

Bit	Name	Function
7-0	VDS_SK_U_CENTER	**Skin color correction U center position control bit [7:0]** This field contains the skin color center position U value, the value is 2's.

VDS_PROC 45 REG S3_2D, R/W

Bit	7	6	5	4	3	2	1	0
	VDS_SK_V_CENTER							

Bit	Name	Function
7-0	VDS_SK_V_CENTER	**Skin color correction V center position control bit [7:0]** This field contains the skin color center position U value, the value is 2's.

VDS_PROC 46 REG S3_2E, R/W

Bit	7	6	5	4	3	2	1	0
	VDS_SK_Y_LOW_TH							

Bit	Name	Function
7-0	VDS_SK_Y_LOW_TH	**Skin color correction Y low threshold control bit [7:0]** Y low threshold value for skin color correction, if y less than this threshold, no skin color correction done.

VDS_PROC 47 REG S3_2F, R/W

Bit	7	6	5	4	3	2	1	0
	VDS_SK_Y_HIGH_TH							

Bit	Name	Function
7-0	VDS_SK_Y_HIGH_TH	**Skin color correction Y high threshold control bit [7:0]** Y high threshold value for skin color correction, if y larger than this threshold, no skin color correction done.

VDS_PROC 48 REG S3_30, R/W

Bit	7	6	5	4	3	2	1	0
	VDS_SK_RANGE							

Bit	Name	Function
7-0	VDS_SK_RANGE	**Skin color correction range control bit [7:0]** The skin color correction will done just when the value abs(u-u_center)+abs(v-v_enter) less than this programmable range.

VDS_PROC 49 REG S3_31, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED		VDS_SK_BYPS	VDS_SK_Y_EN	VDS_SK_GAIN			

Bit	Name	Function
3-0	VDS_SK_GAIN	**Skin color correction gain control bit [3:0]** This register defines the degree of the skin color correction, the higher the value, the more skin color correction done. Its range is (0~1)*16
4	VDS_SK_Y_EN	**Skin color Y detect enable, active high** When this bit is 1, take the Y value as the condition of skin color correction, just when the Y value larger than y_low_th and less the y_high_th, the correction can be done.
5	VDS_SK_BYPS	**Skin color correction bypass control, active high** When this bit is 1, the skin color correction will be bypassed.
7-6	RESERVED	

VDS_PROC 50 REG S3_32, R/W

Bit	7	6	5	4	3	2	1	0
	VDS_SVM_SIGMOID_BYPS	VDS_SVM_VCLK_DELAY			VDS_SVM_2ND_BYPS	VDS_SVM_POL_FLIP	VDS_SVM_BPF_CNTRL	

Bit	Name	Function
1-0	VDS_SVM_BPF_CNTRL	**SVM data generation select control [1:0]** VDS_SVM_BPF_CNTRL [1:0] / 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.
2	VDS_SVM_POL_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.
3	VDS_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.
6-4	VDS_SVM_VCLK_DELAY	**To match YUV pipe, SVM data delay by VCLK control bit [2:0]** This field define the SVM compensation delay from 1 to 8 VCLKs
7	VDS_SVM_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.

VDS_PROC 51 REG S3_33, R/W

Bit	7	6	5	4	3	2	1	0
	VDS_SVM_GAIN							

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

VDS_PROC 52 REG S3_34, R/W

Bit	7	6	5	4	3	2	1	0
	VDS_SVM_OFFSET							

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

VDS_PROC 53 REG S3_35, R/W

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

Bit	Name	Function
7-0	VDS_Y_GAIN [7:0]	**Y dynamic range expansion gain control bit [7:0]** This field contains the Y gain value in dynamic range expansion process, its range is (0 ~ 2)*128.

VDS_PROC 54 REG S3_36, R/W

Bit	7	6	5	4	3	2	1	0
	VDS_UCOS_GAIN							

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

VDS_PROC 55 REG S3_37, R/W

Bit	7	6	5	4	3	2	1	0
	VDS_VCOS_GAIN							

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

VDS_PROC 56 REG S3_38, R/W

Bit	7	6	5	4	3	2	1	0
	VDS_USIN_GAIN							

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

VDS_PROC 57 REG S3_39, R/W

Bit	7	6	5	4	3	2	1	0
	VDS_VSIN_GAIN							

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

VDS_PROC 58 REG S3_3A, R/W

Bit	7	6	5	4	3	2	1	0
	VDS_Y_OFST							

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

VDS_PROC 59 REG S3_3B, R/W

Bit	7	6	5	4	3	2	1	0
	VDS_U_OFST							

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

VDS_PROC 60 REG S3_3C, R/W

Bit	7	6	5	4	3	2	1	0
	VDS_V_OFST							

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

VDS_PROC 61 REG S3_3D, R/W

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

Bit	Name	Function
7-0	VDS_SYNC_LEV [7:0]	**Sync level bit [7:0]** 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 62 REG S3_3E, R/W

Bit	7	6	5	4	3	2	1	0
	VDS_BLK_BF_EN	RESERVED		VDS_DYN_BYPS	VDS_CONVT_BYPS	RESERVED		VDS_SYNC_LEV [8]

Bit	Name	Function
0	VDS_SYNC_LEV [8]	**Sync level bit [8]** 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.
2-1	RESERVED	
3	VDS_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	VDS_DYN_BYPS	**Dynamic range expansion bypass control, active high** When this bit is 1, data will bypass the dynamic range expansion process.
6-5	RESERVED	
7	VDS_BLK_BF_EN	**Blanking set up enable, active high** When this bit is 1, final composite blank (dis_hb|dis_vb) will cut the garbage data in blanking interval.

VDS_PROC 63 REG S3_3F, R/W

Bit	7	6	5	4	3	2	1	0
	VDS_UV_BLK_VAL							

Bit	Name	Function
7-0	VDS_UV_BLK_VAL	**UV blanking amplitude value control bit[7:0]** This filed indicates the amplitude value of UV in blanking interval, the highest bit of this programmable register is sign bit.

VDS_PROC 64 REG S3_40, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED		VDS_SVM_V4CLK_DELAY		RESERVED	VDS_IN_DREG_BYPS	VDS_2ND_INT_BYPS	VDS_1ST_INT_BYPS

Bit	Name	Function
0	VDS_1ST_INT_BYPS	**The 1st stage interpolation bypass control, active high** When this bit is 1, the 1st stage interpolation (in YUV domain) will be bypassed, Y use tap19, and UV use tap7.
1	VDS_2ND_INT_BYPS	**The 2nd stage interpolation bypass control, active high** When this bit is 1, the 2nd stage interpolation (in RGB domain) will be bypassed, all RGB use tap11.
2	VDS_IN_DREG_BYPS	**Input data bypass the negedge trigger 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.
3	RESERVED	
5-4	VDS_SVM_V4CLK_DELAY	**SVM delay be V2CLK control bit [1:0]** This field define the SVM delay from 1 to 4 V2CLKs
7-6	RESERVED	

VDS_SVM_V4CLK_DELAY		SVM delay
0	0	1
0	1	2
1	0	3
1	1	4

VDS_PROC 65 REG S3_41, R/W

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

Bit	Name	Function
7-0	VDS_PK_LINE_BUF_SP [7:0]	**Line buffer for 2D peaking write reset position control bit [7:0]** This field contains the write reset position of the line buffer, this position is also the write start position of the buffer.

VDS_PROC 66 REG S3_42, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED	VDS_PK_RAM_BYPS	RESERVED				VDS_PK_LINE_BUF_SP [9:8]	

Bit	Name	Function
1-0	VDS_PK_LINE_BUF_SP [9:8]	**Line buffer for 2D peaking write reset position control bit [9:8]** This field contains the write reset position of the line buffer, this position is also the write start position of the buffer.
5-2	RESERVED	
6	VDS_PK_RAM_BYPS	**Line buffer for 2D peaking one line delay data bypass, active high** When this bit is 1, data will bypass the line buffer.
7	RESERVED	

VDS_PROC 67 REG S3_43, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED				VDS_PK_VH_HH_SEL	VDS_PK_VH_HL_SEL	VDS_PK_VL_HH_SEL	VDS_PK_VL_HL_SEL

Bit	Name	Function
0	VDS_PK_VL_HL_SEL	**2D peaking vertical low-pass signal select the horizontal split filter control** low-pass filter select, 1 for tap3 and 0 for tap5
1	VDS_PK_VL_HH_SEL	**2D peaking vertical low-pass signal select the horizontal split filter control** for high-pass filter select, 1 for tap3 and 0 for tap5
2	VDS_PK_VH_HL_SEL	**2D peaking vertical high-pass signal select the horizontal split filter control** high-pass filter select, 1 for tap3 and 0 for tap5.
3	VDS_PK_VH_HH_SEL	**2D peaking vertical high-pass signal select the horizontal split filter control** low-pass filter select, 1 for tap3 and 0 for tap5
7-4	RESERVED	

VDS_PROC 68 REG S3_44, R/W

Bit	7	6	5	4	3	2	1	0
	VDS_PK_LB_CMP					VDS_PK_LB_CORE		

Bit	Name	Function
2-0	VDS_PK_LB_CORE	**2D peaking vertical low-pass horizontal band-pass signal coring level** Vertical low-pass and horizontal band-pass signal larger than this coring level will remain unchanged, otherwise it will be cut to 0.
7-3	VDS_PK_LB_CMP	**2D peaking vertical low-pass horizontal band-pass signal threshold level** Vertical low-pass and horizontal band-pass signal larger than this coring level will remain unchanged, otherwise the gain will added on it.

VDS_PROC 69 REG S3_45, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED		VDS_PK_LB_GAIN					

Bit	Name	Function
5-0	VDS_PK_LB_GAIN	**2D peaking vertical low-pass horizontal band-pass signal gain control** Vertical low-pass horizontal band-pass signal gain, its range is (0~4)*16.
7-6	RESERVED	

VDS_PROC 70 REG S3_46, R/W

Bit	7	6	5	4	3	2	1	0
	VDS_PK_LH_CMP					VDS_PK_LH_CORE		

Bit	Name	Function
2-0	VDS_PK_LH_CORE	**2D peaking vertical low-pass horizontal high-pass signal coring level** Vertical low-pass and horizontal high-pass signal larger than this coring level will remain unchanged, otherwise it will be cut to 0.
7-3	VDS_PK_LH_CMP	**2D peaking vertical low-pass horizontal high-pass signal threshold level** Vertical low-pass and horizontal high-pass signal larger than this coring level will remain unchanged, otherwise the gain will added on it.

VDS_PROC 71 REG S3_47, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED		VDS_PK_LH_GAIN					

Bit	Name	Function
5-0	VDS_PK_LH_GAIN	**2D peaking vertical low-pass horizontal high-pass signal gain control** Vertical low-pass horizontal high-pass signal gain, its range is (0~4)*16.
7-6	RESERVED	

VDS_PROC 72 REG S3_48, R/W

Bit	7	6	5	4	3	2	1	0
	VDS_PK_HL_CMP					VDS_PK_HL_CORE		

Bit	Name	Function
2-0	VDS_PK_HL_CORE	**2D peaking vertical high-pass horizontal low-pass signal coring level** Vertical high-pass and horizontal low-pass signal larger than this coring level will remain unchanged, otherwise it will be cut to 0.
7-3	VDS_PK_HL_CMP	**2D peaking vertical high-pass horizontal low-pass signal threshold level** Vertical high-pass and horizontal low-pass signal larger than this coring level will remain unchanged, otherwise the gain will added on it.

VDS_PROC 73 REG S3_49, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED		VDS_PK_HL_GAIN					

Bit	Name	Function
5-0	VDS_PK_HL_GAIN	**2D peaking vertical high-pass horizontal low-pass signal gain control** Vertical high-pass horizontal low-pass signal gain, its range is (0~4)*16.
7-6	RESERVED	

VDS_PROC 74 REG S3_4A, R/W

Bit	7	6	5	4	3	2	1	0
	VDS_PK_HB_CMP					VDS_PK_HB_CORE		

Bit	Name	Function
2-0	VDS_PK_HB_CORE	**2D peaking vertical high-pass horizontal band-pass signal coring level** Vertical high-pass and horizontal band-pass signal larger than this coring level will remain unchanged, otherwise it will be cut to 0.
7-3	VDS_PK_HB_CMP	**2D peaking vertical high-pass horizontal band-pass signal threshold level** Vertical high-pass and horizontal band-pass signal larger than this coring level will remain unchanged, otherwise the gain will added on it.

VDS_PROC 75 REG S3_4B, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED		VDS_PK_HB_GAIN					

Bit	Name	Function
5-0	VDS_PK_HB_GAIN	**2D peaking vertical high-pass horizontal band-pass signal gain control** Vertical high-pass horizontal band-pass signal gain, its range is (0~4)*16.
7-6	RESERVED	

VDS_PROC 76 REG S3_4C, R/W

Bit	7	6	5	4	3	2	1	0
	VDS_PK_HH_CMP					VDS_PK_HH_CORE		

Bit	Name	Function
2-0	VDS_PK_HH_CORE	**2D peaking vertical high-pass horizontal high-pass signal coring level** Vertical high-pass and horizontal high-pass signal larger than this coring level will remain unchanged, otherwise it will be cut to 0.
7-3	VDS_PK_HH_CMP	**2D peaking vertical high-pass horizontal high-pass signal threshold level** Vertical high-pass and horizontal high-pass signal larger than this coring level will remain unchanged, otherwise the gain will added on it.

VDS_PROC 77 REG S3_4D, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED		VDS_PK_HH_GAIN					

Bit	Name	Function
5-0	VDS_PK_HH_GAIN	**2D peaking vertical high-pass horizontal high-pass signal gain control** Vertical high-pass horizontal high-pass signal gain, its range is (0~4)*16.
7-6	RESERVED	

VDS_PROC 78 REG S3_4E, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED	VDS_C_VPK_CORE			VDS_C_VPK_BYPS	RESERVED	VDS_PK_Y_V_BYPS	VDS_PK_Y_H_BYPS

Bit	Name	Function
0	VDS_PK_Y_H_BYPS	**Y horizontal peaking bypass control, active high** When this bit is 1, Y horizontal peaking will be bypassed.
1	VDS_PK_Y_V_BYPS	**Y vertical peaking bypass control, active high** When this bit is 1, Y vertical peaking will be bypassed.
2	RESERVED	
3	VDS_C_VPK_BYPS	**UV vertical peaking bypass control, active high** When this bit is 1, UV vertical peaking will be bypassed.
6-4	VDS_C_VPK_CORE	**UV vertical peaking coring level** UV vertical high-pass signal larger than this coring level will remain unchanged, otherwise it will be cut to 0.
7	RESERVED	

VDS_PROC 79 REG S3_4F, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED		VDS_C_VPK_GAIN					

Bit	Name	Function
5-0	VDS_C_VPK_GAIN	**UV vertical peaking gain control bit [5:0]** UV vertical high-pass signal gain control, its range is (0~4)*16.
7-6	RESERVED	

VDS_PROC 80 REG S3_50, R/W

Bit	7	6	5	4	3	2	1	0
	VDS_DO_16B_EN	VDS_DO_UVSEL_FLIP	VDS_DO_UV_DEC_BYPS	VDS_TEST_EN	VDS_TEST_BUS_SEL			

Bit	Name	Function
3-0	VDS_TEST_BUS_SEL	**Test out select control bit [3:0]** This register is used to select internal status bus to test bus.
4	VDS_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.
5	VDS_DO_UV_DEC_BYPS	**16-bit digital out UV decimation filter bypass control, active high** When this bit is 1, 16-bit 422 YUV digital out UV decimation will be bypassed.
6	VDS_DO_UVSEL_FLIP	**16-bit digital out UV flip control** When this bit is 1, 16-bit 422 YUV digital out UV position will be flipped.
7	VDS_DO_16B_EN	**16-bit digital out (422 format yuv) enable** When this bit is 1, digital out is 16-bit 422 YUV format; When it is 0, digital out is 24-bit.

VDS_PROC 81 REG S3_51, R/W

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

Bit	Name	Function
7-0	VDS_GLB_NOISE [7:0]	**Global still detection threshold value control bit [7:0]** This field contains the global noise threshold value. If the total difference of two frame less than this programmable value, the picture is taken as still, otherwise, the picture is taken as moving picture.

VDS_PROC 82 REG S3_52, R/W

Bit	7	6	5	4	3	2	1	0
	VDS_NR_MI_TH_EN	VDS_NR_DIF_LPF5_BYPS	VDS_NR_C_BYPASS	VDS_NR_Y_BYPASS	RESERVED	VDS_GLB_NOISE [10:8]		

Bit	Name	Function
2-0	VDS_GLB_NOISE [10:8]	**Global still detection threshold value control bit [10:8]** This field contains the global noise threshold value. If the total difference of two frame less than this programmable value, the picture is taken as still, otherwise, the picture is taken as moving picture.
3	RESERVED	
4	VDS_NR_Y_BYPASS	**Y bypass the noise reduction process control** When this bit is 1, Y data will bypass the noise reduction process.
5	VDS_NR_C_BYPASS	**UV bypass the noise reduction process control** When this bit is 1, UV data will bypass the noise reduction process.
6	VDS_NR_DIF_LPF5_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	VDS_NR_MI_TH_EN	**Noise reduction threshold control enable** This bit will enable the threshold control, active high.

VDS_PROC 83 REG S3_53, R/W

Bit	7	6	5	4	3	2	1	0
	VDS_NR_MIG_USER_EN	VDS_NR_MI_OFFSET						

Bit	Name	Function
6-0	VDS_NR_MI_OFFSET	**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.
7	VDS_NR_MIG_USER_EN	**Motion index generation user mode enable** When this bit is 1, the motion index generation will use nr_mig_offt[3:0] as Motion index.

VDS_PROC 84 REG S3_54, R/W

Bit	7	6	5	4	3	2	1	0
	VDS_NR_STILL_GAIN				VDS_NR_MI_GAIN			

Bit	Name	Function
3-0	VDS_NR_MI_GAIN	**Motion index generation gain control bit [3:0]** Motion index generation gain control, its range is (0~8)*2.
7-4	VDS_NR_STILL_GAIN	**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.

VDS_PROC 85 REG S3_55, R/W

Bit	7	6	5	4	3	2	1	0
	VDS_NR_GLB_STILL_MENU	VDS_NR_EN_GLB_STILL	RESERVED	VDS_NR_EN_H_NOISY	VDS_NR_MI_THRESH			

Bit	Name	Function
3-0	VDS_NR_MI_THRESH	**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	VDS_NR_EN_H_NOISY	**High noisy picture index enable, active high** Enable high noisy index from de-interlacer, it means the picture's noise is very large.
5	RESERVED	
6	VDS_NR_EN_GLB_STILL	**Global still index enable, active high** This bit enables the global still signal.
7	VDS_NR_GLB_STILL_MENU	**Menu mode control for global still index (used for debug)** 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.

VDS_NR_GLB_STILL_MENU	VDS_NR_EN_GLB_STILL	Sub-block still index
0	0	0
0	1	glb_still
1	x	1

VDS_PROC 86 REG S3_56, R/W

Bit	7	6	5	4	3	2	1	0
	VDS_W_LEV_BYPS	VDS_NR_NOISY_OFFSET						

Bit	Name	Function
6-0	VDS_NR_NOISY_OFFSET	**Motion index generation offset control bit [6:0] for high noisy picture** When the picture is high noisy picture, this field contains the offset control for motion index generation.
7	VDS_W_LEV_BYPS	**White level expansion bypass control, active high** When this bit is 1, Y don't do white level expansion.

VDS_PROC 87 REG S3_57, R/W

Bit	7	6	5	4	3	2	1	0
	VDS_W_LEV							

Bit	Name	Function
7-0	VDS_W_LEV	**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.

VDS_PROC 88 REG S3_58, R/W

Bit	7	6	5	4	3	2	1	0
	VDS_WLEV_GAIN							

Bit	Name	Function
7-0	VDS_WLEV_GAIN	**White level expansion gain 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.

VDS_PROC 89 REG S3_59, R/W

Bit	7	6	5	4	3	2	1	0
	VDS_NS_U_CENTER							

Bit	Name	Function
7-0	VDS_NS_U_CENTER	**Non-linear saturation center position U value control bit [7:0]** This field contains the non-linear saturation center position U value, the value is 2's.

VDS_PROC 90 REG S3_5A, R/W

Bit	7	6	5	4	3	2	1	0
	VDS_NS_V_CENTER							

Bit	Name	Function
7-0	VDS_NS_V_CENTER	**Non-linear saturation center position V value control bit [7:0]** This field contains the non-linear saturation center position V value, the value is 2's.

VDS_PROC 91 REG S3_5B, R/W

Bit	7	6	5	4	3	2	1	0
	VDS_NS_SQUARE_RAD [0]	VDS_NS_U_GAIN						

Bit	Name	Function
6-0	VDS_NS_U_GAIN	**Non-linear saturation U gain control bit [6:0]** This field contains the U gain control for U component in the area which should do non-linear saturation, its range is (0~1)*128.
7	VDS_NS_SQUARE_RAD [0]	**Non-linear saturation range control bit [0]** Non-linear saturation only did When (u-u_center)^2 + (v-v_center)^2 less than this programmable range value.

VDS_PROC 92 REG S3_5C, R/W

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

Bit	Name	Function
7-0	VDS_NS_SQUARE_RAD [8:1]	**Non-linear saturation range control bit [8:1]** Non-linear saturation only did When (u-u_center)^2 + (v-v_center)^2 less than this programmable range value.

VDS_PROC 93 REG S3_5D, R/W

Bit	7	6	5	4	3	2	1	0
	VDS_NS_Y_HIGH_TH [1:0]		VDS_NS_SQUARE_RAD [14:9]					

Bit	Name	Function
5-0	VDS_NS_SQUARE_RAD [14:9]	**Non-linear saturation range control bit [14:9]** Non-linear saturation only did When (u-u_center)^2 + (v-v_center)^2 less than this programmable range value.
7-6	VDS_NS_Y_HIGH_TH [1:0]	**Non-linear saturation Y high threshold control bit [1:0]** This filed defines the Y high threshold value for non-linear saturation, when y detect enable (60[3]=1), if y larger than this programmable value, no non-linear did.

VDS_PROC 94 REG S3_5E, R/W

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

Bit	Name	Function
5-0	VDS_NS_Y_HIGH_TH [7:2]	**Non-linear saturation Y high threshold control bit [7:2]** This filed defines the Y high threshold value for non-linear saturation, when y detect enable (60[3]=1), if y larger than this programmable value, no non-linear did.
7-6	VDS_NS_V_GAIN [1:0]	**Non-linear saturation V gain control bit [1:0]** This field contains the V gain control for V component in the area which should do non-linear saturation, its range is (0~1)*128.

VDS_PROC 95 REG S3_5F, R/W

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

Bit	Name	Function
4-0	VDS_NS_V_GAIN [6:2]	**Non-linear saturation V gain control bit [6:2]** This field contains the V gain control for V component in the area which should do non-linear saturation, its range is (0~1)*128.
7-5	VDS_NS_Y_LOW_TH [2:0]	**Non-linear saturation Y low threshold control bit [2:0]** This filed defines the Y low threshold value for non-linear saturation, when y detect enable (60[3]=1), if y less than this programmable value, no non-linear did.

VDS_PROC 96 REG S3_60, R/W

Bit	7	6	5	4	3	2	1	0
	VDS_C1_TAG_LOW_SLOPE [3:0]				VDS_NS_Y_ACTIVE_EN	VDS_NS_BYPS	VDS_NS_Y_LOW_TH [4:3]	

Bit	Name	Function
1-0	VDS_NS_Y_LOW_TH [4:3]	**Non-linear saturation Y low threshold control bit [4:3]** This filed defines the Y low threshold value for non-linear saturation, when y detect enable (60[3]=1), if y less than this programmable value, no non-linear did.
2	VDS_NS_BYPS	**Non-linear saturation bypass control, active high** When this bit is 1, the process non-linear saturation will be bypassed.
3	VDS_NS_Y_ACTIVE_EN	**Non-linear saturation Y detect enable, active high** When this bit is 1, the process non-linear saturation only done when the Y larger than the value ns_y_low_th and less than the value ns_y_high_th.
7-4	VDS_C1_TAG_LOW_SLOPE [3:0]	**Red enhance angle tan value low threshold value control bit [3:0]** This filed contains the low threshold value for red enhance angle tan value, when the input UV angle tan value less than this programmable value, no enhancement did.

VDS_PROC 97 REG S3_61, R/W

Bit	7	6	5	4	3	2	1	0
	VDS_C1_TAG_HIGH_SLOPE [1:0]		VDS_C1_TAG_LOW_SLOPE [9:4]					

Bit	Name	Function
5-0	VDS_C1_TAG_LOW_SLOPE [9:4]	**Red enhance angle tan value low threshold value control bit [9:4]** This filed contains the low threshold value for red enhance angle tan value, when the input UV angle tan value less than this programmable value, no enhancement did.
7-6	VDS_C1_TAG_HIGH_SLOPE [1:0]	**Red enhance angle tan value high threshold value control bit [1:0]** This filed contains the high threshold value for red enhance angle tan value, when the input UV angle tan value larger than this programmable value, no enhancement did.

VDS_PROC 98 REG S3_62, R/W

Bit	7	6	5	4	3	2	1	0
	VDS_C1_TAG_HIGH_SLOPE [9:2]							

Bit	Name	Function
7-0	VDS_C1_TAG_HIGH_SLOPE [9:2]	**Red enhance angle tan value high threshold value control bit [9:2]** This filed contains the high threshold value for red enhance angle tan value, when the input UV angle tan value larger than this programmable value, no enhancement did.

VDS_PROC 99 REG S3_63, R/W

Bit	7	6	5	4	3	2	1	0
	VDS_C1_U_LOW [3:0]				VDS_C1_GAIN			

Bit	Name	Function
3-0	VDS_C1_GAIN	**Red enhance gain control bit [3:0]** This field contains the gain control for red enhance, its range is (0~1)*16
7-4	VDS_C1_U_LOW [3:0]	**Red enhance U low threshold value control bit [3:0]** This field contains the low threshold value for U component, if input U less then this programmable value, no enhancement did.

VDS_PROC 100 REG S3_64, R/W

Bit	7	6	5	4	3	2	1	0
	VDS_C1_U_HIGH [3:0]				VDS_C1_U_LOW [7:4]			

Bit	Name	Function
3-0	VDS_C1_U_LOW [7:4]	**Red enhance U low threshold value control bit [7:4]** This field contains the low threshold value for U component, if input U less then this programmable value, no enhancement did.
7-4	VDS_C1_U_HIGH [3:0]	**Red enhance U high threshold value control bit [3:0]** This field contains the high threshold value for U component, if input U larger then this programmable value, no enhancement did.

VDS_PROC 101 REG S3_65, R/W

Bit	7	6	5	4	3	2	1	0
	VDS_C1_Y_THRESH [2:0]			VDS_C1_BYPS	VDS_C1_U_HIGH [7:4]			

Bit	Name	Function
3-0	VDS_C1_U_HIGH [7:4]	**Red enhance U high threshold value control bit [7:4]** This field contains the high threshold value for U component, if input U larger then this programmable value, no enhancement did.
4	VDS_C1_BYPS	**Red enhance bypass control, active high** When this bit is 1, red enhancement will be bypassed.
7-5	VDS_C1_Y_THRESH [2:0]	**Red enhance Y threshold value control bit [2:0]** This field contains the Y threshold for red enhancement, when input Y larger than this programmable value, no enhancement did.

VDS_PROC 102 REG S3_66, R/W

Bit	7	6	5	4	3	2	1	0
	VDS_C2_TAG_LOW_SLOPE [2:0]			VDS_C1_Y_THRESH [7:3]				

Bit	Name	Function
4-0	VDS_C1_Y_THRESH [7:3]	**red enhance Y threshold value control bit [7:3]** This field contains the Y threshold for red enhancement, when input Y larger than this programmable value, no enhancement did.
7-5	VDS_C2_TAG_LOW_SLOPE [2:0]	**Green enhance angle tan value low threshold value control bit [2:0]** This filed contains the low threshold value for green enhance angle tan value, when the input UV angle tan value less than this programmable value, no enhancement did.

VDS_PROC 103 REG S3_67, R/W

Bit	7	6	5	4	3	2	1	0
	VDS_C2_TAG_HIGH_SLOPE [0]	VDS_C2_TAG_LOW_SLOPE [9:3]						

Bit	Name	Function
6-0	VDS_C2_TAG_LOW_SLOPE [9:3]	**Green enhance angle tan value low threshold value control bit [9:3]** This filed contains the low threshold value for green enhance angle tan value, when the input UV angle tan value less than this programmable value, no enhancement did.
7	VDS_C2_TAG_HIGH_SLOPE [0]	**Green enhance angle tan value high threshold value control bit [0]** This filed contains the high threshold value for green enhance angle tan value, when the input UV angle tan value larger than this programmable value, no enhancement did.

VDS_PROC 104 REG S3_68, R/W

Bit	7	6	5	4	3	2	1	0
	VDS_C2_TAG_HIGH_SLOPE [8:1]							

Bit	Name	Function
7-0	VDS_C2_TAG_HIGH_SLOPE [8:1]	**Green enhance angle tan value high threshold value control bit [8:1]** This filed contains the high threshold value for green enhance angle tan value, when the input UV angle tan value larger than this programmable value, no enhancement did.

VDS_PROC 105 REG S3_69, R/W

Bit	7	6	5	4	3	2	1	0
	VDS_C2_U_LOW [2:0]			VDS_C2_GAIN				VDS_C2_TAG_HIGH_SLOPE [9]

Bit	Name	Function
0	VDS_C2_TAG_HIGH_SLOPE [9]	**Green enhance angle tan value high threshold value control bit [9]** This filed contains the high threshold value for green enhance angle tan value, when the input UV angle tan value larger than this programmable value, no enhancement did.
4-1	VDS_C2_GAIN	**Color enhance gain control bit [3:0]** This field contains the gain control for green enhance, its range is (0~1)*16
7-5	VDS_C2_U_LOW [2:0]	**Green enhance U low threshold value control bit [2:0]** This field contains the low threshold value for U component, if input U less then this programmable value, no enhancement did.

VDS_PROC 106 REG S3_6A, R/W

Bit	7	6	5	4	3	2	1	0
	VDS_C2_U_HIGH [2:0]			VDS_C2_U_LOW [7:3]				

Bit	Name	Function
4-0	VDS_C2_U_LOW [7:3]	**Green enhance U low threshold value control bit [7:3]** This field contains the low threshold value for U component, if input U less then this programmable value, no enhancement did.
7-5	VDS_C2_U_HIGH [2:0]	**Green enhance U high threshold value control bit [2:0]** This field contains the high threshold value for U component, if input U larger then this programmable value, no enhancement did.

VDS_PROC 107 REG S3_6B, R/W

Bit	7	6	5	4	3	2	1	0
	VDS_C2_Y_THRESH [1:0]		VDS_C2_BYPS	VDS_C2_U_HIGH [7:3]				

Bit	Name	Function
4-0	VDS_C2_U_HIGH [7:3]	**Green enhance U high threshold value control bit [7:3]** This field contains the high threshold value for U component, if input U larger then this programmable value, no enhancement did.
5	VDS_C2_BYPS	**Green enhance bypass control** When this bit is 1, color enhancement will be bypassed.
7-6	VDS_C2_Y_THRESH [1:0]	**Green enhance Y threshold value control bit [1:0]** This field contains the Y threshold for green enhancement, when input Y larger than this programmable value, no enhancement did.

VDS_PROC 108 REG S3_6C, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED		VDS_C2_Y_THRESH [7:2]					

Bit	Name	Function
5-0	VDS_C2_Y_THRESH [7:2]	**Green enhance Y threshold value control bit [7:2]** This field contains the Y threshold for green enhancement, when input Y larger than this programmable value, no enhancement did.
7-6	RESERVED	

VDS_PROC 109 REG S3_6D, R/W

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

Bit	Name	Function
7-0	VDS_EXT_HB_ST [7:0]	**External used horizontal blanking start position control bit [7:0]** This field is used to program horizontal blanking start position, this blanking is for external used.

VDS_PROC 110 REG S3_6E, R/W

Bit	7	6	5	4	3	2	1	0
	VDS_EXT_HB_SP [3:0]				VDS_EXT_HB_ST [11:8]			

Bit	Name	Function
3-0	VDS_EXT_HB_ST [11:8]	**External used horizontal blanking start position control bit [11:8]** This field is used to program horizontal blanking start position, this blanking is for external used.
7-4	VDS_EXT_HB_SP [3:0]	**External used horizontal blanking stop position control bit [3:0]** This field is used to program horizontal blanking stop position, this blanking is for external used.

VDS_PROC 111 REG S3_6F, R/W

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

Bit	Name	Function
7-0	VDS_EXT_HB_SP [11:4]	**External used horizontal blanking stop position control bit [11:4]** This field is used to program horizontal blanking stop position, this blanking is for external used.

VDS_PROC 112 REG S3_70, R/W

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

Bit	Name	Function
7-0	VDS_EXT_VB_ST [7:0]	**External used vertical blanking start position control bit [7:0]** This field is used to program vertical blanking start position, this blanking is for external used.

VDS_PROC 113 REG S3_71, R/W

Bit	7	6	5	4	3	2	1	0
	VDS_EXT_VB_SP [3:0]				RESERVED	VDS_EXT_VB_ST [10:8]		

Bit	Name	Function
2-0	VDS_EXT_VB_ST [10:8]	**External used vertical blanking start position control bit [10:8]** This field is used to program vertical blanking start position, this blanking is for external used.
3	RESERVED	
7-4	VDS_EXT_VB_SP [3:0]	**External used vertical blanking stop position control bit [3:0]** This field is used to program vertical blanking stop position, this blanking is for external used.

VDS_PROC 114 REG S3_72, R/W

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

Bit	Name	Function
6-0	VDS_EXT_VB_SP [10:4]	**External used vertical blanking stop position control bit [10:4]** This field is used to program vertical blanking stop position, this blanking is for external used.
7	VDS_SYNC_IN_SEL	**VDS module input sync selection control** When this bit is 1, the sync to VDS module is from external (out of the CHIP); When this bit is 0, the sync to VDS module is from IF module.

VDS_PROC 115 REG S3_73, R/W

Bit	7	6	5	4	3	2	1	0
	VDS_BLUE_UGAIN				VDS_BLUE_BYPS	VDS_BLUE_RANGE		

Bit	Name	Function
2-0	VDS_BLUE_RANGE	**Blue extend range control bit [2:0]** This field defines the range for blue extend.

VDS_BLUE_RANGE [2:0]			Real range
0	0	0	1
0	0	1	2
0	1	0	4
0	1	1	8
1	0	0	16
1	0	1	32
1	1	0	64
1	1	1	128

Bit	Name	Function
3	VDS_BLUE_BYPS	**Blue extend bypass control, active high** When this bit is 1, the blue extend process will be bypassed
7-4	VDS_BLUE_UGAIN	**Blue extend U gain control bit [3:0]** This field defines the U gain for U component in the area which should do blue extend, its range is (0~1)*16.

VDS_PROC 116 REG S3_74, R/W

Bit	7	6	5	4	3	2	1	0
	VDS_BLUE_Y_LEV				VDS_BLUE_VGAIN			

Bit	Name	Function
3-0	VDS_BLUE_VGAIN	**Blue extend V gain control bit [3:0]** This field defines the V gain for V component in the area which should do blue extend, its range is (0~1)*16.
7-4	VDS_BLUE_Y_LEV	**Blue extend Y level threshold control bit [3:0]** This field defines the Y threshold value of blue extend, the real level in the circuit is 16*blue_y_th + 15, the blue extend process done only when Y value larger than this level (real level).

PIP 00 REG S3_80, R/W

Bit	7	6	5	4	3	2	1	0
	PIP_DYN_BYPS	PIP_SUB_16B_SEL	PIP_Y_DELAY		PIP_TAP3_BYPS	PIP_V_DELAY	PIP_U_DELAY	PIP_UV_FLIP

Bit	Name	Function
0	PIP_UV_FLIP	**422 to 444 conversion UV flip control** This bit is used to flip UV, when this bit is 1, UV position will be flipped.
1	PIP_U_DELAY	**UV 422 to 444 conversion U delay** When this bit is 1, U will delay 1 clock, otherwise, no delay for internal pipe.
2	PIP_V_DELAY	**UV 422 to 444 conversion V delay** When this bit is 1, V will delay 1 clock, otherwise, no delay for internal pipe.
3	PIP_TAP3_BYPS	**Tap3 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
5-4	PIP_Y_DELAY	**Y compensation 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.

PIP_Y_DELAY [1:0]		Y delay
0	0	1
0	1	2
1	0	3
1	1	4

6	PIP_SUB_16B_SEL	**PIP 16-bit sub-picture select, active high** When this bit is 1, select 16-bit sub-picture; When it is 0, select 24-bit sub-picture.
7	PIP_DYN_BYPS	**Dynamic range expansion bypass control, active high** When this bit is 1, data will bypass the dynamic range expansion process.

PIP 01 REG S3_81, R/W

Bit	7	6	5	4	3	2	1	0
	PIP_EN	RESERVED			PIP_DREG_BYPS	RESERVED		PIP_CONVT_BYPS

Bit	Name	Function
0	PIP_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.
2-1	RESERVED	
3	PIP_DREG_BYPS	**Input data bypass the negedge trigger 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.
6-4	RESERVED	
7	PIP_EN	**PIP enable, active high** When this bit is 1, PIP insertion is enabled, otherwise, no PIP

PIP 02 REG S3_82, R/W

Bit	7	6	5	4	3	2	1	0
	PIP_Y_GAIN							

Bit	Name	Function
0	PIP_Y_GAIN	**Y dynamic range expansion gain control bit [7:0]** This field contains the Y gain value in dynamic range expansion process, its range is (0 ~ 2)*128.

PIP 03 REG S3_83, R/W

Bit	7	6	5	4	3	2	1	0
	PIP_U_GAIN							

Bit	Name	Function
0	PIP_U_GAIN	**U dynamic range expansion gain control bit [7:0]** This field contains the U gain value in dynamic range expansion process, its range is (0 ~ 4)*64.

PIP 04 REG S3_84, R/W

Bit	7	6	5	4	3	2	1	0
	PIP_V_GAIN							

Bit	Name	Function
0	PIP_V_GAIN	**V dynamic range expansion gain control bit [7:0]** This field contains the V gain value in dynamic range expansion process, its range is (0 ~ 4)*64.

PIP 05 REG S3_85, R/W

Bit	7	6	5	4	3	2	1	0
	PIP_Y_OFST							

Bit	Name	Function
0	PIP_Y_OFST	**Y dynamic range expansion offset control bit [7:0]** This field contains the Y offset value in dynamic range expansion process, its range is –128 ~ 127.

PIP 06 REG S3_86, R/W

Bit	7	6	5	4	3	2	1	0
	PIP_U_OFST							

Bit	Name	Function
0	PIP_U_OFST	**U dynamic range expansion offset control bit [7:0]** This field contains the U offset value in dynamic range expansion process, its range is –128 ~ 127.

PIP 07 REG S3_87, R/W

Bit	7	6	5	4	3	2	1	0
	PIP_V_OFST							

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

PIP 08 REG S3_88, R/W

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

Bit	Name	Function
0	PIP_H_ST [7:0]	**PIP window horizontal start position control bit [7:0]** This field contains the horizontal start position of PIP window.

PIP 09 REG S3_89, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED				PIP_H_ST [11:8]			

Bit	Name	Function
3-0	PIP_H_ST [11:8]	**PIP window horizontal start position control bit [11:8]** This field contains the horizontal start position of PIP window.
7-4	RESERVED	

PIP 10 REG S3_8A, R/W

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

Bit	Name	Function
0	PIP_H_SP [7:0]	**PIP window horizontal stop position control bit [7:0]** This field contains the horizontal stop position of PIP window.

PIP 11 REG S3_8B, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED				PIP_H_SP [11:8]			

Bit	Name	Function
3-0	PIP_H_SP [11:8]	**PIP window horizontal stop position control bit [11:8]** This field contains the horizontal stop position of PIP window.
7-4	RESERVED	

PIP 12 REG S3_8C, R/W

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

Bit	Name	Function
0	PIP_V_ST [7:0]	**PIP window vertical start position control bit [7:0]** This field contains the vertical start position of PIP window.

PIP 13 REG S3_8D, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED					PIP_H_ST [10:8]		

Bit	Name	Function
2-0	PIP_V_ST [10:8]	**PIP window vertical start position control bit [10:8]** This field contains the vertical start position of PIP window.
7-3	RESERVED	

PIP 14 REG S3_8E, R/W

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

Bit	Name	Function
0	PIP_V_SP [7:0]	**PIP window vertical stop position control bit [7:0]** This field contains the vertical stop position of PIP window.

PIP 15 REG S3_8F, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED					PIP_H_SP [10:8]		

Bit	Name	Function
2-0	PIP_V_SP [10:8]	**PIP window vertical stop position control bit [10:8]** This field contains the vertical stop position of PIP window.
7-3	RESERVED	

Chapter 09. OSD REGISTERS

OSD_TOP_00 REG S0_90, R/W

Bit	7	6	5	4	3	2	1	0
	OSD_MENU_EN	OSD_DISP_EN	OSD_VERTICAL_ZOOM		OSD_HORIZONTAL_ZOOM			OSD_SW_RESET

Bit	Name	Function
0	OSD_SW_RESET	**Software reset for module , active high** When this bit is 1, it reset osd_top module
3-1	OSD_HORIZONTAL_ZOOM	**Osd horizontal zoom select**

OSD_HORIZONTAL_ZOOM [2:0]			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_VERTICAL_ZOOM	**Osd vertical zoom select**

OSD_VERTICAL_ZOOM [1:0]		SIZE
0	0	1
0	1	2
1	0	3
1	1	4

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

OSD_TOP_01 REG S0_91, R/W

Bit	7	6	5	4	3	2	1	0
	OSD_MENU_MOD_SEL				OSD_MENU_ICON_SEL			

Bit	Name	Function
3-0	OSD_MENU_ICON_SEL	**Osd menu icons select**

OSD_MENU_ICON_SEL [3:0]				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	OSD_MENU_MOD_SEL	**Osd icons modification select**

OSD_MENU_MOD_SEL [3:0]				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 S0_92, R/W

Bit	7	6	5	4	3	2	1	0
	OSD_MENU_BAR_BORD_COR [1:0]		OSD_MENU_BAR_FONT_BGCOR			OSD_MENU_BAR_FONT_FORCOR		

Bit	Name	Function
2-0	OSD_MENU_BAR_FONT_FORCOR	**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	OSD_MENU_BAR_FONT_BGCOR	**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	OSD_MENU_BAR_BORD_COR [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 S0_93, R/W

Bit	7	6	5	4	3	2	1	0
	OSD_COMMAND_FINISH	OSD_MENU_SEL_BGCOR			OSD_MENU_SEL_FORCOR			OSD_MENU_BAR_BORD_COR [2]

Bit	Name	Function
0	OSD_MENU_BAR_BORD_COR [2]	**Menu or bar border color.** It is the bit 2 of menu or bar border color.
3-1	OSD_MENU_SEL_FORCOR	**Selected icon or bar's icon foreground color.**
6-4	OSD_MENU_SEL_BGCOR	**Selected icon or bar's icon background color.**
7	OSD_COMMAND_FINISH	**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 S0_94, R/W

Bit	7	6	5	4	3	2	1	0
	OSD_TEST_SEL				OSD_INT_NG_LAT	OSD_YCBCR_RGB_FORMATE	RESERVED	OSD_MENU_DISP_STYLE

Bit	Name	Function
0	OSD_MENU_DISP_STYLE	**Menu display in row or column mode.** When 1, osd menu displays in row style, else in column style.
2	OSD_YCBCR_RGB_FORMATE	**YCbCr or RGB output.** Osd display in YCbCr or RGB format, when set to 1, display in YCbCr mode
3	OSD_INT_NG_LAT	**V2clk latch osd data with negative enable.** When set to 1, V2CLK clock can latch osd data with negative edge
7-4	OSD_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.

OSD_REG_05 REG S0_95, R/W

Bit	7	6	5	4	3	2	1	0
	OSD_MENU_HORI_START							

Bit	Name	Function
7-0	OSD_MENU_HORI_START	**Menu or bar horizontal start address** The real address is { MENU_BAR_HORZ_START [7:0], 3'h0}.

OSD_REG_06 REG S0_96, R/W

Bit	7	6	5	4	3	2	1	0
	OSD_MENU_VER_START							

Bit	Name	Function
7-0	OSD_MENU_VER_START	**Menu or bar vertical start address** The real address is { MENU_BAR_VIRT_START [7:0], 3'h0}.

OSD_REG_07 REG S0_97, R/W

Bit	7	6	5	4	3	2	1	0
	OSD_BAR_LENGTH							

Bit	Name	Function
7-0	OSD_BAR_LENGTH	**BAR DISPLAY TOTAL LENGTH** Bar display on screen's total length, when horizontal zoom is 0.

OSD_REG_08 REG S0_98, R/W

Bit	7	6	5	4	3	2	1	0
	OSD_BAR_FOREGROUND_VALUE							

Bit	Name	Function
7-0	OSD_BAR_FOREGROUND_VALUE	**Bar foreground color value.** The value of this register indicates the real value of icon, such as brightness's value is 8'hf0, then this register is also programmed to 8'hf0.

Chapter 10. MODE_DETECT REGISTERS

MODE_DET 00 REG S1_60, R/W

Bit	7	6	5	4	3	2	1	0
	MD_HPERIOD_UNLOCK_VALUE			MD_HPERIOD_LOCK_VALUE				

Bit	Name	Function
4-0	MD_HPERIOD_LOCK_VALUE	**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	MD_HPERIOD_UNLOCK_VALUE	**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 S1_61, R/W

Bit	7	6	5	4	3	2	1	0
	MD_VPERIOD_UNLOCK_VALUE			MD_VPERIOD_LOCK_VALUE				

Bit	Name	Function
4-0	MD_VPERIOD_LOCK_VALUE	**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	MD_VPERIOD_UNLOCK_VALUE	**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 S1_62, R/W

Bit	7	6	5	4	3	2	1	0
	MD_WEN_CNTRL		MD_NTSC_INT_CNTRL					

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

MD_WEN_CNTRL [1:0]		Error Range
0	0	1
0	1	2
1	0	3
1	1	4

MODE_DET 03 REG S1_63, R/W

Bit	7	6	5	4	3	2	1	0
	MD_VS_FLIP	MD_HS_FLIP	MD_PAL_INT_CNTRL					

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

MODE_DET 04 REG S1_64, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED	MD_NTSC_PRG_CNTRL						

Bit	Name	Function
6-0	MD_NTSC_PRG_CNTRL	**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 S1_65, R/W

Bit	7	6	5	4	3	2	1	0
	MD_SEL_VGA60	MD_VGA_CNTRL						

Bit	Name	Function
6-0	MD_VGA_CNTRL	**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	MD_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 S1_66, R/W

Bit	7	6	5	4	3	2	1	0
	MD_VGA_75HZ_CNTRL							

Bit	Name	Function
7-0	MD_VGA_75HZ_CNTRL	**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 S1_67, R/W

Bit	7	6	5	4	3	2	1	0
	MD_VGA_85HZ_CNTRL							

Bit	Name	Function
7-0	MD_VGA_85HZ_CNTRL	**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 S1_68, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED	MD_V1250_VCNTRL						

Bit	Name	Function
6-0	MD_V1250_VCNTRL	**Vertical 1250 Line Mode Vertical Detect Value** All vertical 1250 lines mode vertical detect value
7	RESERVED	

MODE_DET 09 REG S1_69, R/W

Bit	7	6	5	4	3	2	1	0
	MD_V1250_HCNTRL							

Bit	Name	Function
7-0	MD_V1250_HCNTRL	**Vertical 1250 Line Mode Horizontal Detect Value** Vertical 1250 lines, horizontal 866 pixels mode detect value

MODE_DET 10 REG S1_6A, R/W

Bit	7	6	5	4	3	2	1	0
	MD_SVGA_60HZ_CNTRL							

Bit	Name	Function
7-0	MD_SVGA_60HZ_CNTRL	**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 S1_6B, R/W

Bit	7	6	5	4	3	2	1	0
	MD_SVGA_75HZ_CNTRL							

Bit	Name	Function
7-0	MD_SVGA_75HZ_CNTRL	**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 S1_6C, R/W

Bit	7	6	5	4	3	2	1	0
	MD_SVGA_85HZ_CNTRL							

Bit	Name	Function
7-0	MD_SVGA_85HZ_CNTRL	**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 S1_6D, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED	MD_XGA_CNTRL						

Bit	Name	Function
6-0	MD_XGA_CNTRL	**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 S1_6E, R/W

Bit	7	6	5	4	3	2	1	0
	MD_XGA_60HZ_CNTRL							

Bit	Name	Function
7-0	MD_XGA_60HZ_CNTRL	**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 S1_6F, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED	MD_XGA_70HZ_CNTRL						

Bit	Name	Function
6-0	MD_XGA_70HZ_CNTRL	**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 S1_70, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED	MD_XGA_75HZ_CNTRL						

Bit	Name	Function
6-0	MD_XGA_75HZ_CNTRL	**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 S1_71, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED	MD_XGA_85HZ_CNTRL						

Bit	Name	Function
6-0	MD_XGA_85HZ_CNTRL	**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 S1_72, R/W

Bit	7	6	5	4	3	2	1	0
	MD_SXGA_CNTRL							

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

MODE_DET 19 REG S1_73, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED	MD_SXGA_60HZ_CNTRL						

Bit	Name	Function
6-0	MD_SXGA_60HZ_CNTRL	**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 S1_74, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED	MD_SXGA_75HZ_CNTRL						

Bit	Name	Function
6-0	MD_SXGA_75HZ_CNTRL	**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 S1_75, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED	MD_SXGA_85HZ_CNTRL						

Bit	Name	Function
6-0	MD_SXGA_85HZ_CNTRL	**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 S1_76, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED	MD_HD720P_CNTRL						

Bit	Name	Function
6-0	MD_HD720P_CNTRL	**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 S1_77, R/W

Bit	7	6	5	4	3	2	1	0
	MD_HD720P_60HZ_CNTRL							

Bit	Name	Function
7-0	MD_HD720P_60HZ_CNTRL	**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 S1_78, R/W

Bit	7	6	5	4	3	2	1	0
	MD_HD720P_50HZ_CNTRL							

Bit	Name	Function
7-0	MD_HD720P_50HZ_CNTRL	**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 S1_79, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED	MD_HD1125I_CNTRL						

Bit	Name	Function
6-0	MD_HD1125I_CNTRL	**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 S1_7A, R/W

Bit	7	6	5	4	3	2	1	0
	MD_HD2200_1125I_CNTRL							

Bit	Name	Function
7-0	MD_HD2200_1125I_CNTRL	**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 S1_7B, R/W

Bit	7	6	5	4	3	2	1	0
	MD_HD2640_1125I_CNTRL							

Bit	Name	Function
7-0	MD_HD2640_1125I_CNTRL	**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 S1_7C, R/W

Bit	7	6	5	4	3	2	1	0
	MD_HD1125P_CNTRL							

Bit	Name	Function
7-0	MD_HD1125P_CNTRL	**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 S1_7D, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED	MD_HD2200_1125P_CNTRL						

Bit	Name	Function
6-0	MD_HD2200_1125P_CNTRL	**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 S1_7E, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED	MD_HD2640_1125P_CNTRL						

Bit	Name	Function
6-0	MD_HD2640_1125P_CNTRL	**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 S1_7F, R/W

Bit	7	6	5	4	3	2	1	0
	MD_HD1250P_CNTRL							

Bit	Name	Function
7-0	MD_HD1250P_CNTRL	**1080P Mode 2376x1250P Vertical Detect Value** If the vertical period number is equal to the defined value, It is HD2376x1250P mode

MODE_DET 32 REG S1_80, R/W

Bit	7	6	5	4	3	2	1	0
	MD_USER_DEF_VCNTRL							

Bit	Name	Function
7-0	MD_USER_DEF_VCNTRL	**User Defined Mode Vertical Detect Value** If the vertical period number is equal to the defined value, It is user-defined mode.

MODE_DET 33 REG S1_81, R/W

Bit	7	6	5	4	3	2	1	0
	MD_USER_DEF_HCNTRL							

Bit	Name	Function
7-0	MD_USER_DEF_HCNTRL	**User Defined Mode Horizontal Detect Value** If the horizontal period number is equal to the defined value, It is user-defined mode.

MODE_DET 34 REG S1_82, R/W

Bit	7	6	5	4	3	2	1	0
	MD_H_USER_ID	MD_DET_BYPS_H	MD_TIMER_DET_EN_V	MD_TIMER_DET_EN_H	MD_SW_USER_ID	MD_SW_DET_EN	MD_NOSYNC_USER_ID	MD_NOSYNC_DET_EN

Bit	Name	Function
0	MD_NOSYNC_DET_EN	**Sync Connection Detect Enable** Detect the horizontal sync signal if connect or not. 0: user mode 1: auto detect
1	MD_NOSYNC_USER_ID	**Sync Connection Detect User Defined ID** User defined indicator in user mode. 0: sync connected. 1: no sync connected.
2	MD_SW_DET_EN	**Mode Switch Detect Enable** Enable bit of auto detect if the mode changed or not. 0: user mode 1: auto detect
3	MD_SW_USER_ID	**Mode Switch Detect User Defined ID** User defined indicator in user mode. 0->1: mode changed. 1->0: mode changed.
4	MD_TIMER_DET_EN_H	**Horizontal Unstable Estimation Timer Detect Enable** Enable the timer detect result in horizontal unstable estimation. 0: use the hstable indicator in hperiod detect. 1: use the timer detected unstable indicator.
5	MD_TIMER_DET_EN_V	**Vertical Unstable Estimation Timer Detect Enable** Enable the timer detect result in vertical unstable estimation. 0: use the vstable indicator in vperiod detect. 1: use the timer detected unstable indicator.
6	MD_DET_BYPS_H	**Horizontal Unstable Estimation Bypass Control** Bypass the horizontal unstable estimation 0: auto mode 1: user mode
7	MD_H_USER_ID	**Horizontal Unstable Estimation User Defined ID** User defined indicator in user mode. 0: stable 1: unstable

MODE_DET 35 REG S1_83, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED		MD_UNSTABLE_LOCK_VALUE				MD_V_USER_ID	MD_DET_BYPS_V

Bit	Name	Function
0	MD_DET_BYPS_V	**Vertical Unstable Estimation Bypass Control** Bypass the vertical unstable estimation auto detect 0: auto mode 1: user mode
1	MD_V_USER_ID	**Vertical Unstable Estimation User Defined ID** User defined indicator in user mode. 0: stable 1: unstable
5-2	MD_UNSTABLE_LOCK_VALUE	**Unstable Estimation Lock Value** If the internal counter equals the defined value, the unstable indicator will be high. Horizontal and vertical estimation shared this value.
7-6	RESERVED	

Chapter 11. ADC REGISTERS

ADC CLK CONTROL 00 REG S5_00, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED			ADC_CLK_ICLK1X	ADC_CLK_ICLK2X	ADC_CLK_PLLAD	ADC_CLK_PA	

Bit	Name	Function
1-0	ADC_CLK_PA	**Clock selection for PA_ADC** When = 00, PA_ADC input clock is from PLLAD's CLKO2 When = 01, PA_ADC input clock is from PCLKIN When = 10, PA_ADC input clock is from V4CLK When = 11, reserved
2	ADC_CLK_PLLAD	**Clock selection for PLLAD** When = 0, PLLAD input clock is from sync processor When = 1, PLLAD input clock is from OSC
3	ADC_CLK_ICLK2X	**ICLK2X control** When = 0, ICLK2X = ADC output clock When = 1, ICLK2X = ADC output clock / 2
4	ADC_CLK_ICLK1X	**ICLK1X control** When = 0, ICLK1X = ICLK2X When = 1, ICLK1X = ICLK2X /2
7-5	RESERVED	**Reserved**

ADC CONTROL 00 REG S5_02, R/W

Bit	7	6	5	4	3	2	1	0
	ADC_INPUT_SEL		ADC_SOGCTRL					ADC_SOGEN

Bit	Name	Function
0	ADC_SOGEN	**ADC SOG enable** When = 0, ADC disable SOG mode When = 1, ADC enable SOG mode
5-1	ADC_SOGCTRL	**SOG control signal**
7-6	ADC_INPUT_SEL	**ADC input selection** When = 00, R0/G0/B0/SOG0 as input When = 01, R1/G1/B1/SOG1 as input When = 10, R2/G2/B2 as input When = 11, reserved

ADC CONTROL 01 REG S5_03, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED		ADC_FLTR		ADC_RYSEL_B	ADC_RYSEL_G	ADC_RYSEL_R	ADC_POWDZ

Bit	Name	Function
0	ADC_POWDZ	**ADC power down control** When = 0, ADC in power down mode When = 1, ADC work normally
1	ADC_RYSEL_R	**Clamp to ground or midscale for R ADC** When = 0, clamp to GND When = 1, clamp to midscale
2	ADC_RYSEL_G	**Clamp to ground or midscale for G ADC** When = 0, clamp to GND When = 1, clamp to midscale
3	ADC_RYSEL_B	**Clamp to ground or midscale for B ADC** When = 0, clamp to GND When = 1, clamp to midscale
5-4	ADC_FLTR	**ADC internal filter control** When = 00, 150MHz When = 01, 110MHz When = 10, 70MHz When = 11, 40MHz
7-6	RESERVED	**Reserved**

ADC CONTROL 02 REG S5_04, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED			ADC_TR_ISEL			ADC_TR_RSEL	

Bit	Name	Function
1-0	ADC_TR_RSEL	**REF test resistor selection**
4-2	ADC_TR_ISEL	**REF test currents selection**
7-5	RESERVED	

ADC CONTROL 03 REG S5_05, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED			ADC_TA_CTRL				ADC_TA_EN

Bit	Name	Function
0	ADC_TA_EN	**ADC test enable** When = 0, ADC work normally When = 1, ADC is in test mode
4-1	ADC_TA_CTRL	**ADC test bus control bit**
7-5	RESERVED	**Reserved**

ADC CONTROL 04 REG S5_06, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED	ADC_ROFCTRL						

Bit	Name	Function
6-0	ADC_ROFCTRL	**Offset control for R channel of ADC**
7	RESERVED	**Reserved**

ADC CONTROL 05 REG S5_07, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED	ADC_GOFCTRL						

Bit	Name	Function
6-0	ADC_GOFCTRL	**Offset control for G channel of ADC**
7	RESERVED	**Reserved**

ADC CONTROL 06 REG S5_08, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED	ADC_BOFCTRL						

Bit	Name	Function
6-0	ADC_BOFCTRL	Offset control for B channel of ADC
7	RESERVED	Reserved

ADC CONTROL 07 REG S5_09, R/W

Bit	7	6	5	4	3	2	1	0
	ADC_RGCTRL							

Bit	Name	Function
7-0	ADC_RGCTRL	Gain control for R channel of ADC

ADC CONTROL 08 REG S5_0A, R/W

Bit	7	6	5	4	3	2	1	0
	ADC_GGCTRL							

Bit	Name	Function
7-0	ADC_GGCTRL	Gain control for G channel of ADC

ADC CONTROL 09 REG S5_0B, R/W

Bit	7	6	5	4	3	2	1	0
	ADC_BGCTRL							

Bit	Name	Function
7-0	ADC_BGCTRL	Gain control for B channel of ADC

ADC CONTROL 10 REG S5_0C, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED				ADC_TEST			ADC_CKBS

Bit	Name	Function
0	ADC_CKBS	**ADC output clock invert control** When = 0, default When = 1, ADC output clock will be invert
4-1	ADC_TEST	**For ADC test reserved**
7-5	RESERVED	**Reserved**

ADC AUTO_OFST 00 REG S5_0E, R/W

Bit	7	6	5	4	3	2	1	0
	ADC_AUTO_OFST_TEST	RESERVED	ADC_AUTO_OFST_STEP		ADC_AUTO_OFST_DELAY		ADC_AUTO_OFST_PRD	ADC_AUTO_OFST_EN

Bit	Name	Function
0	ADC_AUTO_OFST_EN	**Auto offset adjustment enable** When = 0, auto offset adjustment disable When = 1, auto offset adjustment enable
1	ADC_AUTO_OFST_PRD	**Offset adjustment by frame** When = 0, offset adjustment by frame When = 1, offset adjustment by line
3-2	ADC_AUTO_OFST_DELAY	**Horizontal sample delay control** When = 00, offset adjustment horizontal sample delay 1 pipe When = 01, offset adjustment horizontal sample delay 2 pipe When = 10, offset adjustment horizontal sample delay 3 pipe When = 11, offset adjustment horizontal sample delay 4 pipe
5-4	ADC_AUTO_OFST_STEP	**Offset adjustment step control** When = 00, offset adjustment by absolute difference. When = 01, offset adjustment by 1 When = 10, offset adjustment by 2 When = 11, offset adjustment by 3
6	RESERVED	**Reserved**
7	ADC_AUTO_OFST_TEST	**Auto offset adjustment test control**

ADC AUTO_OFST 01 REG S5_0F, R/W

Bit	7	6	5	4	3	2	1	0
	ADC_AUTO_OFST_V_RANGE				ADC_AUTO_OFST_U_RANGE			

Bit	Name	Function
3-0	ADC_AUTO_OFST_U_RANGE	**U channel offset detection range**
		Define U channel offset detection range 0~15
7-4	ADC_AUTO_OFST_V_RANGE	**V channel offset detection range**
		Define V channel offset detection range 0~15

PLLAD CONTROL 00 REG S5_11, R/W

Bit	7	6	5	4	3	2	1	0
	PLLAD_LAT	PLLAD_BPS	PLLAD_FS	PLLAD_PDZ	PLLAD_TS	PLLAD_TEST	PLLAD_LEN	PLLAD_VCORST

Bit	Name	Function
0	PLLAD_VCORST	**VCORST**
		Initial VCO control voltage
1	PLLAD_LEN	**LEN**
		Enable signal for clock
2	PLLAD_TEST	**TEST**
		Test clock selection
3	PLLAD_TS	**TS**
		Test clock selection and HSL clock selection
4	PLLAD_PDZ	**PDZ**
		When = 0, PLLAD is power down mode When = 1, PLLAD work normally
5	PLLAD_FS	**FS, VCO gain selection**
		When = 0, default When = 1, high gain selected
6	PLLAD_BPS	**BPS**
		When = 0, default When = 1, bypass input clock to CKO1 and CKO2
7	PLLAD_LAT	**Latch control for PLLAD control**
		This bit's rising edge is used to trigger PLLAD control bit: ND, MD, KS, CKOS, ICP

PLLAD CONTROL 01 REG S5_12, R/W

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

Bit	Name	Function
7-0	PLLAD_MD [7:0]	**MD[7:0]** PLLAD feedback divider control

PLLAD CONTROL 02 REG S5_13, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED				PLLAD_MD [11:8]			

Bit	Name	Function
3-0	PLLAD_MD [11:8]	**MD[11:8]** PLLAD feedback divider control
7-4	RESERVED	**Reserved**

PLLAD CONTROL 03 REG S5_14, R/W

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

Bit	Name	Function
7-0	PLLAD_ND [7:0]	**ND[7:0]** PLLAD input divider control

PLLAD CONTROL 04 REG S5_15, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED				PLLAD_ND [11:8]			

Bit	Name	Function
3-0	PLLAD_ND [11:8]	**ND[11:8]** PLLAD input divider control
7-4	RESERVED	**Reserved**

PLLAD CONTROL 05 REG S5_16, R/W

Bit	7	6	5	4	3	2	1	0
	PLLAD_CKOS		PLLAD_KS		PLLAD_S		PLLAD_R	

Bit	Name	Function
1-0	PLLAD_R	**R** Skew control for testing
3-2	PLLAD_S	**S** Skew control for testing
5-4	PLLAD_KS	**KS** VCO post divider control, it is determined by CKO frequency When = 00, divide by 1 (162MHz~80MHz) When = 01, divide by 2 (80MHz~40MHz) When = 10, divide by 4 (40MHz~20MHz) When = 11, divide by 8 (20MHz~min MHz)
7-6	PLLAD_CKOS	**CKOS** PLLAD CKO2 output clock selection

PLLAD_KS	PLLAD_CKOS	CKO2 freq / CKO1 freq
00	00	1
	01	1/2
	10	1/4
	11	1/8
01	00	2
	01	1
	10	1/2
	11	1/4
10	00	4
	01	2
	10	1
	11	1/2
11	00	8
	01	4
	10	2
	11	1

PLLAD CONTROL 06 REG S5_17, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED					PLLAD_ICP		

Bit	Name	Function
2-0	PLLAD_ICP	**ICP** Charge pump current selection When = 000, Icp = 50uA When = 001, Icp = 100uA When = 010, Icp = 150uA When = 011, Icp = 250uA When = 100, Icp = 350uA When = 101, Icp = 500uA When = 110, Icp = 750uA When = 111, Icp = 1mA
7-3	RESERVED	**Reserved**

PA_ADC CONTROL 00 REG S5_18, R/W

Bit	7	6	5	4	3	2	1	0
	PA_ADC_LAT	PA_ADC_LOCKOFF	PA_ADC_S					PA_ADC_BYPSZ

Bit	Name	Function
0	PA_ADC_BYPSZ	**BYPSZ, PA for ADC bypass control** When = 0, PA_ADC is bypass When = 1, PA_ADC work normally
5-1	PA_ADC_S	**PA_ADC phase control**
6	PA_ADC_LOCKOFF	**LOCKOFF** When = 0, default When = 1, PA_ADC lock circuit disable
7	PA_ADC_LAT	**PA_ADC latch signal** This bit's rising edge is used to trigger PA_ADC_CNTRL_[5:1]

PA_SYNC PROC CONTROL 00 REG S5_19, R/W

Bit	7	6	5	4	3	2	1	0
	PA_SP_LAT	PA_SP_LOCKOFF	PA_SP_S					PA_SP_BYPSZ

Bit	Name	Function
0	PA_SP_BYPSZ	**BYPSZ, PA for PLLAD bypass control** When = 0, PA_PLLAD is bypass When = 1, PA_PLLAD work normally
5-1	PA_SP_S	**PA_PLLAD phase control**
6	PA_SP_LOCKOFF	**LOCKOFF** When = 0, default When = 1, PA_PLLAD lock circuit disable
7	PA_SP_LAT	**PA_PLLAD latch signal** This bit's rising edge is used to trigger PA_PLLAD_CNTRL_[5:1]

DEC_REG_00 REG S5_1E, R/W

Bit	7	6	5	4	3	2	1	0
	DEC_WEN_MODE	RESERVED						

Bit	Name	Function
6-0	RESERVED	
7	DEC_WEN_MODE	**Write enable mode enable.** When this bit is 1, then decimator will drop data by write enable signal generated by horizontal sync, else write enable is not used.

DEC_REG_01 REG S5_1F, R/W

Bit	7	6	5	4	3	2	1	0
	DEC_IDREG_EN	DEC_TEST_SEL				DEC_MATRIX_BYPS	DEC2_BYPS	DEC1_BYPS

Bit	Name	Function
0	DEC1_BYPS	**The 4x to 2x decimator bypass enable** When 1, the 4x to 2x decimator bypass.
1	DEC2_BYPS	**The 2x to 1x decimator bypass enable** When 1, the 2x to 1x decimator hypass.
2	DEC_MATRIX_BYPS	**Color space convert bypass enable** When set to 1, color space convert module bypass.
6-3	DEC_TEST_SEL	**Test logic output select.** DEC_TEST_SEL[0], test logic output enable, when set to 1, test logic can output. DEC_TEST_SEL[3:1] select 8 test logics to test bus.
7	DEC_IDREG_EN	**Negative clock edge latch input hsync and vsync enable** When set to 1, decimator 4x clock will latch HSYNC and VSYNC with falling edge.

Chapter 12. SYNC_PROC REGISTERS

SYNC_PROC 00 REG S5_20, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED			SP_JITTER_SYNC	SP_EXT_SYNC_SEL	SP_SOG_P_INV	SP_SOG_P_ATO	SP_SOG_SRC_SEL

Bit	Name	Function
0	SP_SOG_SRC_SEL	**sog_src_sel** Sog signal source select. 0: from ADC. 1: select hs as sog source.
1	SP_SOG_P_ATO	**sog_p_ato** sog auto correct polarity
2	SP_SOG_P_INV	**Sog invert** Invert sog.
3	SP_EXT_SYNC_SEL	**ext_sync_sel** Ext 2 set Hs_Hs select
4	SP_JITTER_SYNC	**Sync using both rising and falling trigger** Use falling and rising edge to sync input Hsync
7-5	RSERVED	

SYNC_PROC 01 REG S5_21, R/W

Bit	7	6	5	4	3	2	1	0
	SP_SYNC_TGL_THD							

Bit	Name	Function
7-0	SP_SYNC_TGL_THD	**h active detect control** Sync toggle times threshold

SYNC_PROC 02 REG S5_22, R/W

Bit	7	6	5	4	3	2	1	0
	SP_L_DLT_REG							

Bit	Name	Function
7-0	SP_L_DLT_REG	**h active detect control** Sync pulse width different threshold (little than this as equal).

SYNC_PROC 03 REG S5_23, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED							

Bit	Name	Function
7-0	RESERVED	

SYNC_PROC 04 REG S5_24, R/W

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

Bit	Name	Function
7-0	SP_T_DLT_REG [7:0]	**H active detect control** H total width different threshold

SYNC_PROC 05 REG S5_25, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED				SP_T_DLT_REG [11:8]			

Bit	Name	Function
3-0	SP_T_DLT_REG [11:8]	**H active detect control** H total width different threshold
7-4	RESERVED	

SYNC_PROC 06 REG S5_26, R/W

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

Bit	Name	Function
7-0	SP_SYNC_PD_THD [7:0]	**H active detect control** H sync pulse width threshold

SYNC_PROC 07 REG S5_27, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED				SP_SYNC_PD_THD [11:8]			

Bit	Name	Function
3-0	SP_SYNC_PD_THD [11:8]	**H active detect control** H sync pulse width threshold
7-4	RESERVED	

SYNC_PROC 08 REG S5_2A, R/W

Bit	7	6	5	4	3	2	1	0
	SP_PRD_EQ_THD							

Bit	Name	Function
7-0	SP_PRD_EQ_THD	**H active detect control** How many continue legal line as valid

SYNC_PROC 09 REG S5_2D, R/W

Bit	7	6	5	4	3	2	1	0
	SP_VSYNC_TGL_THD							

Bit	Name	Function
7-0	SP_VSYNC_TGL_THD	**V active detect control** V sync toggle times threshold

SYNC_PROC 10 REG S5_2E, R/W

Bit	7	6	5	4	3	2	1	0
	SP_SYNC_WIDTH_DTHD							

Bit	Name	Function
7-0	SP_SYNC_WIDTH_DTHD	**V active detect control** V sync pulse width threshod

SYNC_PROC 11 REG S5_2F, R/W

Bit	7	6	5	4	3	2	1	0
	SP_V_PRD_EQ_THD							

Bit	Name	Function
7-0	SP_V_PRD_EQ_THD	**V active detect control** How many continue legal v sync as valid

SYNC_PROC 12 REG S5_31, R/W

Bit	7	6	5	4	3	2	1	0
	SP_VT_DLT_REG							

Bit	Name	Function
7-0	SP_VT_DLT_REG	**v active detect control** V total different threshold

SYNC_PROC 13 REG S5_32, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED							SP_VSIN_INV_REG

Bit	Name	Function
0	SP_VSIN_INV_REG	**V active detect control** Input v sync invert to v active detect
7-1	RESERVED	

SYNC_PROC 14 REG S5_33, R/W

Bit	7	6	5	4	3	2	1	0
	SP_H_TIMER_VAL							

Bit	Name	Function
7-0	SP_H_TIMER_VAL	**Timer value control** H timer value for h detect

SYNC_PROC 15 REG S5_34, R/W

Bit	7	6	5	4	3	2	1	0
	SP_V_TIMER_VAL							

Bit	Name	Function
7-0	SP_V_TIMER_VAL	**Timer value control** V timer for V detect

SYNC_PROC 16 REG S5_35, R/W

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

Bit	Name	Function
7-0	SP_DLT_REG [7:0]	**Sync separation control** Sync pulse width difference threshold

SYNC_PROC 17 REG S5_36, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED				SP_DLT_REG [11:8]			

Bit	Name	Function
3-0	SP_DLT_REG [11:8]	**Sync separate control** MSB for sync pulse width difference compare value
7-4	RESERVED	

SYNC_PROC 18 REG S5_37, R/W

Bit	7	6	5	4	3	2	1	0
	SP_H_PULSE_IGNOR							

Bit	Name	Function
7-0	SP_H_PULSE_IGNOR	**Sync separation control** H pulse less than this value will be ignore this counter is start when sync large different

SYNC_PROC 19 REG S5_38, R/W

Bit	7	6	5	4	3	2	1	0
	SP_PRE_COAST							

Bit	Name	Function
7-0	SP_PRE_COAST	**Sync separation control** Set the coast will valid before vertical sync (line number)

SYNC_PROC 20 REG S5_39, R/W

Bit	7	6	5	4	3	2	1	0
	SP_POST_COAST							

Bit	Name	Function
7-0	SP_POST_COAST	**Sync separation control** When line cnt reach this value coast goes down

SYNC_PROC 21 REG S5_3A, R/W

Bit	7	6	5	4	3	2	1	0
	SP_H_TOTAL_EQ_THD							

Bit	Name	Function
7-0	SP_H_TOTAL_EQ_THD	**Sync separation control** How many regular line regard it as legal

SYNC_PROC 22 REG S5_3B, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED	SP_SDCS_VSSP_REG_H			RESERVED	SP_SDCS_VSST_REG_H		

Bit	Name	Function
2-0	SP_SDCS_VSST_REG_H	**Sync separation control** High bit of SD vs. start position
3	RESERVED	
6-4	SP_SDCS_VSSP_REG_H	**Sync separation control** High bit of SD vs. stop position
7	RESERVED	

SYNC_PROC 23 REG S5_3E, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED		SP_DIS_SUB_COAST	SP_H_PROTECT	SP_CS_INV_REG	SP_H_COAST	SP_HD_MODE	SP_CS_P_SWAP

Bit	Name	Function
0	SP_CS_P_SWAP	**Sync separation control** cs_p_swap cs edge reference select default rising edge
1	SP_HD_MODE	hd_mode 1: HD mode 0: SD mode
2	SP_H_COAST	h_coast 1: with sub coast out
3	SP_CS_INV_REG	cs_inv_reg cs input invert
4	SP_H_PROTECT	H count overflow protect
5	SP_DIS_SUB_COAST	Disable sub coast
7-6	RESERVED	

SYNC_PROC 24 REG S5_3F, R/W

Bit	7	6	5	4	3	2	1	0
	SP_SDCS_VSST_REG_L							

Bit	Name	Function
7-0	SP_SDCS_VSST_REG_L	**Sync separation control** SD vs. start position

SYNC_PROC 25 REG S5_40, R/W

Bit	7	6	5	4	3	2	1	0
	SP_SDCS_VSSP_REG_L							

Bit	Name	Function
7-0	SP_SDCS_VSSP_REG_L	**Sync separation control** SD vs. stop position

SYNC_PROC 26 REG S5_41, R/W

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

Bit	Name	Function
7-0	SP_CS_CLP_ST [7:0]	**Sync separation control** SOG clamp start position

SYNC_PROC 27 REG S5_42, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED				SP_CS_CLP_ST [11:8]			

Bit	Name	Function
3-0	SP_CS_CLP_ST [11:8]	**Sync separation control** SOG clamp start position MSB
7-4	RESERVED	

SYNC_PROC 28 REG S5_43, R/W

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

Bit	Name	Function
7-0	SP_CS_CLP_SP [7:0]	**Sync separation control** SOG clamp stop position

SYNC_PROC 29 REG S5_44, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED				SP_CS_CLP_SP [11:8]			

Bit	Name	Function
3-0	SP_CS_CLP_SP [11:8]	**Sync separation control** SOG clamp stop position MSB
7-4	RESERVED	

SYNC_PROC 30 REG S5_45, R/W

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

Bit	Name	Function
7-0	SP_CS_HS_ST [7:0]	**Sync separation control** If the horizontal period number is equal to the defined value, in XGA modes, It's XGA 75Hz mode.

SYNC_PROC 31 REG S5_46, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED				SP_CS_HS_ST [11:8]			

Bit	Name	Function
3-0	SP_CS_HS_ST [11:8]	**Sync separation control** SOG HS start position MSB
7-4	RESERVED	

SYNC_PROC 32 REG S5_47, R/W

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

Bit	Name	Function
7-0	SP_CS_HS_SP [7:0]	**Sync separation control** SOG hs stop position

SYNC_PROC 34 REG S5_48, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED				SP_CS_HS_SP [11:8]			

Bit	Name	Function
3-0	SP_CS_HS_SP [11:8]	**Sync separation control** SOG hs stop position MSB
7-4	RESERVED	

SYNC_PROC 35 REG S5_49, R/W

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

Bit	Name	Function
7-0	SP_RT_HS_ST [7:0]	**Retiming control** Retiming hs start position

SYNC_PROC 36 REG S5_4A, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED				SP_RT_HS_ST [11:8]			

Bit	Name	Function
3-0	SP_RT_HS_ST [11:8]	**Retiming control** Retiming hs start position MSB
7-4	RESERVED	

SYNC_PROC 37 REG S5_4B, R/W

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

Bit	Name	Function
7-0	SP_RT_HS_SP [7:0]	**Retiming control** Retiming hs stop postion

SYNC_PROC 38 REG S5_4C, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED				SP_RT_HS_SP [11:8]			

Bit	Name	Function
3-0	SP_RT_HS_SP [11:8]	**Retiming control** Retiming hs stop position MSB.
7-4	RESERVED	

SYNC_PROC 39 REG S5_4D, R/W

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

Bit	Name	Function
7-0	SP_H_CST_ST [7:0]	**Retiming control** H coast start position (total-this value)

SYNC_PROC 40 REG S5_4E, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED				SP_H_CST_ST [11:8]			

Bit	Name	Function
3-0	SP_H_CST_ST [11:8]	**Retiming control** H coast start position (total-this value)
7-4	RESERVED	

SYNC_PROC 41 REG S5_4F, R/W

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

Bit	Name	Function
7-0	SP_H_CST_SP [7:0]	**Retiming control** H coast stop position

SYNC_PROC 42 REG S5_50, R/W

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

Bit	Name	Function
3-0	SP_H_CST_SP [11:8]	**Retiming control** H coast stop position MSB
7-4	RESERVED	

SYNC_PROC 43 REG S5_51, R/W

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

Bit	Name	Function
7-0	SP_RT_VS_ST [7:0]	**Retiming control** Retiming vs start position

SYNC_PROC 44 REG S5_52, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED				SP_RT_VS_ST [11:8]			

Bit	Name	Function
3-0	SP_RT_VS_ST [11:8]	**Retiming control** Retiming vs start position MSB
7-4	RESERVED	

SYNC_PROC 45 REG S5_53, R/W

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

Bit	Name	Function
7-0	SP_RT_VS_SP [7:0]	**Retiming control** Retiming vs stop position

SYNC_PROC 46 REG S5_54, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED				SP_RT_VS_SP [11:8]			

Bit	Name	Function
2-0	SP_RT_VS_SP [11:8]	**Retiming control** Retiming vs stop position MSB
7-3	RESERVED	

SYNC_PROC 47 REG S5_55, R/W

Bit	7	6	5	4	3	2	1	0
	SP_HCST_AUTO_EN	SP_VS_POL_ATO	SP_VS_INV_REG	SP_HS_POL_ATO	SP_HS_INV_REG	SP_HS_EP_DLY_SEL		

Bit	Name	Function
2-0	SP_HS_EP_DLY_SEL	**Retiming control** Hs pulse delay sel for (sync with vs)
3	SP_HS_INV_REG	**Retiming control** hs_inv_reg inver hs to retimming module
4	SP_HS_POL_ATO	**Retiming control** hs auto correct in retiming module.
5	SP_VS_INV_REG	**Retiming control** vs inv_reg invert hs to retiming module
6	SP_VS_POL_ATO	**Retiming control** vs auto correct in retiming module
7	SP_HCST_AUTO_EN	**Retiming control** If enable h coast will start at (V total - hcst_st)

SYNC_PROC 48 REG S5_56, R/W

Bit	7	6	5	4	3	2	1	0
	SP_CLAMP_INV_REG	SP_VS_PROC_INV_REG	SP_HS_PROC_INV_REG	SP_SYNC_BYPS	SP_CLP_SRC_SEL	SP_CLAMP_MANUAL	SP_HS2PLL_INV_REG	SP_SOG_MODE

Bit	Name	Function
0	SP_SOG_MODE	**Out control** 1: SOG mode; 0: normal mode
1	SP_HS2PLL_INV_REG	**Out control** When =1, HS to PLL invert
2	SP_CLAMP_MANUAL	**Out control** 1: clamp turn on off by control by software (default) 0: for test
3	SP_CLP_SRC_SEL	**Out control** Clamp source select 1: pixel clock generate 0: 27Mhz clock generate
4	SP_SYNC_BYPS	**Out control** External sync bypass to decimator
5	SP_HS_PROC_INV_REG	**Out control** HS to decimator invert
6	SP_VS_PROC_INV_REG	**Out control** VS to decimator invert
7	SP_CLAMP_INV_REG	**Out control** Clamp to ADC invert

SYNC_PROC 49 REG S5_57, R/W

Bit	7	6	5	4	3	2	1	0
	SP_HS_REG	SP_HS_LOOP_SEL	RESERVED		SP_COAST_VALUE_REG	SP_NO_COAST_REG	SP_COAST_INV_REG	SP_NO_CLAMP_REG

Bit	Name	Function
0	SP_NO_CLAMP_REG	**Out control** Clamp always be 0
1	SP_COAST_INV_REG	**Out control** Coast invert
2	SP_NO_COAST_REG	**Out control** Coast always be REG S5_57[3]
3	SP_COAST_VALUE_REG	**Out control** Coast use 1x clk generate
5-4	RESERVED	**Out control**
6	SP_HS_LOOP_SEL	**Out control** Bypass PLL HS to 57 core
7	SP_HS_REG	**Out control** When sub_coast enable will select this value

SYNC_PROC 50 REG S5_58, R/W

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

Bit	Name	Function
7-0	SP_HT_DIFF_REG [7:0]	**Auto clamp control** H total difference less this value as valid for auto clamp enable control

SYNC_PROC 51 REG S5_59, R/W

Bit	7	6	5	4	3	2	1	0
	RESVERD				SP_HT_DIFF_REG [11:8]			

Bit	Name	Function
3-0	SP_HT_DIFF_REG [11:8]	**Auto clamp control** H total difference less this value as valid for auto clamp enable control
7-4	RESERVED	

SYNC_PROC 52 REG S5_5A, R/W

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

Bit	Name	Function
7-0	SP_VT_DIFF_REG [7:0]	**Auto clamp control** V total difference less this value as valid for auto clamp enable control

SYNC_PROC 53 REG S5_5B, R/W

Bit	7	6	5	4	3	2	1	0
	RESVERD				SP_VT_DIFF_REG [10:8]			

Bit	Name	Function
2-0	SP_VT_DIFF_REG [10:8]	**Auto clamp control** V total difference less this value as valid for auto clamp enable control
7-4	RESERVED	

SYNC_PROC 54 REG S5_5C, R/W

Bit	7	6	5	4	3	2	1	0
	SP_STBLE_CNT_REG							

Bit	Name	Function
7-0	SP_STBLE_CNT_REG	**Auto clamp control** Stable indicate frame threshold for auto clamp enable control

SYNC_PROC 55 REG S5_63, R/W

Bit	7	6	5	4	3	2	1	0
	RESERVED	SP_TEST_SIGNAL_SEL			SP_TEST_MODULE			SP_TEST_EN

Bit	Name	Function
0	SP_TEST_EN	**Test control**
		Test bus enable
3-1	SP_TEST_MODULE	**Test control**
		test module select # 0 none # 1 hs_pol_det module # 2 hs_act_det module # 3 vs_pol_det module # 4 vs_act_det module # 5 cs_sep module # 6 retiming module # 7 out proc module
6-4	SP_TEST_SIGNAL_SEL	**Test control**
		Test signal select
7	RESERVED	