Radeon R3xx 3D Register Reference Guide

Trademarks

AMD, the AMD Arrow logo, Athlon, and combinations thereof, ATI, ATI logo, Radeon, and Crossfire are trademarks of Advanced Micro Devices, Inc.

Microsoft and Windows are registered trademarks of Microsoft Corporation.

Other product names used in this publication are for identification purposes only and may be trademarks of their respective companies.

Disclaimer

The contents of this document are provided in connection with Advanced Micro Devices, Inc. ("AMD") products. AMD makes no representations or warranties with respect to the accuracy or completeness of the contents of this publication and reserves the right to make changes to specifications and product descriptions at any time without notice. No license, whether express, implied, arising by estoppel, or otherwise, to any intellectual property rights are granted by this publication. Except as set forth in AMD's Standard Terms and Conditions of Sale, AMD assumes no liability whatsoever, and disclaims any express or implied warranty, relating to its products including, but not limited to, the implied warranty of merchantability, fitness for a particular purpose, or infringement of any intellectual property right. AMD's products are not designed, intended, authorized or warranted for use as components in systems intended for surgical implant into the body, or in other applications intended to support or sustain life, or in any other application in which the failure of AMD's product could create a situation where personal injury, death, or severe property or environmental damage may occur. AMD reserves the right to discontinue or make changes to its products at any time without notice.

1. Registers

1.1 Color Buffer Registers

CB:RB3D_AARESOLVE_CTL · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4e88			
DESCRIPTION: *Resolve Buffer Control. Unpipelined*			
Field Name	Bits	Default	Description
AARESOLVE_MODE	0	0x0	Specifies if the color buffer is in resolve mode. The cache must be empty before changing this register. POSSIBLE VALUES: 00 - Normal operation. 01 - Resolve operation.
AARESOLVE_GAMMA	1	none	Specifies the gamma and degamma to be applied to the samples before and after filtering, respectively. POSSIBLE VALUES: 00 - 1.0 01 - 2.2

CB:RB3D_AARESOLVE_OFFSET · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4e80			
DESCRIPTION: *Resolve buffer destination address. The cache must be empty before changing this register if the cb is in resolve mode. Unpipelined*			
Field Name	Bits	Default	Description
AARESOLVE_OFFSET	31:5	none	256-bit aligned 3D resolve destination offset.

CB:RB3D_AARESOLVE_PITCH · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4e84			
DESCRIPTION: *Resolve Buffer Pitch and Tiling Control. The cache must be empty before changing this register if the cb is in resolve mode. Unpipelined*			
Field Name	Bits	Default	Description
AARESOLVE_PITCH	13:1	none	3D destination pitch in multiples of 2-pixels.

CB:RB3D_ABLENDCNTL · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4e08			
DESCRIPTION: *Alpha Blend Control for Alpha Channel. Pipelined through the blender.*			
Field Name	Bits	Default	Description
COMB_FCN	14:12	none	Combine Function , Allows modification of how the SRCBLEND and DESTBLEND are combined. POSSIBLE VALUES: 00 - Add and Clamp 01 - Add but no Clamp

			02 - Subtract Dst from Src, and Clamp 03 - Subtract Dst from Src, and don`t Clamp 04 - Minimum of Src, Dst (the src and dst blend functions are forced to D3D_ONE) 05 - Maximum of Src, Dst (the src and dst blend functions are forced to D3D_ONE) 06 - Subtract Src from Dst, and Clamp 07 - Subtract Src from Dst, and don`t Clamp
SRCBLEND	21:16	none	Source Blend Function , Alpha blending function (SRC). POSSIBLE VALUES: 00 - RESERVED 01 - D3D_ZERO 02 - D3D_ONE 03 - D3D_SRCCOLOR 04 - D3D_INVSRCCOLOR 05 - D3D_SRCALPHA 06 - D3D_INVSRCALPHA 07 - D3D_DESTALPHA 08 - D3D_INVDESTALPHA 09 - D3D_DESTCOLOR 10 - D3D_INVDESTCOLOR 11 - D3D_SRCALPHASAT 12 - D3D_BOTHSRCALPHA 13 - D3D_BOTHINVSRCALPHA 14 - RESERVED 15 - RESERVED 16 - RESERVED 17 - RESERVED 18 - RESERVED 19 - RESERVED 20 - RESERVED 21 - RESERVED 22 - RESERVED 23 - RESERVED 24 - RESERVED 25 - RESERVED 26 - RESERVED 27 - RESERVED 28 - RESERVED 29 - RESERVED 30 - RESERVED 31 - RESERVED 32 - GL_ZERO 33 - GL_ONE 34 - GL_SRC_COLOR 35 - GL_ONE_MINUS_SRC_COLOR 36 - GL_DST_COLOR 37 - GL_ONE_MINUS_DST_COLOR 38 - GL_SRC_ALPHA 39 - GL_ONE_MINUS_SRC_ALPHA 40 - GL_DST_ALPHA 41 - GL_ONE_MINUS_DST_ALPHA

			42 - GL_SRC_ALPHA_SATURATE 43 - GL_CONSTANT_COLOR 44 - GL_ONE_MINUS_CONSTANT_COLOR 45 - GL_CONSTANT_ALPHA 46 - GL_ONE_MINUS_CONSTANT_ALPHA 47 - RESERVED 48 - RESERVED 49 - RESERVED 50 - RESERVED 51 - RESERVED 52 - RESERVED 53 - RESERVED 54 - RESERVED 55 - RESERVED 56 - RESERVED 57 - RESERVED 58 - RESERVED 59 - RESERVED 60 - RESERVED 61 - RESERVED 62 - RESERVED 63 - RESERVED
DESTBLEND	29:24	none	Destination Blend Function , Alpha blending function (DST). POSSIBLE VALUES: 00 - RESERVED 01 - D3D_ZERO 02 - D3D_ONE 03 - D3D_SRCCOLOR 04 - D3D_INVSRCCOLOR 05 - D3D_SRCALPHA 06 - D3D_INVSRCALPHA 07 - D3D_DESTALPHA 08 - D3D_INVDESTALPHA 09 - D3D_DESTCOLOR 10 - D3D_INVDESTCOLOR 11 - RESERVED 12 - RESERVED 13 - RESERVED 14 - RESERVED 15 - RESERVED 16 - RESERVED 17 - RESERVED 18 - RESERVED 19 - RESERVED 20 - RESERVED 21 - RESERVED 22 - RESERVED 23 - RESERVED 24 - RESERVED 25 - RESERVED 26 - RESERVED

			27 - RESERVED 28 - RESERVED 29 - RESERVED 30 - RESERVED 31 - RESERVED 32 - GL_ZERO 33 - GL_ONE 34 - GL_SRC_COLOR 35 - GL_ONE_MINUS_SRC_COLOR 36 - GL_DST_COLOR 37 - GL_ONE_MINUS_DST_COLOR 38 - GL_SRC_ALPHA 39 - GL_ONE_MINUS_SRC_ALPHA 40 - GL_DST_ALPHA 41 - GL_ONE_MINUS_DST_ALPHA 42 - RESERVED 43 - GL_CONSTANT_COLOR 44 - GL_ONE_MINUS_CONSTANT_COLOR 45 - GL_CONSTANT_ALPHA 46 - GL_ONE_MINUS_CONSTANT_ALPHA 47 - RESERVED 48 - RESERVED 49 - RESERVED 50 - RESERVED 51 - RESERVED 52 - RESERVED 53 - RESERVED 54 - RESERVED 55 - RESERVED 56 - RESERVED 57 - RESERVED 58 - RESERVED 59 - RESERVED 60 - RESERVED 61 - RESERVED 62 - RESERVED 63 - RESERVED

CB:RB3D_BLENDCNTL · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4e04			
DESCRIPTION: *Alpha Blend Control for Color Channels. Pipelined through the blender.*			
Field Name	Bits	Default	Description
ALPHA_BLEND_ENABLE	0	0x0	Allow alpha blending with the destination. POSSIBLE VALUES: 00 - Disable 01 - Enable
SEPARATE_ALPHA_ENABLE	1	0x0	Enables use of RB3D_ABLENDCNTL POSSIBLE VALUES: 00 - Disabled (Use RB3D_BLENDCNTL)

			01 - Enabled (Use RB3D_ABLENDCNTL)
READ_ENABLE	2	0x1	When blending is enabled, this enables memory reads. Memory reads will still occur when this is disabled if they are for reasons not related to blending. POSSIBLE VALUES: 00 - Disable reads 01 - Enable reads
DISCARD_SRC_PIXELS	5:3	0x0	Discard pixels when blending is enabled based on the src color. POSSIBLE VALUES: 00 - Disable 01 - Discard pixels if src alpha == 0 02 - Discard pixels if src color == 0 03 - Discard pixels if (src alpha == 0) && (src color == 0) 04 - Discard pixels if src alpha == 1 05 - Discard pixels if src color == 1 06 - Discard pixels if (src alpha == 1) && (src color == 1) 07 - (reserved)
COMB_FCN	14:12	none	Combine Function , Allows modification of how the SRCBLEND and DESTBLEND are combined. POSSIBLE VALUES: 00 - Add and Clamp 01 - Add but no Clamp 02 - Subtract Dst from Src, and Clamp 03 - Subtract Dst from Src, and don`t Clamp 04 - Minimum of Src, Dst (the src and dst blend functions are forced to D3D_ONE) 05 - Maximum of Src, Dst (the src and dst blend functions are forced to D3D_ONE) 06 - Subtract Src from Dst, and Clamp 07 - Subtract Src from Dst, and don`t Clamp
SRCBLEND	21:16	none	Source Blend Function , Alpha blending function (SRC). POSSIBLE VALUES: 00 - RESERVED 01 - D3D_ZERO 02 - D3D_ONE 03 - D3D_SRCCOLOR 04 - D3D_INVSRCCOLOR 05 - D3D_SRCALPHA 06 - D3D_INVSRCALPHA 07 - D3D_DESTALPHA 08 - D3D_INVDESTALPHA 09 - D3D_DESTCOLOR 10 - D3D_INVDESTCOLOR 11 - D3D_SRCALPHASAT

			12 - D3D_BOTHSRCALPHA 13 - D3D_BOTHINVSRCALPHA 14 - RESERVED 15 - RESERVED 16 - RESERVED 17 - RESERVED 18 - RESERVED 19 - RESERVED 20 - RESERVED 21 - RESERVED 22 - RESERVED 23 - RESERVED 24 - RESERVED 25 - RESERVED 26 - RESERVED 27 - RESERVED 28 - RESERVED 29 - RESERVED 30 - RESERVED 31 - RESERVED 32 - GL_ZERO 33 - GL_ONE 34 - GL_SRC_COLOR 35 - GL_ONE_MINUS_SRC_COLOR 36 - GL_DST_COLOR 37 - GL_ONE_MINUS_DST_COLOR 38 - GL_SRC_ALPHA 39 - GL_ONE_MINUS_SRC_ALPHA 40 - GL_DST_ALPHA 41 - GL_ONE_MINUS_DST_ALPHA 42 - GL_SRC_ALPHA_SATURATE 43 - GL_CONSTANT_COLOR 44 - GL_ONE_MINUS_CONSTANT_COLOR 45 - GL_CONSTANT_ALPHA 46 - GL_ONE_MINUS_CONSTANT_ALPHA 47 - RESERVED 48 - RESERVED 49 - RESERVED 50 - RESERVED 51 - RESERVED 52 - RESERVED 53 - RESERVED 54 - RESERVED 55 - RESERVED 56 - RESERVED 57 - RESERVED 58 - RESERVED 59 - RESERVED 60 - RESERVED 61 - RESERVED 62 - RESERVED 63 - RESERVED
DESTBLEND	29:24	none	Destination Blend Function , Alpha blending function

			(DST). POSSIBLE VALUES: 00 - RESERVED 01 - D3D_ZERO 02 - D3D_ONE 03 - D3D_SRCCOLOR 04 - D3D_INVSRCCOLOR 05 - D3D_SRCALPHA 06 - D3D_INVSRCALPHA 07 - D3D_DESTALPHA 08 - D3D_INVDESTALPHA 09 - D3D_DESTCOLOR 10 - D3D_INVDESTCOLOR 11 - RESERVED 12 - RESERVED 13 - RESERVED 14 - RESERVED 15 - RESERVED 16 - RESERVED 17 - RESERVED 18 - RESERVED 19 - RESERVED 20 - RESERVED 21 - RESERVED 22 - RESERVED 23 - RESERVED 24 - RESERVED 25 - RESERVED 26 - RESERVED 27 - RESERVED 28 - RESERVED 29 - RESERVED 30 - RESERVED 31 - RESERVED 32 - GL_ZERO 33 - GL_ONE 34 - GL_SRC_COLOR 35 - GL_ONE_MINUS_SRC_COLOR 36 - GL_DST_COLOR 37 - GL_ONE_MINUS_DST_COLOR 38 - GL_SRC_ALPHA 39 - GL_ONE_MINUS_SRC_ALPHA 40 - GL_DST_ALPHA 41 - GL_ONE_MINUS_DST_ALPHA 42 - RESERVED 43 - GL_CONSTANT_COLOR 44 - GL_ONE_MINUS_CONSTANT_COLOR 45 - GL_CONSTANT_ALPHA 46 - GL_ONE_MINUS_CONSTANT_ALPHA 47 - RESERVED 48 - RESERVED 49 - RESERVED

			50 - RESERVED 51 - RESERVED 52 - RESERVED 53 - RESERVED 54 - RESERVED 55 - RESERVED 56 - RESERVED 57 - RESERVED 58 - RESERVED 59 - RESERVED 60 - RESERVED 61 - RESERVED 62 - RESERVED 63 - RESERVED

CB:RB3D_CCTL · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4e00			
DESCRIPTION: *Unpipelined.*			
Field Name	Bits	Default	Description
NUM_MULTIWRITES	6:5	0x0	A quad is replicated and written to this many buffers. POSSIBLE VALUES: 00 - 1 buffer. This is the only mode where the cb processes the end of packet command. 01 - 2 buffers 02 - 3 buffers 03 - 4 buffers
CLRCMP_FLIPE_ENABLE	7	0x0	Enables equivalent of rage128 CMP_EQ_FLIP color compare mode. This is used to ensure 3D data does not get chromakeyed away by logic in the backend. POSSIBLE VALUES: 00 - Disable color compare. 01 - Enable color compare.
AA_COMPRESSION_ENABLE	9	none	Enables AA color compression. The cache must be empty before this is changed. POSSIBLE VALUES: 00 - Disable AA compression 01 - Enable AA compression
Reserved	10	none	Set to 0

CB:RB3D_CLRCMP_CLR · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4e20			
DESCRIPTION: *Color Compare Color. Stalls the 2d/3d datapath until it is idle.*			
Field Name	Bits	Default	Description
CLRCMP_CLR	31:0	none	Like RB2D_CLRCMP_CLR, but a separate register is provided to keep 2D and 3D state separate.

CB:RB3D_CLRCMP_FLIPE · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4e1c			
DESCRIPTION: *Color Compare Flip. Stalls the 2d/3d datapath until it is idle.*			
Field Name	Bits	Default	Description
CLRCMP_FLIPE	31:0	none	Like RB2D_CLRCMP_FLIPE, but a separate register is provided to keep 2D and 3D state separate.

CB:RB3D_CLRCMP_MSK · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4e24			
DESCRIPTION: *Color Compare Mask. Stalls the 2d/3d datapath until it is idle.*			
Field Name	Bits	Default	Description
CLRCMP_MSK	31:0	none	Like RB2D_CLRCMP_CLR, but separate registers provided to keep 2D and 3D state separate.

CB:RB3D_COLOROFFSET[0-3] · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4e28-0x4e34			
DESCRIPTION: *Color Buffer Address Offset of multibuffer 0. Unpipelined.*			
Field Name	Bits	Default	Description
COLOROFFSET	31:5	none	256-bit aligned 3D destination offset address. The cache must be empty before this is changed.

CB:RB3D_COLORPITCH[0-3] · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4e38-0x4e44			
DESCRIPTION: *Color buffer format and tiling control for all the multibuffers and the pitch of multibuffer 0. Unpipelined. The cache must be empty before any of the registers are changed.*			
Field Name	Bits	Default	Description
COLORPITCH	13:1	none	3D destination pitch in multiples of 2-pixels.
COLORTILE	16	none	Denotes whether the 3D destination is in macrotiled format. POSSIBLE VALUES: 00 - 3D destination is not macrotiled 01 - 3D destination is macrotiled
COLORMICROTILE	18:17	none	Denotes whether the 3D destination is in microtiled format. POSSIBLE VALUES: 00 - 3D destination is no microtiled 01 - 3D destination is microtiled 02 - 3D destination is square microtiled. Only available in 16-bit 03 - (reserved)
COLORENDIAN	20:19	none	Specifies endian control for the color buffer.

			POSSIBLE VALUES: 00 - No swap 01 - Word swap (2 bytes in 16-bit) 02 - Dword swap (4 bytes in a 32-bit) 03 - Half-Dword swap (2 16-bit in a 32-bit)
COLORFORMAT	24:21	0x6	3D destination color format. POSSIBLE VALUES: 00 - (Reserved) 01 - (Reserved) 02 - (Reserved) 03 - ARGB1555 04 - RGB565 05 - (Reserved) 06 - ARGB8888 07 - ARGB32323232 08 - (Reserved) 09 - I8 10 - ARGB16161616 11 - YUV422 packed (VYUY) 12 - YUV422 packed (YVYU) 13 - UV88 14 - (reserved) 15 - ARGB4444

CB:RB3D_COLOR_CHANNEL_MASK · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4e0c			
DESCRIPTION: *3D Color Channel Mask. If all the channels used in the current color format are disabled, then the cb will discard all the incoming quads. Pipelined through the blender.*			
Field Name	Bits	Default	Description
BLUE_MASK	0	0x1	mask bit for blue channel POSSIBLE VALUES: 00 - disable 01 - enable
GREEN_MASK	1	0x1	mask bit for green channel POSSIBLE VALUES: 00 - disable 01 - enable
RED_MASK	2	0x1	mask bit for red channel POSSIBLE VALUES: 00 - disable 01 - enable
ALPHA_MASK	3	0x1	mask bit for alpha channel POSSIBLE VALUES:

			00 - disable 01 - enable

CB:RB3D_COLOR_CLEAR_VALUE · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4e14			
DESCRIPTION: *Clear color that is used when the color mask is set to 00. Unpipelined.*			
Field Name	Bits	Default	Description
BLUE	7:0	none	
GREEN	15:8	none	
RED	23:16	none	
ALPHA	31:24	none	

CB:RB3D_CONSTANT_COLOR · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4e10			
DESCRIPTION: *Constant color used by the blender. Pipelined through the blender.*			
Field Name	Bits	Default	Description
BLUE	7:0	none	blue constant color
GREEN	15:8	none	green constant color
RED	23:16	none	red constant color
ALPHA	31:24	none	alpha constant color

CB:RB3D_DITHER_CTL · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4e50			
DESCRIPTION: *Dithering control register. Pipelined through the blender.*			
Field Name	Bits	Default	Description
DITHER_MODE	1:0	0x0	Dither mode POSSIBLE VALUES: 00 - Truncate 01 - Round 02 - LUT dither 03 - (reserved)
ALPHA_DITHER_MODE	3:2	0x0	POSSIBLE VALUES: 00 - Truncate 01 - Round 02 - LUT dither 03 - (reserved)

CB:RB3D_DSTCACHE_CTLSTAT · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4e4c
DESCRIPTION: *Destination Color Buffer Cache Control/Status. If the cb is in e2 mode, then a flush or free will not occur upon a write to this register, but a sync will be immediately sent if one is requested. If both DC_FLUSH and DC_FREE are zero but DC_FINISH is one, then a sync will be sent immediately -- the cb will not wait for all the previous operations to complete before sending the sync. Unpipelined except when DC_FINISH and DC_FREE are both set to zero.*

Field Name	Bits	Default	Description
DC_FLUSH	1:0	0x0	Setting this bit flushes dirty data from the 3D Dst Cache. Unless the DC_FREE bits are also set, the tags in the cache remain valid. A purge is achieved by setting both DC_FLUSH and DC_FREE. POSSIBLE VALUES: 00 - No effect 01 - No effect 02 - Flushes dirty 3D data 03 - Flushes dirty 3D data
DC_FREE	3:2	0x0	Setting this bit invalidates the 3D Dst Cache tags. Unless the DC_FLUSH bit is also set, the cache lines are not written to memory. A purge is achieved by setting both DC_FLUSH and DC_FREE. POSSIBLE VALUES: 00 - No effect 01 - No effect 02 - Free 3D tags 03 - Free 3D tags
DC_FINISH	4	0x0	POSSIBLE VALUES: 00 - do not send a finish signal to the CP 01 - send a finish signal to the CP after the end of operation

CB:RB3D_ROPCNTL · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4e18			
DESCRIPTION: *3D ROP Control. Stalls the 2d/3d datapath until it is idle.*			
Field Name	Bits	Default	Description
ROP_ENABLE	2	0x0	POSSIBLE VALUES: 00 - Disable ROP. (Forces ROP2 to be 0xC). 01 - Enabled
ROP	11:8	none	ROP2 code for 3D fragments. This value is replicated into 2 nibbles to form the equivalent ROP3 code to control the ROP3 logic. These are the GDI ROP2 codes.

1.2 Fog Registers

FG:FG_ALPHA_FUNC · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4bd4			
DESCRIPTION: *Alpha Function*			
Field Name	Bits	Default	Description
AF_VAL	7:0	0x0	Specifies the alpha compare value.
AF_FUNC	10:8	0x0	Specifies the alpha compare function. POSSIBLE VALUES: 00 - AF_NEVER 01 - AF_LESS 02 - AF_EQUAL 03 - AF_LE 04 - AF_GREATER 05 - AF_NOTEQUAL 06 - AF_GE 07 - AF_ALWAYS
AF_EN	11	0x0	Enables/Disables alpha compare function. POSSIBLE VALUES: 00 - Disable alpha function. 01 - Enable alpha function.
AM_EN	16	0x0	Enables/Disables alpha-to-mask function. POSSIBLE VALUES: 00 - Disable alpha to mask function. 01 - Enable alpha to mask function.
AM_CFG	17	0x0	Specfies number of sub-pixel samples for alpha-to-mask function. POSSIBLE VALUES: 00 - 2/4 sub-pixel samples. 01 - 3/6 sub-pixel samples.
DITH_EN	20	0x0	Enables/Disables RGB Dithering. POSSIBLE VALUES: 00 - Disable Dithering 01 - Enable Dithering.

FG:FG_DEPTH_SRC · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4bd8			
DESCRIPTION: *Where does depth come from?*			
Field Name	Bits	Default	Description
DEPTH_SRC	0	0x0	POSSIBLE VALUES: 00 - Depth comes from scan converter as plane equation. 01 - Depth comes from shader as four discrete values.

FG:FG_FOG_BLEND · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4bc0			
DESCRIPTION: *Fog Blending Enable*			
Field Name	Bits	Default	Description
ENABLE	0	0x0	Enable for fog blending POSSIBLE VALUES: 00 - Disables fog (output matches input color). 01 - Enables fog.
FN	2:1	0x0	Fog generation function POSSIBLE VALUES: 00 - Fog function is linear 01 - Fog function is exponential 02 - Fog function is exponential squared 03 - Fog is derived from constant fog factor

FG:FG_FOG_COLOR_B · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4bd0			
DESCRIPTION: *Blue Component of Fog Color*			
Field Name	Bits	Default	Description
BLUE	9:0	0x0	Blue component of fog color; (0.9) fixed format.

FG:FG_FOG_COLOR_G · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4bcc			
DESCRIPTION: *Green Component of Fog Color*			
Field Name	Bits	Default	Description
GREEN	9:0	0x0	Green component of fog color; (0.9) fixed format.

FG:FG_FOG_COLOR_R · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4bc8			
DESCRIPTION: *Red Component of Fog Color*			
Field Name	Bits	Default	Description
RED	9:0	0x0	Red component of fog color; (0.9) fixed format.

FG:FG_FOG_FACTOR · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4bc4			
DESCRIPTION: *Constant Factor for Fog Blending*			
Field Name	Bits	Default	Description
FACTOR	9:0	0x0	Constant fog factor; fixed (0.9) format.

1.3 Geometry Assembly Registers

GA:GA_COLOR_CONTROL · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4278			
DESCRIPTION: *Specifies per RGB or Alpha shading method.*			
Field Name	Bits	Default	Description
RGB0_SHADING	1:0	0x0	Specifies solid, flat or Gouraud shading. POSSIBLE VALUES: 00 - Solid fill color 01 - Flat shading 02 - Gouraud shading
ALPHA0_SHADING	3:2	0x0	Specifies solid, flat or Gouraud shading. POSSIBLE VALUES: 00 - Solid fill color 01 - Flat shading 02 - Gouraud shading
RGB1_SHADING	5:4	0x0	Specifies solid, flat or Gouraud shading. POSSIBLE VALUES: 00 - Solid fill color 01 - Flat shading 02 - Gouraud shading
ALPHA1_SHADING	7:6	0x0	Specifies solid, flat or Gouraud shading. POSSIBLE VALUES: 00 - Solid fill color 01 - Flat shading 02 - Gouraud shading
RGB2_SHADING	9:8	0x0	Specifies solid, flat or Gouraud shading. POSSIBLE VALUES: 00 - Solid fill color 01 - Flat shading 02 - Gouraud shading
ALPHA2_SHADING	11:10	0x0	Specifies solid, flat or Gouraud shading. POSSIBLE VALUES: 00 - Solid fill color 01 - Flat shading 02 - Gouraud shading
RGB3_SHADING	13:12	0x0	Specifies solid, flat or Gouraud shading. POSSIBLE VALUES: 00 - Solid fill color 01 - Flat shading 02 - Gouraud shading
ALPHA3_SHADING	15:14	0x0	Specifies solid, flat or Gouraud shading.

			POSSIBLE VALUES: 00 - Solid fill color 01 - Flat shading 02 - Gouraud shading
PROVOKING_VERTEX	17:16	0x0	Specifies, for flat shaded polygons, which vertex holds the polygon color. POSSIBLE VALUES: 00 - Provoking is first vertex 01 - Provoking is second vertex 02 - Provoking is third vertex 03 - Provoking is always last vertex

GA:GA_ENHANCE · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4274			
DESCRIPTION: *GA Enhancement Register*			
Field Name	Bits	Default	Description
DEADLOCK_CNTL	0	0x0	TCL/GA Deadlock control. POSSIBLE VALUES: 00 - No effect. 01 - Prevents TCL interface from deadlocking on GA side.
FASTSYNC_CNTL	1	0x0	Enables Fast register/primitive switching POSSIBLE VALUES: 00 - No effect. 01 - Enables high-performance register/primitive switching.

GA:GA_FOG_OFFSET · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4298			
DESCRIPTION: *Specifies the offset to apply to fog.*			
Field Name	Bits	Default	Description
VALUE	31:0	0x0	32b SPFP scale value.

GA:GA_FOG_SCALE · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4294			
DESCRIPTION: *Specifies the scale to apply to fog.*			
Field Name	Bits	Default	Description
VALUE	31:0	0x0	32b SPFP scale value.

GA:GA_LINE_CNTL · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4234
DESCRIPTION: *Line control*

Field Name	Bits	Default	Description
WIDTH	15:0	0x0	1/2 width of line, in subpixels; (16.0) fixed format.
END_TYPE	17:16	0x0	Specifies how ends of lines should be drawn. POSSIBLE VALUES: 00 - Horizontal 01 - Vertical 02 - Square (horizontal or vertical depending upon slope) 03 - Computed (perpendicular to slope)

GA:GA_LINE_S0 · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4264			
DESCRIPTION: *S Texture Coordinate Value for Vertex 0 of Line (stuff textures -- i.e. AA)*			
Field Name	Bits	Default	Description
S0	31:0	0x0	S texture coordinate value generated for vertex 0 of an antialiased line; 32-bit IEEE float format. Typical 0.0.

GA:GA_LINE_S1 · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4268			
DESCRIPTION: *S Texture Coordinate Value for Vertex 1 of Lines (V2 of parallelogram -- stuff textures -- i.e. AA)*			
Field Name	Bits	Default	Description
S1	31:0	0x0	S texture coordinate value generated for vertex 1 of an antialiased line; 32-bit IEEE float format. Typical 1.0.

GA:GA_LINE_STIPPLE_CONFIG · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4238			
DESCRIPTION: *Line Stipple configuration information.*			
Field Name	Bits	Default	Description
LINE_RESET	1:0	0x0	Specify type of reset to use for stipple accumulation. POSSIBLE VALUES: 00 - No reseting 01 - Reset per line 02 - Reset per packet
STIPPLE_SCALE	31:2	0x0	Specifies, in truncated (30b) floating point, scale to apply to generated texture coordinates.

GA:GA_LINE_STIPPLE_VALUE · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4260			
DESCRIPTION: *Current value of stipple accumulator.*			
Field Name	Bits	Default	Description
STIPPLE_VALUE	31:0	0x0	24b Integer, measuring stipple accumulation in subpixels. (note: field is 32b, but only lower 24b used)

GA:GA_OFFSET · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4290			
DESCRIPTION: *Specifies x & y offsets for vertex data after conversion to FP.*			
Field Name	Bits	Default	Description
X_OFFSET	15:0	0x0	Specifies X offset in S15 format (subpixels).
Y_OFFSET	31:16	0x0	Specifies Y offset in S15 format (subpixels).

GA:GA_POINT_MINMAX · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4230			
DESCRIPTION: *Specifies maximum and minimum point & sprite sizes for per vertex size specification.*			
Field Name	Bits	Default	Description
MIN_SIZE	15:0	0x0	Minimum point & sprite radius (in subsamples) size to allow.
MAX_SIZE	31:16	0x0	Maximum point & sprite radius (in subsamples) size to allow.

GA:GA_POINT_S0 · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4200			
DESCRIPTION: *S Texture Coordinate of Vertex 0 for Point texture stuffing (LLC)*			
Field Name	Bits	Default	Description
S0	31:0	0x0	S texture coordinate of vertex 0 for point; 32-bit IEEE float format.

GA:GA_POINT_S1 · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4208			
DESCRIPTION: *S Texture Coordinate of Vertex 2 for Point texture stuffing (URC)*			
Field Name	Bits	Default	Description
S1	31:0	0x0	S texture coordinate of vertex 2 for point; 32-bit IEEE float format.

GA:GA_POINT_SIZE · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x421c			
DESCRIPTION: *Dimensions for Points*			
Field Name	Bits	Default	Description
HEIGHT	15:0	0x0	1/2 Height of point; fixed (16.0), subpixel format.
WIDTH	31:16	0x0	1/2 Width of point; fixed (16.0), subpixel format.

GA:GA_POINT_T0 · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4204			
DESCRIPTION: *T Texture Coordinate of Vertex 0 for Point texture stuffing (LLC)*			
Field Name	Bits	Default	Description
T0	31:0	0x0	T texture coordinate of vertex 0 for point; 32-bit IEEE float format.

GA:GA_POINT_T1 · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x420c			
DESCRIPTION: *T Texture Coordinate of Vertex 2 for Point texture stuffing (URC)*			
Field Name	Bits	Default	Description
T1	31:0	0x0	T texture coordinate of vertex 2 for point; 32-bit IEEE float format.

GA:GA_POLY_MODE · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4288			
DESCRIPTION: *Polygon Mode*			
Field Name	Bits	Default	Description
POLY_MODE	1:0	0x0	Polygon mode enable. POSSIBLE VALUES: 00 - Disable poly mode (render triangles). 01 - Dual mode (send 2 sets of 3 polys with specified poly type). 02 - Reserved
FRONT_PTYPE	6:4	0x0	Specifies how to render front-facing polygons. POSSIBLE VALUES: 00 - Draw points. 01 - Draw lines. 02 - Draw triangles. 03 - Reserved 3 - 7.
BACK_PTYPE	9:7	0x0	Specifies how to render back-facing polygons. POSSIBLE VALUES: 00 - Draw points. 01 - Draw lines. 02 - Draw triangles. 03 - Reserved 3 - 7.

GA:GA_ROUND_MODE · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x428c			
DESCRIPTION: *Specifies the rouding mode for geometry & color SPFP to FP conversions.*			
Field Name	Bits	Default	Description
GEOMETRY_ROUND	1:0	0x0	Trunc (0) or round to nearest (1) for geometry (XY). POSSIBLE VALUES: 00 - Round to trunc 01 - Round to nearest
COLOR_ROUND	3:2	0x0	Trunc (0) or round to nearest (1) for colors (RGBA). POSSIBLE VALUES: 00 - Round to trunc 01 - Round to nearest

RGB_CLAMP	4	0x0	Specifies SPFP color clamp range of [0,1] or [-8,8] for RGB. POSSIBLE VALUES: 00 - Clamp to [0,1.0] for RGB 01 - Clamp to [-7.9999, 7.9999] for RGB
ALPHA_CLAMP	5	0x0	Specifies SPFP alpha clamp range of [0,1] or [-8,8]. POSSIBLE VALUES: 00 - Clamp to [0,1.0] for Alpha 01 - Clamp to [-7.9999, 7.9999] for Alpha

GA:GA_SOFT_RESET · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x429c			
DESCRIPTION: *Specifies number of cycles to assert reset, and also causes RB3D soft reset to assert.*			
Field Name	Bits	Default	Description
SOFT_RESET_COUNT	15:0	0x0	Count in cycles (def 256).

GA:GA_SOLID_BA · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4280			
DESCRIPTION: *Specifies blue & alpha components of fill color.*			
Field Name	Bits	Default	Description
COLOR_ALPHA	15:0	0x0	Component alpha value. (S3.12)
COLOR_BLUE	31:16	0x0	Component blue value. (S3.12)

GA:GA_SOLID_RG · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x427c			
DESCRIPTION: *Specifies red & green components of fill color.*			
Field Name	Bits	Default	Description
COLOR_GREEN	15:0	0x0	Component green value (S3.12).
COLOR_RED	31:16	0x0	Component red value (S3.12).

GA:GA_TRIANGLE_STIPPLE · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4214			
DESCRIPTION: *Specifies amount to shift integer position of vertex (screen space) before converting to float for triangle stipple.*			
Field Name	Bits	Default	Description
X_SHIFT	3:0	0x0	Amount to shift x position before conversion to SPFP.
Y_SHIFT	19:16	0x0	Amount to shift y position before conversion to SPFP.

1.4 Graphics Backend Registers

GB:GB_AA_CONFIG · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4020			
DESCRIPTION: *Specifies the graphics pipeline configuration for antialiasing.*			
Field Name	Bits	Default	Description
AA_ENABLE	0	0x0	Enables antialiasing. POSSIBLE VALUES: 00 - Antialiasing disabled(def) 01 - Antialiasing enabled
NUM_AA_SUBSAMPLES	2:1	0x0	Specifies the number of subsamples to use while antialiasing. POSSIBLE VALUES: 00 - 2 subsamples 01 - 3 subsamples 02 - 4 subsamples 03 - 6 subsamples

GB:GB_ENABLE · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4008			
DESCRIPTION: *Specifies top of Raster pipe specific enable controls.*			
Field Name	Bits	Default	Description
POINT_STUFF_ENABLE	0	0x0	Specifies if points will have stuffed texture coordinates. POSSIBLE VALUES: 00 - Disable point texture stuffing. 01 - Enable point texture stuffing.
LINE_STUFF_ENABLE	1	0x0	Specifies if lines will have stuffed texture coordinates. POSSIBLE VALUES: 00 - Disable line texture stuffing. 01 - Enable line texture stuffing.
TRIANGLE_STUFF_ENABLE	2	0x0	Specifies if triangles will have stuffed texture coordinates. POSSIBLE VALUES: 00 - Disable triangle texture stuffing. 01 - Enable triangle texture stuffing.
STENCIL_AUTO	5:4	0x0	Specifies if the auto dec/inc stencil mode should be enabled, and how. POSSIBLE VALUES: 00 - Disable stencil auto inc/dec (def). 01 - Enable stencil auto inc/dec based on triangle cw/ccw, force into dzy low bit. 02 - Force 0 into dzy low bit.
TEX0_SOURCE	17:16	0x0	Specifies the source of the texture coordinates for this

			texture. POSSIBLE VALUES: 00 - Replicate VAP source texture coordinates 0 (S,T,[R,Q]). 01 - Stuff with source texture coordinates (S,T). 02 - Stuff with source texture coordinates (S,T,R).
TEX1_SOURCE	19:18	0x0	Specifies the source of the texture coordinates for this texture. POSSIBLE VALUES: 00 - Replicate VAP source texture coordinates 1 (S,T,[R,Q]). 01 - Stuff with source texture coordinates (S,T). 02 - Stuff with source texture coordinates (S,T,R).
TEX2_SOURCE	21:20	0x0	Specifies the source of the texture coordinates for this texture. POSSIBLE VALUES: 00 - Replicate VAP source texture coordinates 2 (S,T,[R,Q]). 01 - Stuff with source texture coordinates (S,T). 02 - Stuff with source texture coordinates (S,T,R).
TEX3_SOURCE	23:22	0x0	Specifies the source of the texture coordinates for this texture. POSSIBLE VALUES: 00 - Replicate VAP source texture coordinates 3 (S,T,[R,Q]). 01 - Stuff with source texture coordinates (S,T). 02 - Stuff with source texture coordinates (S,T,R).
TEX4_SOURCE	25:24	0x0	Specifies the source of the texture coordinates for this texture. POSSIBLE VALUES: 00 - Replicate VAP source texture coordinates 4 (S,T,[R,Q]). 01 - Stuff with source texture coordinates (S,T). 02 - Stuff with source texture coordinates (S,T,R).
TEX5_SOURCE	27:26	0x0	Specifies the source of the texture coordinates for this texture. POSSIBLE VALUES: 00 - Replicate VAP source texture coordinates 5 (S,T,[R,Q]). 01 - Stuff with source texture coordinates (S,T). 02 - Stuff with source texture coordinates (S,T,R).
TEX6_SOURCE	29:28	0x0	Specifies the source of the texture coordinates for this texture.

			POSSIBLE VALUES: 00 - Replicate VAP source texture coordinates 6 (S,T,[R,Q]). 01 - Stuff with source texture coordinates (S,T). 02 - Stuff with source texture coordinates (S,T,R).
TEX7_SOURCE	31:30	0x0	Specifies the source of the texture coordinates for this texture. POSSIBLE VALUES: 00 - Replicate VAP source texture coordinates 7 (S,T,[R,Q]). 01 - Stuff with source texture coordinates (S,T). 02 - Stuff with source texture coordinates (S,T,R).

GB:GB_FIFO_SIZE · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4024			
DESCRIPTION: *Specifies the sizes of the various FIFO`s in the sc/rs/us. This register must be the first one written*			
Field Name	Bits	Default	Description
SC_IFIFO_SIZE	1:0	0x0	Size of scan converter input FIFO (XYZ) POSSIBLE VALUES: 00 - 32 words 01 - 64 words 02 - 128 words 03 - 256 words
SC_TZFIFO_SIZE	3:2	0x0	Size of scan converter top-of-pipe Z FIFO POSSIBLE VALUES: 00 - 16 words 01 - 32 words 02 - 64 words 03 - 128 words
SC_BFIFO_SIZE	5:4	0x0	Size of scan converter input FIFO (B) POSSIBLE VALUES: 00 - 32 words 01 - 64 words 02 - 128 words 03 - 256 words
RS_TFIFO_SIZE	7:6	0x0	Size of ras input FIFO (Texture) POSSIBLE VALUES: 00 - 64 words 01 - 128 words 02 - 256 words 03 - 512 words
RS_CFIFO_SIZE	9:8	0x0	Size of ras input FIFO (Color)

			POSSIBLE VALUES: 00 - 64 words 01 - 128 words 02 - 256 words 03 - 512 words
US_RAM_SIZE	11:10	0x0	Size of us RAM POSSIBLE VALUES: 00 - 64 words 01 - 128 words 02 - 256 words 03 - 512 words
US_OFIFO_SIZE	13:12	0x0	Size of us output FIFO (RGBA) POSSIBLE VALUES: 00 - 16 words 01 - 32 words 02 - 64 words 03 - 128 words
US_WFIFO_SIZE	15:14	0x0	Size of us output FIFO (W) POSSIBLE VALUES: 00 - 16 words 01 - 32 words 02 - 64 words 03 - 128 words
RS_HIGHWATER_COL	18:16	0x0	High water mark for RS color FIFO (0-7, default 7)
RS_HIGHWATER_TEX	21:19	0x0	High water mark for RS texture FIFO (0-7, default 7)
US_OFIFO_HIGHWATER	23:22	0x0	High water mark for US output FIFO (0-12, default 4) POSSIBLE VALUES: 00 - 0 words 01 - 4 words 02 - 8 words 03 - 12 words
US_CUBE_FIFO_HIGHWATER	27:24	0x0	High water mark for US texture output FIFO (0-15, default 11)

GB:GB_MSPOS0 · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4010			
DESCRIPTION: *Specifies the position of multisamples 0 through 2*			
Field Name	Bits	Default	Description
MS_X0	3:0	0x0	Specifies the x and y position (in subpixels) of multisample 0
MS_Y0	7:4	0x0	Specifies the x and y position (in subpixels) of multisample 0

MS_X1	11:8	0x0	Specifies the x and y position (in subpixels) of multisample 1
MS_Y1	15:12	0x0	Specifies the x and y position (in subpixels) of multisample 1
MS_X2	19:16	0x0	Specifies the x and y position (in subpixels) of multisample 2
MS_Y2	23:20	0x0	Specifies the x and y position (in subpixels) of multisample 2
MSBD0_Y	27:24	0x0	Specifies the minimum y distance (in subpixels) between the pixel edge and the multisample bounding box. This value is used in the tile scan converter
MSBD0_X	31:28	0x0	msbd0_x[2:0] specifies the minimum x distance (in subpixels) between the pixel edge and the multisample bounding box. This value is used in the tile scan converter. The special case value of 8 is represented by msbd0_x[2:0]=7. msbd0_x[3] is used to force a bounding box based tile scan conversion instead of an intercept based one. This value should always be set to 0.

GB:GB_MSPOS1 · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4014			
DESCRIPTION: *Specifies the position of multisamples 3 through 5*			
Field Name	Bits	Default	Description
MS_X3	3:0	0x0	Specifies the x and y position (in subpixels) of multisample 3
MS_Y3	7:4	0x0	Specifies the x and y position (in subpixels) of multisample 3
MS_X4	11:8	0x0	Specifies the x and y position (in subpixels) of multisample 4
MS_Y4	15:12	0x0	Specifies the x and y position (in subpixels) of multisample 4
MS_X5	19:16	0x0	Specifies the x and y position (in subpixels) of multisample 5
MS_Y5	23:20	0x0	Specifies the x and y position (in subpixels) of multisample 5
MSBD1	27:24	0x0	Specifies the minimum distance (in subpixels) between the pixel edge and the multisample bounding box. This value is used in the quad scan converter

GB:GB_SELECT · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x401c			
DESCRIPTION: *Specifies various polygon specific selects (fog, depth, perspective).*			
Field Name	Bits	Default	Description
FOG_SELECT	2:0	0x0	Specifies source for outgoing (GA to SU) fog value. POSSIBLE VALUES:

			00 - Select C0A 01 - Select C1A 02 - Select C2A 03 - Select C3A 04 - Select 1/(1/W) 05 - Select Z
DEPTH_SELECT	3	0x0	Specifies source for outgoing (GA/SU & SU/RAS) depth value. POSSIBLE VALUES: 00 - Select Z 01 - Select 1/(1/W)
W_SELECT	4	0x0	Specifies source for outgoing (1/W) value, used to disable perspective correct colors/textures. POSSIBLE VALUES: 00 - Select (1/W) 01 - Select 1.0

GB:GB_TILE_CONFIG · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4018			
DESCRIPTION: *Specifies the graphics pipeline configuration for rasterization*			
Field Name	Bits	Default	Description
ENABLE	0	0x1	Enables tiling, otherwise all tiles receive all polygons. POSSIBLE VALUES: 00 - Tiling disabled. 01 - Tiling enabled (def).
PIPE_COUNT	3:1	0x0	Specifies the number of active pipes and contexts. POSSIBLE VALUES: 00 - RV350 03 - R300
TILE_SIZE	5:4	0x1	Specifies width & height (square), in pixels. POSSIBLE VALUES: 00 - 8 pixels (not supported by zb/cb) 01 - 16 pixels 02 - 32 pixels (not supported by zb/cb)
SUPER_SIZE	8:6	0x0	Specifies number of tiles and config in super chip configuration. POSSIBLE VALUES: 00 - 1x1 tile (one 1x1). 01 - 2 tiles (two 1x1 : ST-A,B). 02 - 4 tiles (one 2x2). 03 - 8 tiles (two 2x2 : ST-A,B). 04 - 16 tiles (one 4x4). 05 - 32 tiles (two 4x4 : ST-A,B).

			06 - 64 tiles (one 8x8). 07 - 128 tiles (two 8x8 : ST-A,B).
SUPER_X	11:9	0x0	X Location of chip within super tile.
SUPER_Y	14:12	0x0	Y Location of chip within super tile.
SUPER_TILE	15	0x0	Tile location of chip in a multi super tile config (Super size of 2,8,32 or 128). POSSIBLE VALUES: 00 - ST-A tile. 01 - ST-B tile.
SUBPIXEL	16	0x0	Specifies the subpixel precision. POSSIBLE VALUES: 00 - Select 1/12 subpixel precision. 01 - Select 1/16 subpixel precision.
QUADS_PER_RAS	18:17	0x0	unused
BB_SCAN	19	0x0	unused

1.5 Rasterizer Registers

RS:RS_COUNT · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4300			
DESCRIPTION: *This register specifies the rasterizer input packet configuration*			
Field Name	Bits	Default	Description
IT_COUNT	6:0	0x0	Specifies the total number of texture address components contained in the rasterizer input packet (0:32).
IC_COUNT	10:7	0x0	Specifies the total number of colors contained in the rasterizer input packet (0:4).
W_COUNT	11	0x0	Specifies the total number of w values contained in the rasterizer input packet (0 or 1).
W_ADDR	17:12	0x0	Specifies the relative rasterizer input packet location of w (if w_count==1)
HIRES_EN	18	0x0	Enable high resolution texture coordinate output when q is equal to 1

RS:RS_INST_[0-15] · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4330-0x436c			
DESCRIPTION: *This table specifies what happens during each rasterizer instruction*			
Field Name	Bits	Default	Description
TEX_ID	2:0	0x0	Specifies the index (into the RS_IP table) of the texture address output during this rasterizer instruction
TEX_CN	5:3	0x0	Write enable for texture address POSSIBLE VALUES: 00 - No write - texture coordinate not valid 01 - write - texture valid
TEX_ADDR	10:6	0x0	Specifies the destination address (within the current pixel stack frame) of the texture address output during this rasterizer instruction
COL_ID	13:11	0x0	Specifies the index (into the RS_IP table) of the color output during this rasterizer instruction
COL_CN	16:14	0x0	Write enable for color POSSIBLE VALUES: 00 - No write - color not valid 01 - write - color valid
COL_ADDR	21:17	0x0	Specifies the destination address (within the current pixel stack frame) of the color output during this rasterizer instruction
TEX_ADJ	22	0x0	Specifies whether to sample texture coordinates at the real or adjusted pixel centers POSSIBLE VALUES: 00 - Sample texture coordinates at real pixel centers 01 - Sample texture coordinates at adjusted pixel centers

COL_BIAS	24:23	0x0	unused

RS:RS_INST_COUNT · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4304			
DESCRIPTION: *This register specifies the number of rasterizer instructions*			
Field Name	Bits	Default	Description
INST_COUNT	3:0	0x0	Number of rasterizer instructions (1:16)
W_EN	4	0x0	Specifies that the rasterizer needs to generate w
TX_OFFSET	7:5	0x0	Defines texture coordinate offset (based on min/max coordinate range of triangle) used to minimize or eliminate peroidic errors on texels sampled right on their edges POSSIBLE VALUES: 00 - 0.0 01 - range/8K 02 - range/16K 03 - range/32K 04 - range/64K 05 - range/128K 06 - range/256K 07 - range/512K

RS:RS_IP_[0-7] · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4310 -0x432c			
DESCRIPTION: *This table specifies the source location and format for up to 8 texture addresses (i[0]:i[7]) and four colors (c[0]:c[3])*			
Field Name	Bits	Default	Description
TEX_PTR	5:0	0x0	Specifies the relative rasterizer input packet location of texture address (i[i]).
COL_PTR	8:6	0x0	Specifies the relative rasterizer input packet location of the color (c[i]).
COL_FMT	12:9	0x0	Specifies the format of the color (c[i]). POSSIBLE VALUES: 00 - Four components (R,G,B,A) 01 - Three components (R,G,B,0) 02 - Three components (R,G,B,1) 04 - One component (0,0,0,A) 05 - Zero components (0,0,0,0) 06 - Zero components (0,0,0,1) 08 - One component (1,1,1,A) 09 - Zero components (1,1,1,0) 10 - Zero components (1,1,1,1)
SEL_S	15:13	0x0	Source select for S, T, R, and Q POSSIBLE VALUES: 00 - C0 - 1st texture component

			01 - C1 - 2nd texture component 02 - C2 - 3rd texture component 03 - C3 - 4th texture component 04 - K0 - The value 0.0 05 - K1 - The value 1.0
SEL_T	18:16	0x0	Source select for S, T, R, and Q POSSIBLE VALUES: 00 - C0 - 1st texture component 01 - C1 - 2nd texture component 02 - C2 - 3rd texture component 03 - C3 - 4th texture component 04 - K0 - The value 0.0 05 - K1 - The value 1.0
SEL_R	21:19	0x0	Source select for S, T, R, and Q POSSIBLE VALUES: 00 - C0 - 1st texture component 01 - C1 - 2nd texture component 02 - C2 - 3rd texture component 03 - C3 - 4th texture component 04 - K0 - The value 0.0 05 - K1 - The value 1.0
SEL_Q	24:22	0x0	Source select for S, T, R, and Q POSSIBLE VALUES: 00 - C0 - 1st texture component 01 - C1 - 2nd texture component 02 - C2 - 3rd texture component 03 - C3 - 4th texture component 04 - K0 - The value 0.0 05 - K1 - The value 1.0

1.6 Clipping Registers

SC:SC_CLIP_0_A · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x43b0			
DESCRIPTION: *OpenGL Clip rectangles*			
Field Name	Bits	Default	Description
XS0	12:0	0x0	Left hand edge of clip rectangle
YS0	25:13	0x0	Upper edge of clip rectangle

SC:SC_CLIP_0_B · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x43b4			
DESCRIPTION: *OpenGL Clip rectangles*			
Field Name	Bits	Default	Description
XS1	12:0	0x0	Right hand edge of clip rectangle
YS1	25:13	0x0	Lower edge of clip rectangle

SC:SC_CLIP_1_A · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x43b8			
Field Name	Bits	Default	Description
XS0	12:0	0x0	
YS0	25:13	0x0	

SC:SC_CLIP_1_B · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x43bc			
Field Name	Bits	Default	Description
XS1	12:0	0x0	
YS1	25:13	0x0	

SC:SC_CLIP_2_A · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x43c0			
Field Name	Bits	Default	Description
XS0	12:0	0x0	
YS0	25:13	0x0	

SC:SC_CLIP_2_B · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x43c4			
Field Name	Bits	Default	Description
XS1	12:0	0x0	
YS1	25:13	0x0	

SC:SC_CLIP_3_A · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x43c8			
Field Name	Bits	Default	Description

XS0	12:0	0x0	
YS0	25:13	0x0	

SC:SC_CLIP_3_B · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x43cc			
Field Name	Bits	Default	Description
XS1	12:0	0x0	
YS1	25:13	0x0	

SC:SC_CLIP_RULE · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x43d0			
DESCRIPTION: *OpenGL Clip boolean function*			
Field Name	Bits	Default	Description
CLIP_RULE	15:0	0x0	OpenGL Clip boolean function. The `inside` flags for each of the four clip rectangles form a 4-bit binary number. The corresponding bit in this 16-bit number specifies whether the pixel is visible.

SC:SC_EDGERULE · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x43a8			
DESCRIPTION: *Edge rules - what happens when an edge falls exactly on a sample point*			
Field Name	Bits	Default	Description
ER_TRI	4:0	0x0	Edge rules for triangles, points, left-right lines, right-left lines, upper-bottom lines, bottom-upper lines. For values 0 to 15, bit 0 specifies whether a sample on a horizontal-bottom edge is in, bit 1 specifies whether a sample on a horizontal-top edge is in, bit 2 species whether a sample on a right edge is in, bit 3 specifies whether a sample on a left edge is in. For values 16 to 31, bit 0 specifies whether a sample on a vertical-right edge is in, bit 1 specifies whether a sample on a vertical-left edge is in, bit 2 species whether a sample on a bottom edge is in, bit 3 specifies whether a sample on a top edge is in POSSIBLE VALUES: 00 - L-in,R-in,HT-in,HB-in 01 - L-in,R-in,HT-in,HB-out 02 - L-in,R-in,HT-out,HB-in 03 - L-in,R-in,HT-out,HB-out 04 - L-in,R-out,HT-in,HB-in 05 - L-in,R-out,HT-in,HB-out 06 - L-in,R-out,HT-out,HB-in 07 - L-in,R-out,HT-out,HB-out 08 - L-out,R-in,HT-in,HB-in 09 - L-out,R-in,HT-in,HB-out 10 - L-out,R-in,HT-out,HB-in 11 - L-out,R-in,HT-out,HB-out 12 - L-out,R-out,HT-in,HB-in

			13 - L-out,R-out,HT-in,HB-out 14 - L-out,R-out,HT-out,HB-in 15 - L-out,R-out,HT-out,HB-out 16 - T-in,B-in,VL-in,VR-in 17 - T-in,B-in,VL-in,VR-out 18 - T-in,B-in,VL,VR-in 19 - T-in,B-in,VL-out,VR-out 20 - T-out,B-in,VL-in,VR-in 21 - T-out,B-in,VL-in,VR-out 22 - T-out,B-in,VL-out,VR-in 23 - T-out,B-in,VL-out,VR-out 24 - T-in,B-out,VL-in,VR-in 25 - T-in,B-out,VL-in,VR-out 26 - T-in,B-out,VL-out,VR-in 27 - T-in,B-out,VL-out,VR-out 28 - T-out,B-out,VL-in,VR-in 29 - T-out,B-out,VL-in,VR-out 30 - T-out,B-out,VL-out,VR-in 31 - T-out,B-out,VL-out,VR-out
ER_POINT	9:5	0x0	Edge rules for triangles, points, left-right lines, right-left lines, upper-bottom lines, bottom-upper lines. For values 0 to 15, bit 0 specifies whether a sample on a horizontal-bottom edge is in, bit 1 specifies whether a sample on a horizontal-top edge is in, bit 2 species whether a sample on a right edge is in, bit 3 specifies whether a sample on a left edge is in. For values 16 to 31, bit 0 specifies whether a sample on a vertical-right edge is in, bit 1 specifies whether a sample on a vertical-left edge is in, bit 2 species whether a sample on a bottom edge is in, bit 3 specifies whether a sample on a top edge is in POSSIBLE VALUES: 00 - L-in,R-in,HT-in,HB-in 01 - L-in,R-in,HT-in,HB-out 02 - L-in,R-in,HT-out,HB-in 03 - L-in,R-in,HT-out,HB-out 04 - L-in,R-out,HT-in,HB-in 05 - L-in,R-out,HT-in,HB-out 06 - L-in,R-out,HT-out,HB-in 07 - L-in,R-out,HT-out,HB-out 08 - L-out,R-in,HT-in,HB-in 09 - L-out,R-in,HT-in,HB-out 10 - L-out,R-in,HT-out,HB-in 11 - L-out,R-in,HT-out,HB-out 12 - L-out,R-out,HT-in,HB-in 13 - L-out,R-out,HT-in,HB-out 14 - L-out,R-out,HT-out,HB-in 15 - L-out,R-out,HT-out,HB-out 16 - T-in,B-in,VL-in,VR-in 17 - T-in,B-in,VL-in,VR-out 18 - T-in,B-in,VL,VR-in 19 - T-in,B-in,VL-out,VR-out 20 - T-in,B-out,VL-in,VR-in

			21 - T-in,B-out,VL-in,VR-out 22 - T-in,B-out,VL-out,VR-in 23 - T-in,B-out,VL-out,VR-out 24 - T-out,B-in,VL-in,VR-in 25 - T-out,B-in,VL-in,VR-out 26 - T-out,B-in,VL-out,VR-in 27 - T-out,B-in,VL-out,VR-out 28 - T-out,B-out,VL-in,VR-in 29 - T-out,B-out,VL-in,VR-out 30 - T-out,B-out,VL-out,VR-in 31 - T-out,B-out,VL-out,VR-out
ER_LINE_LR	14:10	0x0	Edge rules for triangles, points, left-right lines, right-left lines, upper-bottom lines, bottom-upper lines. For values 0 to 15, bit 0 specifies whether a sample on a horizontal-bottom edge is in, bit 1 specifies whether a sample on a horizontal-top edge is in, bit 2 species whether a sample on a right edge is in, bit 3 specifies whether a sample on a left edge is in. For values 16 to 31, bit 0 specifies whether a sample on a vertical-right edge is in, bit 1 specifies whether a sample on a vertical-left edge is in, bit 2 species whether a sample on a bottom edge is in, bit 3 specifies whether a sample on a top edge is in POSSIBLE VALUES: 00 - L-in,R-in,HT-in,HB-in 01 - L-in,R-in,HT-in,HB-out 02 - L-in,R-in,HT-out,HB-in 03 - L-in,R-in,HT-out,HB-out 04 - L-in,R-out,HT-in,HB-in 05 - L-in,R-out,HT-in,HB-out 06 - L-in,R-out,HT-out,HB-in 07 - L-in,R-out,HT-out,HB-out 08 - L-out,R-in,HT-in,HB-in 09 - L-out,R-in,HT-in,HB-out 10 - L-out,R-in,HT-out,HB-in 11 - L-out,R-in,HT-out,HB-out 12 - L-out,R-out,HT-in,HB-in 13 - L-out,R-out,HT-in,HB-out 14 - L-out,R-out,HT-out,HB-in 15 - L-out,R-out,HT-out,HB-out 16 - T-in,B-in,VL-in,VR-in 17 - T-in,B-in,VL-in,VR-out 18 - T-in,B-in,VL,VR-in 19 - T-in,B-in,VL-out,VR-out 20 - T-in,B-out,VL-in,VR-in 21 - T-in,B-out,VL-in,VR-out 22 - T-in,B-out,VL-out,VR-in 23 - T-in,B-out,VL-out,VR-out 24 - T-out,B-in,VL-in,VR-in 25 - T-out,B-in,VL-in,VR-out 26 - T-out,B-in,VL-out,VR-in 27 - T-out,B-in,VL-out,VR-out 28 - T-out,B-out,VL-in,VR-in

			29 - T-out,B-out,VL-in,VR-out 30 - T-out,B-out,VL-out,VR-in 31 - T-out,B-out,VL-out,VR-out
ER_LINE_RL	19:15	0x0	Edge rules for triangles, points, left-right lines, right-left lines, upper-bottom lines, bottom-upper lines. For values 0 to 15, bit 0 specifies whether a sample on a horizontal-bottom edge is in, bit 1 specifies whether a sample on a horizontal-top edge is in, bit 2 species whether a sample on a right edge is in, bit 3 specifies whether a sample on a left edge is in. For values 16 to 31, bit 0 specifies whether a sample on a vertical-right edge is in, bit 1 specifies whether a sample on a vertical-left edge is in, bit 2 species whether a sample on a bottom edge is in, bit 3 specifies whether a sample on a top edge is in POSSIBLE VALUES: 00 - L-in,R-in,HT-in,HB-in 01 - L-in,R-in,HT-in,HB-out 02 - L-in,R-in,HT-out,HB-in 03 - L-in,R-in,HT-out,HB-out 04 - L-in,R-out,HT-in,HB-in 05 - L-in,R-out,HT-in,HB-out 06 - L-in,R-out,HT-out,HB-in 07 - L-in,R-out,HT-out,HB-out 08 - L-out,R-in,HT-in,HB-in 09 - L-out,R-in,HT-in,HB-out 10 - L-out,R-in,HT-out,HB-in 11 - L-out,R-in,HT-out,HB-out 12 - L-out,R-out,HT-in,HB-in 13 - L-out,R-out,HT-in,HB-out 14 - L-out,R-out,HT-out,HB-in 15 - L-out,R-out,HT-out,HB-out 16 - T-in,B-in,VL-in,VR-in 17 - T-in,B-in,VL-in,VR-out 18 - T-in,B-in,VL,VR-in 19 - T-in,B-in,VL-out,VR-out 20 - T-in,B-out,VL-in,VR-in 21 - T-in,B-out,VL-in,VR-out 22 - T-in,B-out,VL-out,VR-in 23 - T-in,B-out,VL-out,VR-out 24 - T-out,B-in,VL-in,VR-in 25 - T-out,B-in,VL-in,VR-out 26 - T-out,B-in,VL-out,VR-in 27 - T-out,B-in,VL-out,VR-out 28 - T-out,B-out,VL-in,VR-in 29 - T-out,B-out,VL-in,VR-out 30 - T-out,B-out,VL-out,VR-in 31 - T-out,B-out,VL-out,VR-out
ER_LINE_TB	24:20	0x0	Edge rules for triangles, points, left-right lines, right-left lines, upper-bottom lines, bottom-upper lines. For values 0 to 15, bit 0 specifies whether a sample on a horizontal-bottom edge is in, bit 1 specifies whether a sample on a

			horizontal-top edge is in, bit 2 species whether a sample on a right edge is in, bit 3 specifies whether a sample on a left edge is in. For values 16 to 31, bit 0 specifies whether a sample on a vertical-right edge is in, bit 1 specifies whether a sample on a vertical-left edge is in, bit 2 species whether a sample on a bottom edge is in, bit 3 specifies whether a sample on a top edge is in POSSIBLE VALUES: 00 - L-in,R-in,HT-in,HB-in 01 - L-in,R-in,HT-in,HB-out 02 - L-in,R-in,HT-out,HB-in 03 - L-in,R-in,HT-out,HB-out 04 - L-in,R-out,HT-in,HB-in 05 - L-in,R-out,HT-in,HB-out 06 - L-in,R-out,HT-out,HB-in 07 - L-in,R-out,HT-out,HB-out 08 - L-out,R-in,HT-in,HB-in 09 - L-out,R-in,HT-in,HB-out 10 - L-out,R-in,HT-out,HB-in 11 - L-out,R-in,HT-out,HB-out 12 - L-out,R-out,HT-in,HB-in 13 - L-out,R-out,HT-in,HB-out 14 - L-out,R-out,HT-out,HB-in 15 - L-out,R-out,HT-out,HB-out 16 - T-in,B-in,VL-in,VR-in 17 - T-in,B-in,VL-in,VR-out 18 - T-in,B-in,VL,VR-in 19 - T-in,B-in,VL-out,VR-out 20 - T-in,B-out,VL-in,VR-in 21 - T-in,B-out,VL-in,VR-out 22 - T-in,B-out,VL-out,VR-in 23 - T-in,B-out,VL-out,VR-out 24 - T-out,B-in,VL-in,VR-in 25 - T-out,B-in,VL-in,VR-out 26 - T-out,B-in,VL-out,VR-in 27 - T-out,B-in,VL-out,VR-out 28 - T-out,B-out,VL-in,VR-in 29 - T-out,B-out,VL-in,VR-out 30 - T-out,B-out,VL-out,VR-in 31 - T-out,B-out,VL-out,VR-out
ER_LINE_BT	29:25	0x0	Edge rules for triangles, points, left-right lines, right-left lines, upper-bottom lines, bottom-upper lines. For values 0 to 15, bit 0 specifies whether a sample on a horizontal-bottom edge is in, bit 1 specifies whether a sample on a horizontal-top edge is in, bit 2 species whether a sample on a right edge is in, bit 3 specifies whether a sample on a left edge is in. For values 16 to 31, bit 0 specifies whether a sample on a vertical-right edge is in, bit 1 specifies whether a sample on a vertical-left edge is in, bit 2 species whether a sample on a bottom edge is in, bit 3 specifies whether a sample on a top edge is in

			POSSIBLE VALUES: 00 - L-in,R-in,HT-in,HB-in 01 - L-in,R-in,HT-in,HB-out 02 - L-in,R-in,HT-out,HB-in 03 - L-in,R-in,HT-out,HB-out 04 - L-in,R-out,HT-in,HB-in 05 - L-in,R-out,HT-in,HB-out 06 - L-in,R-out,HT-out,HB-in 07 - L-in,R-out,HT-out,HB-out 08 - L-out,R-in,HT-in,HB-in 09 - L-out,R-in,HT-in,HB-out 10 - L-out,R-in,HT-out,HB-in 11 - L-out,R-in,HT-out,HB-out 12 - L-out,R-out,HT-in,HB-in 13 - L-out,R-out,HT-in,HB-out 14 - L-out,R-out,HT-out,HB-in 15 - L-out,R-out,HT-out,HB-out 16 - T-in,B-in,VL-in,VR-in 17 - T-in,B-in,VL-in,VR-out 18 - T-in,B-in,VL,VR-in 19 - T-in,B-in,VL-out,VR-out 20 - T-in,B-out,VL-in,VR-in 21 - T-in,B-out,VL-in,VR-out 22 - T-in,B-out,VL-out,VR-in 23 - T-in,B-out,VL-out,VR-out 24 - T-out,B-in,VL-in,VR-in 25 - T-out,B-in,VL-in,VR-out 26 - T-out,B-in,VL-out,VR-in 27 - T-out,B-in,VL-out,VR-out 28 - T-out,B-out,VL-in,VR-in 29 - T-out,B-out,VL-in,VR-out 30 - T-out,B-out,VL-out,VR-in 31 - T-out,B-out,VL-out,VR-out

SC:SC_HYPERZ_EN · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x43a4			
DESCRIPTION: *Hierarchical Z Enable*			
Field Name	Bits	Default	Description
HZ_EN	0	0x0	Enable for hierarchical Z. POSSIBLE VALUES: 00 - Disables Hyper-Z. 01 - Enables Hyper-Z.
HZ_MAX	1	0x0	Specifies whether to compute min or max z value POSSIBLE VALUES: 00 - HZ block computes minimum z value 01 - HZ block computes maximum z value
HZ_ADJ	4:2	0x0	Specifies adjustment to get added or subtracted from

			computed z value POSSIBLE VALUES: 00 - Add or Subtract 1/256 << ze 01 - Add or Subtract 1/128 << ze 02 - Add or Subtract 1/64 << ze 03 - Add or Subtract 1/32 << ze 04 - Add or Subtract 1/16 << ze 05 - Add or Subtract 1/8 << ze 06 - Add or Subtract 1/4 << ze 07 - Add or Subtract 1/2 << ze
HZ_Z0MIN	5	0x0	Specifies whether vertex 0 z contains minimum z value POSSIBLE VALUES: 00 - Vertex 0 does not contain minimum z value 01 - Vertex 0 does contain minimum z value
HZ_Z0MAX	6	0x0	Specifies whether vertex 0 z contains maximum z value POSSIBLE VALUES: 00 - Vertex 0 does not contain maximum z value 01 - Vertex 0 does contain maximum z value

SC:SC_SCISSOR0 · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x43e0			
DESCRIPTION: *Scissor rectangle specification*			
Field Name	Bits	Default	Description
XS0	12:0	0x0	Left hand edge of scissor rectangle
YS0	25:13	0x0	Upper edge of scissor rectangle

SC:SC_SCISSOR1 · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x43e4			
DESCRIPTION: *Scissor rectangle specification*			
Field Name	Bits	Default	Description
XS1	12:0	0x0	Right hand edge of scissor rectangle
YS1	25:13	0x0	Lower edge of scissor rectangle

SC:SC_SCREENDOOR · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x43e8			
DESCRIPTION: *Screen door sample mask*			
Field Name	Bits	Default	Description
SCREENDOOR	23:0	0x0	Screen door sample mask - 1 means sample may be covered, 0 means sample is not covered

1.7 Setup Unit Registers

SU:SU_CULL_MODE · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x42b8			
DESCRIPTION: *Culling Enables*			
Field Name	Bits	Default	Description
CULL_FRONT	0	0x0	Enable for front-face culling. POSSIBLE VALUES: 00 - Do not cull front-facing triangles. 01 - Cull front-facing triangles.
CULL_BACK	1	0x0	Enable for back-face culling. POSSIBLE VALUES: 00 - Do not cull back-facing triangles. 01 - Cull back-facing triangles.
FACE	2	0x0	X-Ored with cross product sign to determine positive facing POSSIBLE VALUES: 00 - Positive cross product is front (CCW). 01 - Negative cross product is front (CW).

SU:SU_DEPTH_OFFSET · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x42c4			
DESCRIPTION: *SU Depth Offset value*			
Field Name	Bits	Default	Description
OFFSET	31:0	0x0	SPFP Floating point applied to depth before conversion to FXP.

SU:SU_DEPTH_SCALE · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x42c0			
DESCRIPTION: *SU Depth Scale value*			
Field Name	Bits	Default	Description
SCALE	31:0	0x3F800000	SPFP Floating point applied to depth before conversion to FXP.

SU:SU_POLY_OFFSET_BACK_OFFSET · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x42b0			
DESCRIPTION: *Back-Facing Polygon Offset Offset*			
Field Name	Bits	Default	Description
OFFSET	31:0	0x0	Specifies polygon offset offset for back-facing polygons; 32b IEEE float format; applied after Z scale & offset (0 to 2^24-1 range)

SU:SU_POLY_OFFSET_BACK_SCALE · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x42ac			
DESCRIPTION: *Back-Facing Polygon Offset Scale*			
Field Name	Bits	Default	Description
SCALE	31:0	0x0	Specifies polygon offset scale for back-facing polygons; 32-bit IEEE float format; applied after Z scale & offset (0 to 2^24-1 range); slope computed in subpixels (1/12 or 1/16)

SU:SU_POLY_OFFSET_ENABLE · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x42b4			
DESCRIPTION: *Enables for polygon offset*			
Field Name	Bits	Default	Description
FRONT_ENABLE	0	0x0	Enables front facing polygon`s offset. POSSIBLE VALUES: 00 - Disable front offset. 01 - Enable front offset.
BACK_ENABLE	1	0x0	Enables back facing polygon`s offset. POSSIBLE VALUES: 00 - Disable back offset. 01 - Enable back offset.
PARA_ENABLE	2	0x0	Forces all parallelograms to have FRONT_FACING for poly offset -- Need to have FRONT_ENABLE also set to have Z offset for parallelograms. POSSIBLE VALUES: 00 - Disable front offset for parallelograms. 01 - Enable front offset for parallelograms.

SU:SU_POLY_OFFSET_FRONT_OFFSET · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x42a8			
DESCRIPTION: *Front-Facing Polygon Offset Offset*			
Field Name	Bits	Default	Description
OFFSET	31:0	0x0	Specifies polygon offset offset for front-facing polygons; 32b IEEE float format; applied after Z scale & offset (0 to 2^24-1 range)

SU:SU_POLY_OFFSET_FRONT_SCALE · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x42a4			
DESCRIPTION: *Front-Facing Polygon Offset Scale*			
Field Name	Bits	Default	Description
SCALE	31:0	0x0	Specifies polygon offset scale for front-facing polygons; 32b IEEE float format; applied after Z scale & offset (0 to 2^24-1 range); slope computed in subpixels (1/12 or 1/16)

SU:SU_REG_DEST · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x42c8			
DESCRIPTION: *SU Raster pipe destination select for registers*			
Field Name	Bits	Default	Description
SELECT	3:0	0xF	Select which of the 2 pipes (enable per pipe) to send register read/write to. b0: P0 enable, b3: P1 enable

SU:SU_TEX_WRAP · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x42a0			
DESCRIPTION: *Enables for Cylindrical Wrapping*			
Field Name	Bits	Default	Description
T0C0	0	0x0	POSSIBLE VALUES: 00 - Disable cylindrical wrapping for tex 0 comp 0. 01 - Enable cylindrical wrapping for tex 0 comp 0.
T0C1	1	0x0	POSSIBLE VALUES: 00 - Disable cylindrical wrapping for tex 0 comp 1. 01 - Enable cylindrical wrapping for tex 0 comp 1.
T0C2	2	0x0	POSSIBLE VALUES: 00 - Disable cylindrical wrapping for tex 0 comp 2. 01 - Enable cylindrical wrapping for tex 0 comp 2.
T0C3	3	0x0	POSSIBLE VALUES: 00 - Disable cylindrical wrapping for tex 0 comp 3. 01 - Enable cylindrical wrapping for tex 0 comp 3.
T1C0	4	0x0	POSSIBLE VALUES: 00 - Disable cylindrical wrapping for tex 1 comp 0. 01 - Enable cylindrical wrapping for tex 1 comp 0.
T1C1	5	0x0	POSSIBLE VALUES: 00 - Disable cylindrical wrapping for tex 1 comp 1. 01 - Enable cylindrical wrapping for tex 1 comp 1.
T1C2	6	0x0	POSSIBLE VALUES: 00 - Disable cylindrical wrapping for tex 1 comp 2. 01 - Enable cylindrical wrapping for tex 1 comp 2.
T1C3	7	0x0	POSSIBLE VALUES: 00 - Disable cylindrical wrapping for tex 1 comp 3. 01 - Enable cylindrical wrapping for tex 1 comp 3.
T2C0	8	0x0	POSSIBLE VALUES: 00 - Disable cylindrical wrapping for tex 2 comp 0. 01 - Enable cylindrical wrapping for tex 2 comp 0.
T2C1	9	0x0	POSSIBLE VALUES: 00 - Disable cylindrical wrapping for tex 2 comp 1. 01 - Enable cylindrical wrapping for tex 2 comp 1.
T2C2	10	0x0	POSSIBLE VALUES: 00 - Disable cylindrical wrapping for tex 2 comp 2. 01 - Enable cylindrical wrapping for tex 2 comp 2.
T2C3	11	0x0	POSSIBLE VALUES: 00 - Disable cylindrical wrapping for tex 2 comp 3.

			01 - Enable cylindrical wrapping for tex 2 comp 3.
T3C0	12	0x0	POSSIBLE VALUES: 00 - Disable cylindrical wrapping for tex 3 comp 0. 01 - Enable cylindrical wrapping for tex 3 comp 0.
T3C1	13	0x0	POSSIBLE VALUES: 00 - Disable cylindrical wrapping for tex 3 comp 1. 01 - Enable cylindrical wrapping for tex 3 comp 1.
T3C2	14	0x0	POSSIBLE VALUES: 00 - Disable cylindrical wrapping for tex 3 comp 2. 01 - Enable cylindrical wrapping for tex 3 comp 2.
T3C3	15	0x0	POSSIBLE VALUES: 00 - Disable cylindrical wrapping for tex 3 comp 3. 01 - Enable cylindrical wrapping for tex 3 comp 3.
T4C0	16	0x0	POSSIBLE VALUES: 00 - Disable cylindrical wrapping for tex 4 comp 0. 01 - Enable cylindrical wrapping for tex 4 comp 0.
T4C1	17	0x0	POSSIBLE VALUES: 00 - Disable cylindrical wrapping for tex 4 comp 1. 01 - Enable cylindrical wrapping for tex 4 comp 1.
T4C2	18	0x0	POSSIBLE VALUES: 00 - Disable cylindrical wrapping for tex 4 comp 2. 01 - Enable cylindrical wrapping for tex 4 comp 2.
T4C3	19	0x0	POSSIBLE VALUES: 00 - Disable cylindrical wrapping for tex 4 comp 3. 01 - Enable cylindrical wrapping for tex 4 comp 3.
T5C0	20	0x0	POSSIBLE VALUES: 00 - Disable cylindrical wrapping for tex 5 comp 0. 01 - Enable cylindrical wrapping for tex 5 comp 0.
T5C1	21	0x0	POSSIBLE VALUES: 00 - Disable cylindrical wrapping for tex 5 comp 1. 01 - Enable cylindrical wrapping for tex 5 comp 1.
T5C2	22	0x0	POSSIBLE VALUES: 00 - Disable cylindrical wrapping for tex 5 comp 2. 01 - Enable cylindrical wrapping for tex 5 comp 2.
T5C3	23	0x0	POSSIBLE VALUES: 00 - Disable cylindrical wrapping for tex 5 comp 3. 01 - Enable cylindrical wrapping for tex 5 comp 3.
T6C0	24	0x0	POSSIBLE VALUES: 00 - Disable cylindrical wrapping for tex 6 comp 0. 01 - Enable cylindrical wrapping for tex 6 comp 0.
T6C1	25	0x0	POSSIBLE VALUES: 00 - Disable cylindrical wrapping for tex 6 comp 1. 01 - Enable cylindrical wrapping for tex 6 comp 1.
T6C2	26	0x0	POSSIBLE VALUES: 00 - Disable cylindrical wrapping for tex 6 comp 2. 01 - Enable cylindrical wrapping for tex 6 comp 2.
T6C3	27	0x0	POSSIBLE VALUES: 00 - Disable cylindrical wrapping for tex 6 comp 3.

			01 - Enable cylindrical wrapping for tex 6 comp 3.
T7C0	28	0x0	POSSIBLE VALUES: 00 - Disable cylindrical wrapping for tex 7 comp 0. 01 - Enable cylindrical wrapping for tex 7 comp 0.
T7C1	29	0x0	POSSIBLE VALUES: 00 - Disable cylindrical wrapping for tex 7 comp 1. 01 - Enable cylindrical wrapping for tex 7 comp 1.
T7C2	30	0x0	POSSIBLE VALUES: 00 - Disable cylindrical wrapping for tex 7 comp 2. 01 - Enable cylindrical wrapping for tex 7 comp 2.
T7C3	31	0x0	POSSIBLE VALUES: 00 - Disable cylindrical wrapping for tex 7 comp 3. 01 - Enable cylindrical wrapping for tex 7 comp 3.

1.8 Texture Registers

TX:TX_BORDER_COLOR_[0-15] · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x45c0-0x45fc			
DESCRIPTION: *Border Color for Map 0*			
Field Name	Bits	Default	Description
BORDER_COLOR	31:0	none	Color used for borders. Format is the same as the texture being bordered.

TX:TX_CHROMA_KEY_[0-15] · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4580-0x45bc			
DESCRIPTION: *Texture Chroma Key for Map 0*			
Field Name	Bits	Default	Description
CHROMA_KEY	31:0	none	Color used for chroma key compare. Format is the same as the texture being keyed.

TX:TX_ENABLE · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4104			
DESCRIPTION: *Texture Enables for Maps 0 to 15*			
Field Name	Bits	Default	Description
TEX_0_ENABLE	0	none	Texture Map 0 Enable. POSSIBLE VALUES: 00 - Disable, T0(ARGB) = 1,0,0,0 01 - Enable
TEX_1_ENABLE	1	none	Texture Map 1 Enable. POSSIBLE VALUES: 00 - Disable, T1(ARGB) = 1,0,0,0 01 - Enable
TEX_2_ENABLE	2	none	Texture Map 2 Enable. POSSIBLE VALUES: 00 - Disable, T2(ARGB) = 1,0,0,0 01 - Enable
TEX_3_ENABLE	3	none	Texture Map 3 Enable. POSSIBLE VALUES: 00 - Disable, T3(ARGB) = 1,0,0,0 01 - Enable
TEX_4_ENABLE	4	none	Texture Map 4 Enable. POSSIBLE VALUES: 00 - Disable, T4(ARGB) = 1,0,0,0 01 - Enable
TEX_5_ENABLE	5	none	Texture Map 5 Enable.

			POSSIBLE VALUES: 00 - Disable, T5(ARGB) = 1,0,0,0 01 - Enable
TEX_6_ENABLE	6	none	Texture Map 6 Enable. POSSIBLE VALUES: 00 - Disable, T6(ARGB) = 1,0,0,0 01 - Enable
TEX_7_ENABLE	7	none	Texture Map 7 Enable. POSSIBLE VALUES: 00 - Disable, T7(ARGB) = 1,0,0,0 01 - Enable
TEX_8_ENABLE	8	none	Texture Map 8 Enable. POSSIBLE VALUES: 00 - Disable, T8(ARGB) = 1,0,0,0 01 - Enable
TEX_9_ENABLE	9	none	Texture Map 9 Enable. POSSIBLE VALUES: 00 - Disable, T9(ARGB) = 1,0,0,0 01 - Enable
TEX_10_ENABLE	10	none	Texture Map 10 Enable. POSSIBLE VALUES: 00 - Disable, T10(ARGB) = 1,0,0,0 01 - Enable
TEX_11_ENABLE	11	none	Texture Map 11 Enable. POSSIBLE VALUES: 00 - Disable, T11(ARGB) = 1,0,0,0 01 - Enable
TEX_12_ENABLE	12	none	Texture Map 12 Enable. POSSIBLE VALUES: 00 - Disable, T12(ARGB) = 1,0,0,0 01 - Enable
TEX_13_ENABLE	13	none	Texture Map 13 Enable. POSSIBLE VALUES: 00 - Disable, T13(ARGB) = 1,0,0,0 01 - Enable
TEX_14_ENABLE	14	none	Texture Map 14 Enable. POSSIBLE VALUES: 00 - Disable, T14(ARGB) = 1,0,0,0

			01 - Enable
TEX_15_ENABLE	15	none	Texture Map 15 Enable. POSSIBLE VALUES: 00 - Disable, T15(ARGB) = 1,0,0,0 01 - Enable

TX:TX_FILTER0_[0-15] · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4400-0x443c			
DESCRIPTION: *Texture Filter State for Map 0*			
Field Name	Bits	Default	Description
CLAMP_S	2:0	none	Clamp mode for first texture coordinate POSSIBLE VALUES: 00 - Wrap (repeat) 01 - Mirror 02 - Clamp to last texel (0.0 to 1.0) 03 - MirrorOnce to last texel (-1.0 to 1.0) 04 - Clamp half way to border color (0.0 to 1.0) 05 - MirrorOnce half way to border color (-1.0 to 1.0) 06 - Clamp to border color (0.0 to 1.0) 07 - MirrorOnce to border color (-1.0 to 1.0)
CLAMP_T	5:3	none	Clamp mode for second texture coordinate POSSIBLE VALUES: 00 - Wrap (repeat) 01 - Mirror 02 - Clamp to last texel (0.0 to 1.0) 03 - MirrorOnce to last texel (-1.0 to 1.0) 04 - Clamp half way to border color (0.0 to 1.0) 05 - MirrorOnce half way to border color (-1.0 to 1.0) 06 - Clamp to border color (0.0 to 1.0) 07 - MirrorOnce to border color (-1.0 to 1.0)
CLAMP_R	8:6	none	Clamp mode for third texture coordinate POSSIBLE VALUES: 00 - Wrap (repeat) 01 - Mirror 02 - Clamp to last texel (0.0 to 1.0) 03 - MirrorOnce to last texel (-1.0 to 1.0) 04 - Clamp half way to border color (0.0 to 1.0) 05 - MirrorOnce half way to border color (-1.0 to 1.0) 06 - Clamp to border color (0.0 to 1.0) 07 - MirrorOnce to border color (-1.0 to 1.0)
MAG_FILTER	10:9	none	Filter used when texture is magnified POSSIBLE VALUES: 00 - Reserved 01 - Point

			02 - Linear 03 - Reserved
MIN_FILTER	12:11	none	Filter used when texture is minified POSSIBLE VALUES: 00 - Reserved 01 - Point 02 - Linear 03 - Reserved
MIP_FILTER	14:13	none	Filter used between mipmap levels POSSIBLE VALUES: 00 - None 01 - Point 02 - Linear 03 - Reserved
VOL_FILTER	16:15	none	Filter used between layers of a volume POSSIBLE VALUES: 00 - None (no filter specifed, select from MIN/MAG filters) 01 - Point 02 - Linear 03 - Reserved
MAX_MIP_LEVEL	20:17	none	LOD index of largest (finest) mipmap to use (0 is largest). Ranges from 0 to NUM_LEVELS.
Reserved	23:21	none	
ID	31:28	none	Logical id for this physical texture

TX:TX_FILTER1_[0-15] · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4440-0x447c			
DESCRIPTION: *Texture Filter State for Map 0*			
Field Name	Bits	Default	Description
CHROMA_KEY_MODE	1:0	none	Chroma Key Mode POSSIBLE VALUES: 00 - Disable 01 - ChromaKey (kill pixel if any sample matches chroma key) 02 - ChromaKeyBlend (set sample to 0 if it matches chroma key)
MC_ROUND	2	none	Bilinear rounding mode POSSIBLE VALUES: 00 - Normal rounding on all components (+0.5) 01 - MPEG4 rounding on all components (+0.25)
LOD_BIAS	12:3	none	(s4.5). Ranges from -16.0 to 15.99. Mipmap LOD bias measured in mipmap levels. Added to the signed,

			computed LOD before the LOD is clamped.
Reserved	13	none	
MC_COORD_TRUNCATE	14	none	MPEG coordinate truncation mode POSSIBLE VALUES: 00 - Dont truncate coordinate fractions. 01 - Truncate coordinate fractions to 0.0 and 0.5 for MPEG

TX:TX_FORMAT0_[0-15] · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4480-0x44bc			
DESCRIPTION: *Texture Format State for Map 0*			
Field Name	Bits	Default	Description
TXWIDTH	10:0	none	Image width - 1. The largest image is 2048 texels. When wrapping or mirroring, must be a power of 2. When mipmapping, must be a power of 2 or padded to a power of 2 in memory. Can always be non-square, except for cube maps which must be square.
TXHEIGHT	21:11	none	Image height - 1. The largest image is 2048 texels. When wrapping or mirroring, must be a power of 2. When mipmapping, must be a power of 2 or padded to a power of 2 in memory. Can always be non-square, except for cube maps which must be square.
TXDEPTH	25:22	none	LOG2(depth) of volume texture
NUM_LEVELS	29:26	none	Number of mipmap levels minus 1. Ranges from 0 to 11. Equivalent to LOD index of smallest (coarsest) mipmap to use.
PROJECTED	30	none	Specifies whether texture coords are projected. POSSIBLE VALUES: 00 - Non-Projected 01 - Projected
TXPITCH_EN	31	none	Indicates when TXPITCH should be used instead of TXWIDTH for image addressing POSSIBLE VALUES: 00 - Use TXWIDTH for image addressing 01 - Use TXPITCH for image addressing

TX:TX_FORMAT1_[0-15] · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x44c0-0x44fc			
DESCRIPTION: *Texture Format State for Map 0*			
Field Name	Bits	Default	Description
TXFORMAT	4:0	none	Texture Format. Components are numbered right to left. Parenthesis indicate typical uses of each format. POSSIBLE VALUES:

			00 - TX_FMT_8 01 - TX_FMT_16 02 - TX_FMT_4_4 03 - TX_FMT_8_8 04 - TX_FMT_16_16 05 - TX_FMT_3_3_2 06 - TX_FMT_5_6_5 07 - TX_FMT_6_5_5 08 - TX_FMT_11_11_10 09 - TX_FMT_10_11_11 10 - TX_FMT_4_4_4_4 11 - TX_FMT_1_5_5_5 12 - TX_FMT_8_8_8_8 13 - TX_FMT_2_10_10_10 14 - TX_FMT_16_16_16_16 15 - Reserved 16 - Reserved 17 - Reserved 18 - TX_FMT_Y8 19 - TX_FMT_AVYU444 20 - TX_FMT_VYUY422 21 - TX_FMT_YVYU422 22 - TX_FMT_16_MPEG 23 - TX_FMT_16_16_MPEG 24 - TX_FMT_16f 25 - TX_FMT_16f_16f 26 - TX_FMT_16f_16f_16f_16f 27 - TX_FMT_32f 28 - TX_FMT_32f_32f 29 - TX_FMT_32f_32f_32f_32f 30 - TX_FMT_W24_FP
SIGNED_COMP0	5	none	Component0 filter should interpret texel data as signed or unsigned. (Ignored for Y/YUV formats.) POSSIBLE VALUES: 00 - Component0 filter should interpret texel data as unsigned 01 - Component0 filter should interpret texel data as signed
SIGNED_COMP1	6	none	Component1 filter should interpret texel data as signed or unsigned. (Ignored for Y/YUV formats.) POSSIBLE VALUES: 00 - Component1 filter should interpret texel data as unsigned 01 - Component1 filter should interpret texel data as signed
SIGNED_COMP2	7	none	Component2 filter should interpret texel data as signed or unsigned. (Ignored for Y/YUV formats.) POSSIBLE VALUES: 00 - Component2 filter should interpret texel data as

			unsigned 01 - Component2 filter should interpret texel data as signed
SIGNED_COMP3	8	none	Component3 filter should interpret texel data as signed or unsigned. (Ignored for Y/YUV formats.) POSSIBLE VALUES: 00 - Component3 filter should interpret texel data as unsigned 01 - Component3 filter should interpret texel data as signed
SEL_ALPHA	11:9	none	Specifies swizzling for alpha channel at the input of the pixel shader. (Ignored for Y/YUV formats.) POSSIBLE VALUES: 00 - Select Texture Component0 for the Alpha Channel. 01 - Select Texture Component1 for the Alpha Channel. 02 - Select Texture Component2 for the Alpha Channel. 03 - Select Texture Component3 for the Alpha Channel. 04 - Select the value 0 for the Alpha Channel. 05 - Select the value 1 for the Alpha Channel.
SEL_RED	14:12	none	Specifies swizzling for red channel at the input of the pixel shader. (Ignored for Y/YUV formats.) POSSIBLE VALUES: 00 - Select Texture Component0 for the Red Channel. 01 - Select Texture Component1 for the Red Channel. 02 - Select Texture Component2 for the Red Channel. 03 - Select Texture Component3 for the Red Channel. 04 - Select the value 0 for the Red Channel. 05 - Select the value 1 for the Red Channel.
SEL_GREEN	17:15	none	Specifies swizzling for green channel at the input of the pixel shader. (Ignored for Y/YUV formats.) POSSIBLE VALUES: 00 - Select Texture Component0 for the Green Channel. 01 - Select Texture Component1 for the Green Channel. 02 - Select Texture Component2 for the Green Channel. 03 - Select Texture Component3 for the Green Channel. 04 - Select the value 0 for the Green Channel. 05 - Select the value 1 for the Green Channel.
SEL_BLUE	20:18	none	Specifies swizzling for blue channel at the input of the pixel shader. (Ignored for Y/YUV formats.)

			POSSIBLE VALUES: 00 - Select Texture Component0 for the Blue Channel. 01 - Select Texture Component1 for the Blue Channel. 02 - Select Texture Component2 for the Blue Channel. 03 - Select Texture Component3 for the Blue Channel. 04 - Select the value 0 for the Blue Channel. 05 - Select the value 1 for the Blue Channel.
GAMMA	21	none	Optionally remove gamma from texture before passing to shader. Only apply to 8bit or less components. POSSIBLE VALUES: 00 - Disable gamma removal 01 - Enable gamma removal
YUV_TO_RGB	23:22	none	YUV to RGB conversion mode POSSIBLE VALUES: 00 - Disable YUV to RGB conversion 01 - Enable YUV to RGB conversion (with clamp) 02 - Enable YUV to RGB conversion (without clamp)
SWAP_YUV	24	none	POSSIBLE VALUES: 00 - Disable swap YUV mode 01 - Enable swap YUV mode (hw inverts upper bit of U and V)
TEX_COORD_TYPE	26:25	none	Specifies coordinate type. POSSIBLE VALUES: 00 - 2D 01 - 3D 02 - Cube 03 - Reserved
CACHE	31:27	none	Multi-texture performance can be optimized and made deterministic by assigning textures to separate regions under sw control. POSSIBLE VALUES: 00 - WHOLE 01 - Reserved 02 - HALF_REGION_0 03 - HALF_REGION_1 04 - FOURTH_REGION_0 05 - FOURTH_REGION_1 06 - FOURTH_REGION_2 07 - FOURTH_REGION_3 08 - EIGHTH_REGION_0 09 - EIGHTH_REGION_1

			10 - EIGHTH_REGION_2 11 - EIGHTH_REGION_3 12 - EIGHTH_REGION_4 13 - EIGHTH_REGION_5 14 - EIGHTH_REGION_6 15 - EIGHTH_REGION_7 16 - SIXTEENTH_REGION_0 17 - SIXTEENTH_REGION_1 18 - SIXTEENTH_REGION_2 19 - SIXTEENTH_REGION_3 20 - SIXTEENTH_REGION_4 21 - SIXTEENTH_REGION_5 22 - SIXTEENTH_REGION_6 23 - SIXTEENTH_REGION_7 24 - SIXTEENTH_REGION_8 25 - SIXTEENTH_REGION_9 26 - SIXTEENTH_REGION_A 27 - SIXTEENTH_REGION_B 28 - SIXTEENTH_REGION_C 29 - SIXTEENTH_REGION_D 30 - SIXTEENTH_REGION_E 31 - SIXTEENTH_REGION_F

TX:TX_FORMAT2_[0-15] · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4500-0x453c			
DESCRIPTION: *Texture Format State for Map 0*			
Field Name	Bits	Default	Description
TXPITCH	13:0	none	Used instead of TXWIDTH for image addressing when TXPITCH_EN is asserted. Pitch is given as number of texels minus one. Maximum pitch is 16K texels.

TX:TX_INVALTAGS · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4100			
DESCRIPTION: *Invalidate texture cache tags*			
Field Name	Bits	Default	Description
RESERVED	31:0	none	

TX:TX_OFFSET_[0-15] · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4540-0x457c			
DESCRIPTION: *Texture Offset State for Map 0*			
Field Name	Bits	Default	Description
ENDIAN_SWAP	1:0	none	Endian Control POSSIBLE VALUES: 00 - No swap 01 - 16 bit swap 02 - 32 bit swap 03 - Half-DWORD swap

MACRO_TILE	2	none	Macro Tile Control POSSIBLE VALUES: 00 - 2KB page is linear 01 - 2KB page is tiled
MICRO_TILE	4:3	none	Micro Tile Control POSSIBLE VALUES: 00 - 32 byte cache line is linear 01 - 32 byte cache line is tiled 02 - 32 byte cache line is tiled square (only applies to 16-bit texel) 03 - Reserved
TXOFFSET	31:5	none	32-byte aligned pointer to base map

1.9 Fragment Shader Registers

US:US_ALU_ALPHA_ADDR_[0-63] · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x47c0-0x48bc			
DESCRIPTION: *This table specifies the Alpha source addresses for up to 64 ALU instruction. The ALU expects 6 source operands - three for color (rgb0, rgb1, rgb2) and three for alpha (a0, a1, a2).*			
Field Name	Bits	Default	Description
ADDR0	5:0	0x0	Specifies the identity of source operands a0, a1, and a2. Values 0 through 31 specify a location within the current pixel stack frame. Values 32 through 63 specify a constant.
ADDR1	11:6	0x0	Specifies the identity of source operands a0, a1, and a2. Values 0 through 31 specify a location within the current pixel stack frame. Values 32 through 63 specify a constant.
ADDR2	17:12	0x0	Specifies the identity of source operands a0, a1, and a2. Values 0 through 31 specify a location within the current pixel stack frame. Values 32 through 63 specify a constant.
ADDRD	22:18	0x0	Specifies the address of the pixel stack frame register to which the Alpha result of this instruction is to be written.
WMASK	23	0x0	Specifies whether or not to write the Alpha component of the result for this instruction to the pixel stack frame. POSSIBLE VALUES: 00 - NONE: No not write register. 01 - A: Write the alpha channel only.
OMASK	24	0x0	Specifies whether or not to write the Alpha component of the result of this instruction to the output fifo. POSSIBLE VALUES: 00 - NONE: No not write output. 01 - A: Write the alpha channel only.
TARGET	26:25	0x0	Specifies which frame buffer target to write to. POSSIBLE VALUES: 00 - A: Output to render target A 01 - B: Output to render target B 02 - C: Output to render target C 03 - D: Output to render target D
OMASK_W	27	0x0	Specifies whether or not to write the Alpha component of the result of this instuction to the depth output fifo. POSSIBLE VALUES: 00 - NONE: No not write output to w. 01 - A: Write the alpha channel only.
STAT_WE	31:28	0x0	Specifies which components (R,G,B,A) contribute to the stat count (see performance counter field in US_CONFIG).

US:US_ALU_ALPHA_INST_[0-63] · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x49c0-0x4abc			
DESCRIPTION: *ALU Alpha Instruction*			
Field Name	Bits	Default	Description
SEL_A	4:0	0x0	Specifies the operand and component select for inputs A, B, and C. POSSIBLE VALUES: 00 - src0.r 01 - src0.g 02 - src0.b 03 - src1.r 04 - src1.g 05 - src1.b 06 - src2.r 07 - src2.g 08 - src2.b 09 - src0.a 10 - src1.a 11 - src2.a 12 - srcp.r 13 - srcp.g 14 - srcp.b 15 - srcp.a 16 - 0.0 17 - 1.0 18 - 0.5
MOD_A	6:5	0x0	Specifies the modifier for inputs A, B, and C. POSSIBLE VALUES: 00 - NOP: Do not modify input 01 - NEG: Negate input 02 - ABS: Take absolute value of input 03 - NAB: Take negative absolute value of input
SEL_B	11:7	0x0	Specifies the operand and component select for inputs A, B, and C. POSSIBLE VALUES: 00 - src0.r 01 - src0.g 02 - src0.b 03 - src1.r 04 - src1.g 05 - src1.b 06 - src2.r 07 - src2.g 08 - src2.b 09 - src0.a 10 - src1.a 11 - src2.a 12 - srcp.r 13 - srcp.g

			14 - srcp.b 15 - srcp.a 16 - 0.0 17 - 1.0 18 - 0.5
MOD_B	13:12	0x0	Specifies the modifier for inputs A, B, and C. POSSIBLE VALUES: 00 - NOP: Do not modify input 01 - NEG: Negate input 02 - ABS: Take absolute value of input 03 - NAB: Take negative absolute value of input
SEL_C	18:14	0x0	Specifies the operand and component select for inputs A, B, and C. POSSIBLE VALUES: 00 - src0.r 01 - src0.g 02 - src0.b 03 - src1.r 04 - src1.g 05 - src1.b 06 - src2.r 07 - src2.g 08 - src2.b 09 - src0.a 10 - src1.a 11 - src2.a 12 - srcp.r 13 - srcp.g 14 - srcp.b 15 - srcp.a 16 - 0.0 17 - 1.0 18 - 0.5
MOD_C	20:19	0x0	Specifies the modifier for inputs A, B, and C. POSSIBLE VALUES: 00 - NOP: Do not modify input 01 - NEG: Negate input 02 - ABS: Take absolute value of input 03 - NAB: Take negative absolute value of input
SRCP_OP	22:21	0x0	Specifies how the pre-subtract value (SRCP) is computed POSSIBLE VALUES: 00 - 1.0-2.0*A0 01 - A1-A0 02 - A1+A0 03 - 1.0-A0
OP	26:23	0x0	Specifies the operand for this instruction.

			POSSIBLE VALUES: 00 - OP_MAD: Result = A*B + C 01 - OP_DP: Result = dot product from RGB ALU 02 - OP_MIN: Result = min(A,B) 03 - OP_MAX: Result = max(A,B) 04 - reserved 05 - OP_CND: Result = cnd(A,B,C) = (C>0.5)?A:B 06 - OP_CMP: Result = cmp(A,B,C) = (C>=0.0)?A:B 07 - OP_FRC: Result = fractional(A) 08 - OP_EX2: Result = 2^^A 09 - OP_LN2: Result = log2(A) 10 - OP_RCP: Result = 1/A 11 - OP_RSQ: Result = 1/sqrt(A)
OMOD	29:27	0x0	Specifies the output modifier for this instruction. POSSIBLE VALUES: 00 - Result 01 - Result *2 02 - Result *4 03 - Result *8 04 - Result / 2 05 - Result / 4 06 - Result / 8 07 - Reserved
CLAMP	30	0x0	Specifies clamp mode for this instruction. POSSIBLE VALUES: 00 - Do not clamp output. 01 - Clamp output to the range [0,1].

US:US_ALU_RGB_ADDR_[0-63] · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x46c0-0x47bc			
DESCRIPTION: *This table specifies the RGB source and destination addresses for up to 64 ALU instructions. The ALU expects 6 source operands - three for color (rgb0, rgb1, rgb2) and three for alpha (a0, a1, a2).*			
Field Name	Bits	Default	Description
ADDR0	5:0	0x0	Specifies the identity of source operands rgb0, rgb1, and rgb2. Values 0 through 31 specify a location within the current pixel stack frame. Values 32 through 63 specify a constant.
ADDR1	11:6	0x0	Specifies the identity of source operands rgb0, rgb1, and rgb2. Values 0 through 31 specify a location within the current pixel stack frame. Values 32 through 63 specify a constant.
ADDR2	17:12	0x0	Specifies the identity of source operands rgb0, rgb1, and rgb2. Values 0 through 31 specify a location within the current pixel stack frame. Values 32 through 63 specify a constant.
ADDRD	22:18	0x0	Specifies the address of the pixel stack frame register to

			which the RGB result of this instruction is to be written.
WMASK	25:23	0x0	Specifies which of the R, G, and B components of the result of this instruction are written to the pixel stack frame. POSSIBLE VALUES: 00 - NONE: No not write any output. 01 - R: Write the red channel only. 02 - G: Write the green channel only. 03 - RG: Write the red and green channels. 04 - B: Write the blue channel only. 05 - RB: Write the red and blue channels. 06 - GB: Write the green and blue channels. 07 - RGB: Write the red, green, and blue channels.
OMASK	28:26	0x0	Specifies which of the R, G, and B components of the result of this instruction are written to the output fifo. POSSIBLE VALUES: 00 - NONE: No not write any output. 01 - R: Write the red channel only. 02 - G: Write the green channel only. 03 - RG: Write the red and green channels. 04 - B: Write the blue channel only. 05 - RB: Write the red and blue channels. 06 - GB: Write the green and blue channels. 07 - RGB: Write the red, green, and blue channels.
TARGET	30:29	0x0	Specifies which frame buffer target to write to. POSSIBLE VALUES: 00 - A: Output to render target A 01 - B: Output to render target B 02 - C: Output to render target C 03 - D: Output to render target D

US:US_ALU_RGB_INST_[0-63] · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x48c0-0x49bc			
DESCRIPTION: *ALU RGB Instruction*			
Field Name	Bits	Default	Description
SEL_A	4:0	0x0	Specifies the operand and component select for inputs A, B, and C. POSSIBLE VALUES: 00 - src0.rgb 01 - src0.rrr 02 - src0.ggg 03 - src0.bbb 04 - src1.rgb 05 - src1.rrr 06 - src1.ggg 07 - src1.bbb

			08 - src2.rgb 09 - src2.rrr 10 - src2.ggg 11 - src2.bbb 12 - src0.aaa 13 - src1.aaa 14 - src2.aaa 15 - srcp.rgb 16 - srcp.rrr 17 - srcp.ggg 18 - srcp.bbb 19 - srcp.aaa 20 - 0.0 21 - 1.0 22 - 0.5 23 - src0.gbr 24 - src1.gbr 25 - src2.gbr 26 - src0.brg 27 - src1.brg 28 - src2.brg 29 - src0.abg 30 - src1.abg 31 - src2.abg
MOD_A	6:5	0x0	Specifies the modifier for inputs A, B, and C. POSSIBLE VALUES: 00 - NOP: Do not modify input 01 - NEG: Negate input 02 - ABS: Take absolute value of input 03 - NAB: Take negative absolute value of input
SEL_B	11:7	0x0	Specifies the operand and component select for inputs A, B, and C. POSSIBLE VALUES: 00 - src0.rgb 01 - src0.rrr 02 - src0.ggg 03 - src0.bbb 04 - src1.rgb 05 - src1.rrr 06 - src1.ggg 07 - src1.bbb 08 - src2.rgb 09 - src2.rrr 10 - src2.ggg 11 - src2.bbb 12 - src0.aaa 13 - src1.aaa 14 - src2.aaa 15 - srcp.rgb 16 - srcp.rrr

			17 - srcp.ggg 18 - srcp.bbb 19 - srcp.aaa 20 - 0.0 21 - 1.0 22 - 0.5 23 - src0.gbr 24 - src1.gbr 25 - src2.gbr 26 - src0.brg 27 - src1.brg 28 - src2.brg 29 - src0.abg 30 - src1.abg 31 - src2.abg
MOD_B	13:12	0x0	Specifies the modifier for inputs A, B, and C. POSSIBLE VALUES: 00 - NOP: Do not modify input 01 - NEG: Negate input 02 - ABS: Take absolute value of input 03 - NAB: Take negative absolute value of input
SEL_C	18:14	0x0	Specifies the operand and component select for inputs A, B, and C. POSSIBLE VALUES: 00 - src0.rgb 01 - src0.rrr 02 - src0.ggg 03 - src0.bbb 04 - src1.rgb 05 - src1.rrr 06 - src1.ggg 07 - src1.bbb 08 - src2.rgb 09 - src2.rrr 10 - src2.ggg 11 - src2.bbb 12 - src0.aaa 13 - src1.aaa 14 - src2.aaa 15 - srcp.rgb 16 - srcp.rrr 17 - srcp.ggg 18 - srcp.bbb 19 - srcp.aaa 20 - 0.0 21 - 1.0 22 - 0.5 23 - src0.gbr 24 - src1.gbr 25 - src2.gbr

			26 - src0.brg 27 - src1.brg 28 - src2.brg 29 - src0.abg 30 - src1.abg 31 - src2.abg
MOD_C	20:19	0x0	Specifies the modifier for inputs A, B, and C. POSSIBLE VALUES: 00 - NOP: Do not modify input 01 - NEG: Negate input 02 - ABS: Take absolute value of input 03 - NAB: Take negative absolute value of input
SRCP_OP	22:21	0x0	Specifies how the pre-subtract value (SRCP) is computed POSSIBLE VALUES: 00 - 1.0-2.0*RGB0 01 - RGB1-RGB0 02 - RGB1+RGB0 03 - 1.0-RGB0
OP	26:23	0x0	Specifies the operand for this instruction. POSSIBLE VALUES: 00 - OP_MAD: Result = A*B + C 01 - OP_DP3: Result = A.r*B.r + A.g*B.g + A.b*B.b 02 - OP_DP4: Result = A.r*B.r + A.g*B.g + A.b*B.b + A.a*B.a 03 - OP_D2A: Result = A.r*B.r + A.g*B.g + C.b 04 - OP_MIN: Result = min(A,B) 05 - OP_MAX: Result = max(A,B) 06 - reserved 07 - OP_CND: Result = cnd(A,B,C) = (C>0.5)?A:B 08 - OP_CMP: Result = cmp(A,B,C) = (C>=0.0)?A:B 09 - OP_FRC: Result = frac(A) 10 - OP_SOP: Result = ex2,ln2,rcp,rsq from Alpha ALU
OMOD	29:27	0x0	Specifies the output modifier for this instruction. POSSIBLE VALUES: 00 - Result 01 - Result *2 02 - Result *4 03 - Result *8 04 - Result / 2 05 - Result / 4 06 - Result / 8 07 - Reserved
CLAMP	30	0x0	Specifies clamp mode for this instruction. POSSIBLE VALUES:

			00 - Do not clamp output. 01 - Clamp output to the range [0,1].
NOP	31	0x0	Specifies whether to insert a NOP instruction after this. This would get specified in order to meet dependency requirements for the pre-subtract inputs. POSSIBLE VALUES: 00 - Do not insert NOP instruction after this one 01 - Insert a NOP instruction after this one

US:US_CODE_ADDR_[0-3] · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4610-0x461c			
DESCRIPTION: *Code Address for Indirection Levels 0 to 3*			
Field Name	Bits	Default	Description
ALU_START	5:0	0x0	Specifies the start address of the ALU microcode segment associated with the current indirection level (0:63)
ALU_SIZE	11:6	0x0	Specifies the size of the ALU microcode segment associated with the current indirection level (1:64)
TEX_START	16:12	0x0	Specifies the start address of the texture microcode segment associated with the current indirection level (0:31)
TEX_SIZE	21:17	0x0	Specifies the size of the texture microcode segment associated with the current indirection level (1:32)
RGBA_OUT	22	0x0	Indicates at least one RGBA output instruction at this level
W_OUT	23	0x0	Indicates at least one W output instruction at this level

US:US_CODE_OFFSET · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4608			
DESCRIPTION: *Specifies the offset and size for the ALU and Texture micrcode. These values are used to support relocatable code, and to support register writes to the code store without requiring a pipeline flush.*			
Field Name	Bits	Default	Description
ALU_OFFSET	5:0	0x0	Specifies the offset for the ALU code. This value is added to the ALU_START field in the US_CODE_ADDR registers (0:63)
ALU_SIZE	12:6	0x0	Specifies the total size for the ALU code for all levels (0:64)
TEX_OFFSET	17:13	0x0	Specifies the offset for the Texture code. This value is added to the TEX_START field in the US_CODE_ADDR registers (0:31)
TEX_SIZE	23:18	0x0	Specifies the total size for the Texture code for all levels (0:32)

US:US_CONFIG · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4600

DESCRIPTION: *Shader Configuration*			
Field Name	Bits	Default	Description
NLEVEL	2:0	0x0	Specifies the valid indirection levels. POSSIBLE VALUES: 00 - Level 3 only (normal DX7-style texturing) 01 - Levels 2 and 3 (DX8-style bump mapping) 02 - Levels 1, 2, and 3 03 - Levels 0, 1, 2, and 3
FIRST_TEX	3	0x0	Specifies whether or not the texture code for the first valid level is enabled POSSIBLE VALUES: 00 - Disabled 01 - Enabled
Reserved	8:4	0x0	
Reserved	13:9	0x0	
Reserved	18:14	0x0	
Reserved	23:19	0x0	

US:US_OUT_FMT_[0-3] · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x46a4-0x46b0			
DESCRIPTION: *Specifies how the shader output is written to the fog unit for each of up to four render targets*			
Field Name	Bits	Default	Description
OUT_FMT	4:0	0x0	Specifies the number and size of components POSSIBLE VALUES: 00 - C4_8 (S/U) 01 - C4_10 (U) 02 - C4_10_GAMMA - (U) 03 - C_16 - (S/U) 04 - C2_16 - (S/U) 05 - C4_16 - (S/U) 06 - C_16_MPEG - (S) 07 - C2_16_MPEG - (S) 08 - C2_4 - (U) 09 - C_3_3_2 - (U) 10 - C_6_5_6 - (S/U) 11 - C_11_11_10 - (S/U) 12 - C_10_11_11 - (S/U) 13 - C_2_10_10_10 - (S/U) 14 - reserved 15 - UNUSED - Render target is not used 16 - C_16_FP - (S10E5) 17 - C2_16_FP - (S10E5) 18 - C4_16_FP - (S10E5) 19 - C_32_FP - (S23E8) 20 - C2_32_FP - (S23E8)

			21 - C4_32_FP - (S23E8)
C0_SEL	9:8	0x0	Specifies the source for components C0, C1, C2, C3 POSSIBLE VALUES: 00 - Alpha 01 - Red 02 - Green 03 - Blue
C1_SEL	11:10	0x0	Specifies the source for components C0, C1, C2, C3 POSSIBLE VALUES: 00 - Alpha 01 - Red 02 - Green 03 - Blue
C2_SEL	13:12	0x0	Specifies the source for components C0, C1, C2, C3 POSSIBLE VALUES: 00 - Alpha 01 - Red 02 - Green 03 - Blue
C3_SEL	15:14	0x0	Specifies the source for components C0, C1, C2, C3 POSSIBLE VALUES: 00 - Alpha 01 - Red 02 - Green 03 - Blue
OUT_SIGN	19:16	0x0	Mask specifying whether components C3, C2, C1 and C0 are signed (C4_8, C_16, C2_16 and C4_16 formats only)

US:US_PIXSIZE · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4604			
DESCRIPTION: *Shader pixel size. This register specifies the size and partitioning of the current pixel stack frame*			
Field Name	Bits	Default	Description
PIX_SIZE	4:0	0x0	Specifies the total size of the current pixel stack frame (1:32)

US:US_TEX_INST_[0-31] · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4620-0x469c			
DESCRIPTION: *Texture Instruction*			
Field Name	Bits	Default	Description
SRC_ADDR	4:0	0x0	Specifies the location (within the shader pixel stack frame) of the texture address for this instruction
DST_ADDR	10:6	0x0	Specifies the location (within the shader pixel stack

			frame) of the returned texture data for this instruction
TEX_ID	14:11	0x0	Specifies the id of the texture map used for this instruction
INST	17:15	0x0	Specifies the operation taking place for this instruction POSSIBLE VALUES: 00 - NOP: Do nothing 01 - LD: Do Texture Lookup (S,T,R) 02 - TEXKILL: Kill pixel if any component is < 0 03 - PROJ: Do projected texture lookup (S/Q,T/Q,R/Q) 04 - LODBIAS: Do texture lookup with lod bias
OMOD	18	0x0	unused

US:US_W_FMT · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x46b4			
DESCRIPTION: *Specifies the source and format for the Depth (W) value output by the shader*			
Field Name	Bits	Default	Description
W_FMT	1:0	0x0	Format for W POSSIBLE VALUES: 00 - W0 - W is always zero 01 - W24 - 24-bit fixed point 02 - W24_FP - 24-bit floating point 03 - Reserved
W_SRC	2	0x0	Source for W POSSIBLE VALUES: 00 - WSRC_US - W comes from shader instruction 01 - WSRC_RAS - W comes from rasterizer

US:US_ALU_CONST_A_[0-31] · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4c0c-0x4dfc			
DESCRIPTION: *Shader Constant Color 0 Alpha Component*			
Field Name	Bits	Default	Description
KA	23:0	0x0	Specifies the alpha component; (S16E7) fixed format.

US:US_ALU_CONST_B_[0-31] · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4c08-0x4df8			
DESCRIPTION: *Shader Constant Color 0 Blue Component*			
Field Name	Bits	Default	Description
KB	23:0	0x0	Specifies the blue component; (S16E7) fixed format.

US:US_ALU_CONST_G_[0-31] · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4c04-0x4df4

DESCRIPTION: *Shader Constant Color 0 Green Component*			
Field Name	Bits	Default	Description
KG	23:0	0x0	Specifies the green component; (S16E7) fixed format.

US:US_ALU_CONST_R_[0-31] · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4c00-0x4df0			
DESCRIPTION: *Shader Constant Color 0 Red Component*			
Field Name	Bits	Default	Description
KR	23:0	0x0	Specifies the red component; (S16E7) fixed format.

1.10 Vertex Registers

VAP:VAP_CLIP_CNTL · [R/W] · 32 bits · Access: 32 · MMReg:0x221c			
DESCRIPTION: *Control Bits for User Clip Planes and Clipping*			
Field Name	Bits	Default	Description
UCP_ENA_0	0	0x0	Enable User Clip Plane 0
UCP_ENA_1	1	0x0	Enable User Clip Plane 1
UCP_ENA_2	2	0x0	Enable User Clip Plane 2
UCP_ENA_3	3	0x0	Enable User Clip Plane 3
UCP_ENA_4	4	0x0	Enable User Clip Plane 4
UCP_ENA_5	5	0x0	Enable User Clip Plane 5
PS_UCP_MODE	15:14	0x0	0 = Cull using distance from center of point 1 = Cull using radius-based distance from center of point 2 = Cull using radius-based distance from center of point, Expand and Clip on intersection 3 = Always expand and clip as trifan
CLIP_DISABLE	16	0x0	Disables clip code generation and clipping process for TCL
UCP_CULL_ONLY_ENA	17	0x0	Cull Primitives against UCPS, but don`t clip
BOUNDARY_EDGE_FLAG_ENA	18	0x0	If set, boundary edges are highlighted, else they are not highlighted

VAP:VAP_CNTL · [R/W] · 32 bits · Access: 32 · MMReg:0x2080			
DESCRIPTION: *Vertex Assembler/Processor Control Register*			
Field Name	Bits	Default	Description
PVS_NUM_SLOTS	3:0	0x0	Specifies the number of vertex slots to be used in the VAP PVS process. A slot represents a single vertex storage location1 across multiple engines (one vertex per engine). By decreasing the number of slots, there is more memory for each vertex, but less parallel processing. Similarly, by increasing the number of slots, thre is less memory per vertex but more vertices being processed in parallel.
PVS_NUM_CNTLRS	7:4	0x0	Specifies the maximum number of controllers to be processing in parallel. In general should be set to max value of TBD. Can be changed for performance analysis.
PVS_NUM_FPUS	11:8	0x0	Specifies the number of Floating Point Units (Vector/Math Engines) to use when processing vertices.
VF_MAX_VTX_NUM	21:18	0x9	This field controls the number of vertices that the vertex fetcher manages for the TCL and Setup Vertex Storage memories (and therefore the number of vertices that can be re-used). This value should be set to 12 for most operation, This number may be modified for performance evaluation. The value is the maximum vertex number used which is one less than the number of

			vertices (i.e. a 12 means 13 vertices will be used)
DX_CLIP_SPACE_DEF	22	0x0	Clip space is defined as: 0: -W < X < W, -W < Y < W, -W < Z < W (OpenGL Definition) 1: -W < X < W, -W < Y < W, 0 < Z < W (DirectX Definition)

VAP:VAP_CNTL_STATUS · [R/W] · 32 bits · Access: 32 · MMReg:0x2140			
DESCRIPTION: *Vertex Assemblen/Processor Control Status*			
Field Name	Bits	Default	Description
VC_SWAP	1:0	0x0	Endian-Swap Control. 0 = No swap 1 = 16-bit swap: 0xAABBCCDD becomes 0xBBAADDCC 2 = 32-bit swap: 0xAABBCCDD becomes 0xDDCCBBAA 3 = Half-dword swap: 0xAABBCCDD becomes 0xCCDDAABB Default = 0
PVS_BYPASS	8	0x0	The TCL engine is logically or physically removed from the circuit.
PVS_BUSY (Access: R)	11	0x0	Transform/Clip/Light (TCL) Engine is Busy. Read-only.
VS_BUSY (Access: R)	24	0x0	Vertex Store is Busy. Read-only.
RCP_BUSY (Access: R)	25	0x0	Reciprocal Engine is Busy. Read-only.
VTE_BUSY (Access: R)	26	0x0	ViewPort Transform Engine is Busy. Read-only.
MIU_BUSY (Access: R)	27	0x0	Memory Interface Unit is Busy. Read-only.
VC_BUSY (Access: R)	28	0x0	Vertex Cache is Busy. Read-only.
VF_BUSY (Access: R)	29	0x0	Vertex Fetcher is Busy. Read-only.
REGPIPE_BUSY (Access: R)	30	0x0	Register Pipeline is Busy. Read-only.
VAP_BUSY (Access: R)	31	0x0	VAP Engine is Busy. Read-only.

VAP:VAP_GB_HORZ_CLIP_ADJ · [R/W] · 32 bits · Access: 32 · MMReg:0x2228			
DESCRIPTION: *Horizontal Guard Band Clip Adjust Register*			
Field Name	Bits	Default	Description
DATA_REGISTER	31:0	0x0	32-bit floating point value. Should be set to 1.0 for no guard band.

VAP:VAP_GB_HORZ_DISC_ADJ · [R/W] · 32 bits · Access: 32 · MMReg:0x222c			
DESCRIPTION: *Horizontal Guard Band Discard Adjust Register*			
Field Name	Bits	Default	Description
DATA_REGISTER	31:0	0x0	32-bit floating point value. Should be set to 1.0 for no guard band.

VAP:VAP_GB_VERT_CLIP_ADJ · [R/W] · 32 bits · Access: 32 · MMReg:0x2220			
DESCRIPTION: *Vertical Guard Band Clip Adjust Register*			
Field Name	Bits	Default	Description
DATA_REGISTER	31:0	0x0	32-bit floating point value. Should be set to 1.0 for no guard band.

VAP:VAP_GB_VERT_DISC_ADJ · [R/W] · 32 bits · Access: 32 · MMReg:0x2224			
DESCRIPTION: *Vertical Guard Band Discard Adjust Register*			
Field Name	Bits	Default	Description
DATA_REGISTER	31:0	0x0	32-bit floating point value. Should be set to 1.0 for no guard band.

VAP:VAP_OUT_VTX_FMT_0 · [R/W] · 32 bits · Access: 32 · MMReg:0x2090			
DESCRIPTION: *VAP Out/GA Vertex Format Register 0*			
Field Name	Bits	Default	Description
VTX_POS_PRESENT	0	0x0	Output the Position Vector
VTX_COLOR_0_PRESENT	1	0x0	Output Color 0 Vector
VTX_COLOR_1_PRESENT	2	0x0	Output Color 1 Vector
VTX_COLOR_2_PRESENT	3	0x0	Output Color 2 Vector
VTX_COLOR_3_PRESENT	4	0x0	Output Color 3 Vector
VTX_PT_SIZE_PRESENT	16	0x0	Output Point Size Vector

VAP:VAP_OUT_VTX_FMT_1 · [R/W] · 32 bits · Access: 32 · MMReg:0x2094			
DESCRIPTION: *VAP Out/GA Vertex Format Register 1*			
Field Name	Bits	Default	Description
TEX_0_COMP_CNT	2:0	0x0	Number of words in texture 0 = Not Present 1 = 1 component 2 = 2 components 3 = 3 components 4 = 4 components

TEX_1_COMP_CNT	5:3	0x0	Number of words in texture 0 = Not Present 1 = 1 component 2 = 2 components 3 = 3 components 4 = 4 components
TEX_2_COMP_CNT	8:6	0x0	Number of words in texture 0 = Not Present 1 = 1 component 2 = 2 components 3 = 3 components 4 = 4 components
TEX_3_COMP_CNT	11:9	0x0	Number of words in texture 0 = Not Present 1 = 1 component 2 = 2 components 3 = 3 components 4 = 4 components
TEX_4_COMP_CNT	14:12	0x0	Number of words in texture 0 = Not Present 1 = 1 component 2 = 2 components 3 = 3 components 4 = 4 components
TEX_5_COMP_CNT	17:15	0x0	Number of words in texture 0 = Not Present 1 = 1 component 2 = 2 components 3 = 3 components 4 = 4 components
TEX_6_COMP_CNT	20:18	0x0	Number of words in texture 0 = Not Present 1 = 1 component 2 = 2 components 3 = 3 components 4 = 4 components
TEX_7_COMP_CNT	23:21	0x0	Number of words in texture 0 = Not Present 1 = 1 component 2 = 2 components 3 = 3 components 4 = 4 components

VAP:VAP_PORT_DATA[0-15] · [W] · 32 bits · Access: 32 · MMReg:0x2000-0x203c			
DESCRIPTION: *Setup Engine Data Port 0 through 15.*			
Field Name	Bits	Default	Description
DATAPORT0 (master with mirrors)	31:0	0x0	1st of 16 consecutive dwords for writing vertex data information.

			Write-only.

VAP:VAP_PORT_DATA_IDX_128 · [W] · 32 bits · Access: 32 · MMReg:0x20b8			
DESCRIPTION: *128-bit Data Port for Indexed Primitives.*			
Field Name	Bits	Default	Description
DATA_IDX_PORT_128	31:0	0x0	128-bit Data Port for Indexed Primitives. Write-only.

VAP:VAP_PORT_IDX[0-15] · [W] · 32 bits · Access: 32 · MMReg:0x2040-0x207c			
DESCRIPTION: *Setup Engine Index Port 0 through 15.*			
Field Name	Bits	Default	Description
IDXPORT0 (master with mirrors)	31:0	0x0	1st of 16 consecutive dwords for writing vertex index information, in the format of: 15:0 Index 0 31:16 Index 1 Write-only.

VAP:VAP_PROG_STREAM_CNTL_[0-7] · [R/W] · 32 bits · Access: 32 · MMReg:0x2150-0x216c			
DESCRIPTION: *Programmable Stream Control Word 0*			
Field Name	Bits	Default	Description
DATA_TYPE_0	3:0	0x0	The data type for element 0 0 = FLOAT_1 (Single IEEE Float) 1 = FLOAT_2 (2 IEEE floats) 2 = FLOAT_3 (3 IEEE Floats) 3 = FLOAT_4 (4 IEEE Floats) 4 = BYTE * (1 DWORD w 4 8-bit fixed point values) (X = [7:0], Y = [15:8], Z = [23:16], W = [31:24]) 5 = D3DCOLOR * (Same as BYTE except has X->Z,Z->X swap for D3D color def) (Z = [7:0], Y = [15:8], X = [23:16], W = [31:24]) 6 = SHORT_2 * (1 DWORD with 2 16-bit fixed point values) (X = [15:0], Y = [31:16], Z = 0.0, W = 1.0) 7 = SHORT_4 * (2 DWORDS with 4(2 per dword) 16-bit fixed point values) (X = DW0 [15:0], Y = DW0 [31:16], Z = DW1 [15:0], W = DW1 [31:16]) 8 = VECTOR_3_TTT * (1 DWORD with 3 10-bit fixed point values) (X = [9:0], Y = [19:10], Z = [29:20], W = 1.0) 9 = VECTOR_3_EET * (1 DWORD with 2 11-bit and 1 10-bit fixed point values) (X = [10:0], Y = [21:11], Z = [31:22], W = 1.0) * These data types use the SIGNED and NORMALIZE flags described below.

SKIP_DWORDS_0	7:4	0x0	The number of DWORDS to skip (discard) after processing the current element.
DST_VEC_LOC_0	12:8	0x0	The vector address in the input memory to write this element
LAST_VEC_0	13	0x0	If set, indicates the last vector of the current vertex stream
SIGNED_0	14	0x0	Determines whether fixed point data types are unsigned (0) or 2`s complement signed (1) data types. See NORMALIZE for complete description of affect
NORMALIZE_0	15	0x0	Determines whether the fixed to floating point conversion will normalize the value (i.e. fixed point value is all fractional bits) or not (i.e. fixed point value is all integer bits). This table describes the fixed to float conversion results SIGNED NORMALIZE FLT RANGE 0 0 0.0 - (2^n - 1) (i.e. 8-bit -> 0.0 - 255.0) 0 1 0.0 - 1.0 1 0 -2^(n-1) - (2^(n-1) - 1) (i.e. 8-bit -> -128.0 - 127.0) 1 1 -1.0 - 1.0 where n is the number of bits in the associated fixed point value For signed, normalize conversion, since the fixed point range is not evenly distributed around 0, there are 3 different methods supported by R300. See the VAP_PSC_SGN_NORM_CNTL description for details.
DATA_TYPE_1	19:16	0x0	
SKIP_DWORDS_1	23:20	0x0	See SKIP_DWORDS_0
DST_VEC_LOC_1	28:24	0x0	See DST_VEC_LOC_0
LAST_VEC_1	29	0x0	See LAST_VEC_0
SIGNED_1	30	0x0	See SIGNED_0
NORMALIZE_1	31	0x0	See NORMALIZE_0

VAP:VAP_PROG_STREAM_CNTL_EXT_[0-7] · [R/W] · 32 bits · Access: 32 · MMReg:0x21e0-0x21fc			
DESCRIPTION: *Programmable Stream Control Extension Word 0*			
Field Name	Bits	Default	Description
SWIZZLE_SELECT_X_0	2:0	0x0	X-Component Swizzle Select 0 = SELECT_X 1 = SELECT_Y 2 = SELECT_Z 3 = SELECT_W 4 = SELECT_FP_ZERO (Floating Point 0.0) 5 = SELECT_FP_ONE (Floating Point 1.0) 6,7 RESERVED
SWIZZLE_SELECT_Y_0	5:3	0x0	Y-Component Swizzle Select (See Above)
SWIZZLE_SELECT_Z_0	8:6	0x0	Z-Component Swizzle Select (See Above)
SWIZZLE_SELECT_W_0	11:9	0x0	W-Component Swizzle Select (See Above)

WRITE_ENA_0	15:12	0x0	4-bit write enable. Bit 0 maps to X Bit 1 maps to Y Bit 2 maps to Z Bit 3 maps to W
SWIZZLE_SELECT_X_1	18:16	0x0	See SWIZZLE_SELECT_X_0
SWIZZLE_SELECT_Y_1	21:19	0x0	See SWIZZLE_SELECT_Y_0
SWIZZLE_SELECT_Z_1	24:22	0x0	See SWIZZLE_SELECT_Z_0
SWIZZLE_SELECT_W_1	27:25	0x0	See SWIZZLE_SELECT_W_0
WRITE_ENA_1	31:28	0x0	See WRITE_ENA_0

VAP:VAP_PSC_SGN_NORM_CNTL · [R/W] · 32 bits · Access: 32 · MMReg:0x21dc			
DESCRIPTION: *Programmable Stream Control Signed Normalize Control*			
Field Name	Bits	Default	Description
SGN_NORM_METHOD_0	1:0	0x0	There are 3 methods of normalizing signed numbers: 0: SGN_NORM_ZERO : value / (2^(n-1)-1), so -128/127 will be less that -1.0, -127/127 will yeild -1.0, 0/127 will yield 0, and 127/127 will yield 1.0 for 8-bit numbers. 1: SGN_NORM_ZERO_CLAMP_MINUS_ONE: Same as SGN_NORM_ZERO except -128/127 will yield -1.0 for 8-bit numbers. 2: SGN_NORM_NO_ZERO: (2 * value + 1)/2^n, so -128 will yield -255/255 = -1.0, 127 will yield 255/255 = 1.0, but 0 will yield 1/255 != 0.
SGN_NORM_METHOD_1	3:2	0x0	See SGN_NORM_METHOD_0
SGN_NORM_METHOD_2	5:4	0x0	See SGN_NORM_METHOD_0
SGN_NORM_METHOD_3	7:6	0x0	See SGN_NORM_METHOD_0
SGN_NORM_METHOD_4	9:8	0x0	See SGN_NORM_METHOD_0
SGN_NORM_METHOD_5	11:10	0x0	See SGN_NORM_METHOD_0
SGN_NORM_METHOD_6	13:12	0x0	See SGN_NORM_METHOD_0
SGN_NORM_METHOD_7	15:14	0x0	See SGN_NORM_METHOD_0
SGN_NORM_METHOD_8	17:16	0x0	See SGN_NORM_METHOD_0
SGN_NORM_METHOD_9	19:18	0x0	See SGN_NORM_METHOD_0
SGN_NORM_METHOD_10	21:20	0x0	See SGN_NORM_METHOD_0
SGN_NORM_METHOD_11	23:22	0x0	See SGN_NORM_METHOD_0
SGN_NORM_METHOD_12	25:24	0x0	See SGN_NORM_METHOD_0
SGN_NORM_METHOD_13	27:26	0x0	See SGN_NORM_METHOD_0
SGN_NORM_METHOD_14	29:28	0x0	See SGN_NORM_METHOD_0
SGN_NORM_METHOD_15	31:30	0x0	See SGN_NORM_METHOD_0

VAP:VAP_PVS_CODE_CNTL_0 · [R/W] · 32 bits · Access: 32 · MMReg:0x22d0

DESCRIPTION: *Programmable Vertex Shader Code Control Register 0*			
Field Name	Bits	Default	Description
PVS_FIRST_INST	9:0	0x0	First Instruction to Execute in PVS.
PVS_XYZW_VALID_INST	19:10	0x0	The PVS Instruction which updates the clip coordinate position for the last time. This value is used to lower the processing priority while trivial clip and back-face culling decisions are made. This field must be set to valid instruction.
PVS_LAST_INST	29:20	0x0	Last Instruction (Inclusive) for the PVS to execute.

VAP:VAP_PVS_CODE_CNTL_1 · [R/W] · 32 bits · Access: 32 · MMReg:0x22d8			
DESCRIPTION: *Programmable Vertex Shader Code Control Register 1*			
Field Name	Bits	Default	Description
PVS_LAST_VTX_SRC_INST	9:0	0x0	The PVS Instruction which uses the Input Vertex Memory for the last time. This value is used to free up the Input Vertex Slots ASAP. This field must be set to a valid instruction.

VAP:VAP_PVS_CONST_CNTL · [R/W] · 32 bits · Access: 32 · MMReg:0x22d4			
DESCRIPTION: *Programmable Vertex Shader Constant Control Register*			
Field Name	Bits	Default	Description
PVS_CONST_BASE_OFFSET	7:0	0x0	Vector Offset into PVS constant memory to the start of the constants for the current shader
PVS_MAX_CONST_ADDR	23:16	0x0	The maximum constant address which should be generated by the shader (Inst Const Addr + Addr Register). If the address which is generated by the shader is outside the range of 0 to PVS_MAX_CONST_ADDR, then (0,0,0,0) is returned as the source operand data.

VAP:VAP_PVS_FLOW_CNTL_ADDRS_[0-15] · [R/W] · 32 bits · Access: 32 · MMReg:0x2230-0x226c			
DESCRIPTION: *Programmable Vertex Shader Flow Control Addresses Register 0*			
Field Name	Bits	Default	Description
PVS_FC_ACT_ADRS_0	7:0	0x0	This field defines the last PVS instruction to execute prior to the control flow redirection. JUMP - The last instruction executed prior to the jump LOOP - The last instruction executed prior to the loop (init loop counter/inc) JSR - The last instruction executed prior to the jump to the subroutine.
PVS_FC_LOOP_CNT_JMP_INST_0	15:8	0x0	This field has multiple definitions as follows: JUMP - The instruction address to jump to. LOOP - The loop count. *Note loop count of 0 must be replaced by a jump.

			JSR - The instruction address to jump to (first inst of subroutine).
PVS_FC_LAST_INST_0	23:16	0x0	This field has multiple definitions as follows: JUMP - Not Applicable LOOP - The last instruction of the loop. JSR - The last instruction of the subroutine.
PVS_FC_RTN_INST_0	31:24	0x0	This field has multiple definitions as follows: JUMP - Not Applicable LOOP - First Instruction of Loop (Typically ACT_ADRS + 1) JSR - First Instruction After JSR (Typically ACT_ADRS + 1)

VAP:VAP_PVS_FLOW_CNTL_LOOP_INDEX_[0-15] · [R/W] · 32 bits · Access: 32 · MMReg:0x2290-0x22cc			
DESCRIPTION: *Programmable Vertex Shader Flow Control Loop Index Register 0*			
Field Name	Bits	Default	Description
PVS_FC_LOOP_INIT_VAL_0	7:0	0x0	This field stores the automatic loop index register init value. This is an 8-bit unsigned value 0-255. This field is only used if the corresponding control flow instruction is a loop.
PVS_FC_LOOP_STEP_VAL_0	15:8	0x0	This field stores the automatic loop index register step value. This is an 8-bit 2`s comp signed value -128-127. This field is only used if the corresponding control flow instruction is a loop.

VAP:VAP_PVS_FLOW_CNTL_OPC · [R/W] · 32 bits · Access: 32 · MMReg:0x22dc			
DESCRIPTION: *Programmable Vertex Shader Flow Control Opcode Register*			
Field Name	Bits	Default	Description
PVS_FC_OPC_0	1:0	0x0	This opcode field determines what type of control flow instruction to execute. 0 = NO_OP 1 = JUMP 2 = LOOP 3 = JSR (Jump to Subroutine)
PVS_FC_OPC_1	3:2	0x0	See PVS_FC_OPC_0.
PVS_FC_OPC_2	5:4	0x0	See PVS_FC_OPC_0.
PVS_FC_OPC_3	7:6	0x0	See PVS_FC_OPC_0.
PVS_FC_OPC_4	9:8	0x0	See PVS_FC_OPC_0.
PVS_FC_OPC_5	11:10	0x0	See PVS_FC_OPC_0.
PVS_FC_OPC_6	13:12	0x0	See PVS_FC_OPC_0.
PVS_FC_OPC_7	15:14	0x0	See PVS_FC_OPC_0.
PVS_FC_OPC_8	17:16	0x0	See PVS_FC_OPC_0.
PVS_FC_OPC_9	19:18	0x0	See PVS_FC_OPC_0.

PVS_FC_OPC_10	21:20	0x0	See PVS_FC_OPC_0.
PVS_FC_OPC_11	23:22	0x0	See PVS_FC_OPC_0.
PVS_FC_OPC_12	25:24	0x0	See PVS_FC_OPC_0.
PVS_FC_OPC_13	27:26	0x0	See PVS_FC_OPC_0.
PVS_FC_OPC_14	29:28	0x0	See PVS_FC_OPC_0.
PVS_FC_OPC_15	31:30	0x0	See PVS_FC_OPC_0.

VAP:VAP_PVS_STATE_FLUSH_REG · [R/W] · 32 bits · Access: 32 · MMReg:0x2284			
Field Name	Bits	Default	Description
DATA_REGISTER (Access: W)	31:0	0x0	This register is used to force a flush of the PVS block when single-buffered updates are performed. The multi-state control of PVS Code and Const memories by the driver is primarily for more flexible PVS state control and for performance testing. When this register address is written, the State Block will force a flush of PVS processing so that both versions of PVS state are available before updates are processed. This register is write only, and the data that is written is unused.

VAP:VAP_PVS_VECTOR_DATA_REG · [R/W] · 32 bits · Access: 32 · MMReg:0x2204			
Field Name	Bits	Default	Description
DATA_REGISTER	31:0	0x0	32-bit data to write to Vector Memory. Used for PVS code and Constant updates.

VAP:VAP_PVS_VECTOR_DATA_REG_128 · [W] · 32 bits · Access: 32 · MMReg:0x2208			
Field Name	Bits	Default	Description
DATA_REGISTER	31:0	0x0	128-bit data path to write to Vector Memory. Used for PVS code and Constant updates.

VAP:VAP_PVS_VECTOR_INDX_REG · [R/W] · 32 bits · Access: 32 · MMReg:0x2200			
Field Name	Bits	Default	Description
OCTWORD_OFFSET	10:0	0x0	Octword offset to begin writing.

VAP:VAP_PVS_VTX_TIMEOUT_REG · [R/W] · 32 bits · Access: 32 · MMReg:0x2288			
Field Name	Bits	Default	Description
CLK_COUNT	31:0	0xFFFFFFFF	This register is used to define the number of core clocks to wait for a vertex to be received by the VAP input controller (while the primitive path is backed up) before forcing any accumulated vertices to be submitted to the vertex processing path.

VAP:VAP_VF_CNTL · [R/W] · 32 bits · Access: 32 · MMReg:0x2084			
DESCRIPTION: *Vertex Fetcher Control*			
Field Name	Bits	Default	Description
PRIM_TYPE	3:0	0x0	Primitive Type 0 : None (will not trigger Setup Engine to run) 1 : Point List 2 : Line List 3 : Line Strip 4 : Triangle List 5 : Triangle Fan 6 : Triangle Strip 7 : Triangle with wFlags (aka, Rage128 `Type-2` triangles) * 8-11 : Unused 12 : Line Loop 13 : Quad List 14 : Quad Strip 15 : Polygon *Encoding 7 indicates whether a 16-bit word of wFlags is present in the stream of indices arriving when the VTX_AMODE is programmed as a `0`. The Setup Engine just steps over the wFlags word; ignoring it. 0 = Stream contains just indices, as: [Index1, Index0] [Index3, Index2] [Index5, Index4] etc... 1 = Stream contains indices and wFlags: [Index1, Index0] [wFlags,Index 2] [Index4, Index3] [wFlags, Index5] etc...
PRIM_WALK	5:4	0x0	Method of Passing Vertex Data. 0 : State-Based Vertex Data. (Vertex data and tokens embedded in command stream.) 1 = Indexes (Indices embedded in command stream; vertex data to be fetched from memory.) 2 = Vertex List (Vertex data to be fetched from memory.) 3 = Vertex Data (Vertex data embedded in command stream.)
RSVD_PREV_USED	10:6	0x0	
INDEX_SIZE	11	0x0	When set, vertex indices are 32-bits/indx, otherwise, 16-bits/indx.
VTX_REUSE_DIS	12	0x0	When set, vertex reuse is disabled. DO NOT SET unless PRIM_WALK is Indexes.
DUAL_INDEX_MODE	13	0x0	When set, the incoming index is treated as two separate

			indices. Bits 23-16 are used as the index for AOS 0 (These are 0 for 16-bit indices) Bits 15-0 are used as the index for AOS 1-15. This mode was added specifically for HOS usage
NUM_VERTICES	31:16	0x0	Number of vertices in the command packet.

VAP:VAP_VF_MAX_VTX_INDX · [R/W] · 32 bits · Access: 32 · MMReg:0x2134			
DESCRIPTION: *Maximum Vertex Indx Clamp*			
Field Name	Bits	Default	Description
MAX_INDX	23:0	0xFFFFFF	If index to be fetched is larger than this value, the fetch indx is set to MAX_INDX

VAP:VAP_VF_MIN_VTX_INDX · [R/W] · 32 bits · Access: 32 · MMReg:0x2138			
DESCRIPTION: *Minimum Vertex Indx Clamp*			
Field Name	Bits	Default	Description
MIN_INDX	23:0	0x0	If index to be fetched is smaller than this value, the fetch indx is set to MIN_INDX

VAP:VAP_VPORT_XOFFSET · [R/W] · 32 bits · Access: 32 · MMReg:0x1d9c, MMReg:0x209c			
DESCRIPTION: *Viewport Transform X Offset*			
Field Name	Bits	Default	Description
VPORT_XOFFSET	31:0	0x0	Viewport Offset for X coordinates. An IEEE float.

VAP:VAP_VPORT_XSCALE · [R/W] · 32 bits · Access: 32 · MMReg:0x1d98, MMReg:0x2098			
DESCRIPTION: *Viewport Transform X Scale Factor*			
Field Name	Bits	Default	Description
VPORT_XSCALE	31:0	0x0	Viewport Scale Factor for X coordinates. An IEEE float.

VAP:VAP_VPORT_YOFFSET · [R/W] · 32 bits · Access: 32 · MMReg:0x1da4, MMReg:0x20a4			
DESCRIPTION: *Viewport Transform Y Offset*			
Field Name	Bits	Default	Description
VPORT_YOFFSET	31:0	0x0	Viewport Offset for Y coordinates. An IEEE float.

VAP:VAP_VPORT_YSCALE · [R/W] · 32 bits · Access: 32 · MMReg:0x1da0, MMReg:0x20a0			
DESCRIPTION: *Viewport Transform Y Scale Factor*			
Field Name	Bits	Default	Description
VPORT_YSCALE	31:0	0x0	Viewport Scale Factor for Y coordinates. An IEEE float.

VAP:VAP_VPORT_ZOFFSET · [R/W] · 32 bits · Access: 32 · MMReg:0x1dac, MMReg:0x20ac			
DESCRIPTION: *Viewport Transform Z Offset*			
Field Name	Bits	Default	Description
VPORT_ZOFFSET	31:0	0x0	Viewport Offset for Z coordinates. An IEEE float.

VAP:VAP_VPORT_ZSCALE · [R/W] · 32 bits · Access: 32 · MMReg:0x1da8, MMReg:0x20a8			
DESCRIPTION: *Viewport Transform Z Scale Factor*			
Field Name	Bits	Default	Description
VPORT_ZSCALE	31:0	0x0	Viewport Scale Factor for Z coordinates. An IEEE float.

VAP:VAP_VTE_CNTL · [R/W] · 32 bits · Access: 32 · MMReg:0x20b0			
DESCRIPTION: *Viewport Transform Engine Control*			
Field Name	Bits	Default	Description
VPORT_X_SCALE_ENA	0	0x0	Viewport Transform Scale Enable for X component
VPORT_X_OFFSET_ENA	1	0x0	Viewport Transform Offset Enable for X component
VPORT_Y_SCALE_ENA	2	0x0	Viewport Transform Scale Enable for Y component
VPORT_Y_OFFSET_ENA	3	0x0	Viewport Transform Offset Enable for Y component
VPORT_Z_SCALE_ENA	4	0x0	Viewport Transform Scale Enable for Z component
VPORT_Z_OFFSET_ENA	5	0x0	Viewport Transform Offset Enable for Z component
VTX_XY_FMT	8	0x0	Indicates that the incoming X, Y have already been multiplied by 1/W0. If OFF, the Setup Engine will bultiply the X, Y coordinates by 1/W0.,
VTX_Z_FMT	9	0x0	Indicates that the incoming Z has already been multiplied by 1/W0. If OFF, the Setup Engine will multiply the Z coordinate by 1/W0.
VTX_W0_FMT	10	0x0	Indicates that the incoming W0 is not 1/W0. If ON, the Setup Engine will perform the reciprocal to get 1/W0.
SERIAL_PROC_ENA	11	0x0	If set, x,y,z viewport transform are performed serially through a single pipeline instead of in parallel. Used to mimic RL300 design.

VAP:VAP_VTX_AOS_ADDR[0-15] · [R/W] · 32 bits · Access: 32 · MMReg:0x20c8-0x2120			
DESCRIPTION: *Array-of-Structures Address 0*			
Field Name	Bits	Default	Description
VTX_AOS_ADDR0	31:2	0x0	Base Address of the Array of Structures.

VAP:VAP_VTX_AOS_ATTR[01-1415] · [R/W] · 32 bits · Access: 32 · MMReg:0x20c4-0x2118			
DESCRIPTION: *Array-of-Structures Attributes 0 & 1*			
Field Name	Bits	Default	Description
VTX_AOS_COUNT0	6:0	0x0	Number of dwords in this structure.
VTX_AOS_STRIDE0	14:8	0x0	Number of dwords from one array element to the next.
VTX_AOS_COUNT1	22:16	0x0	Number of dwords in this structure.
VTX_AOS_STRIDE1	30:24	0x0	Number of dwords from one array element to the next.

VAP:VAP_VTX_NUM_ARRAYS · [R/W] · 32 bits · Access: 32 · MMReg:0x20c0			
DESCRIPTION: *Vertex Array of Structures Control*			
Field Name	Bits	Default	Description
VTX_NUM_ARRAYS	4:0	0x0	The number of arrays required to represent the current vertex type. Each Array is described by the following three fields: VTX_AOS_ADDR, VTX_AOS_COUNT, VTX_AOS_STRIDE.
VC_FORCE_PREFETCH	5	0x0	Force Vertex Data Pre-fetching. If this bit is set, then a 256-bit word will always be fetched, regardless of which dwords are needed. Typically useful when VAP_VF_CNTL.PRIM_WALK is set to Vertex List (Auto-incremented indices).
VC_DIS_CACHE_INVLD	6	0x0	
AOS_0_FETCH_SIZE	16	0x0	Granule Size to Fetch for AOS 0. 0 = 128-bit granule size 1 = 256-bit granule size This allows the driver to program the fetch size based on DWORDS/VTX/AOS combined with AGP vs. LOC Memory. The general belief is that the granule size should always be 256-bits for LOC memory and AGP8X data, but should be 128-bit for AGP2X/4X data if the DWORDS/VTX/AOS is less than TBD (128?) bits.
AOS_1_FETCH_SIZE	17	0x0	See AOS_0_FETCH_SIZE
AOS_2_FETCH_SIZE	18	0x0	See AOS_0_FETCH_SIZE
AOS_3_FETCH_SIZE	19	0x0	See AOS_0_FETCH_SIZE
AOS_4_FETCH_SIZE	20	0x0	See AOS_0_FETCH_SIZE
AOS_5_FETCH_SIZE	21	0x0	See AOS_0_FETCH_SIZE
AOS_6_FETCH_SIZE	22	0x0	See AOS_0_FETCH_SIZE
AOS_7_FETCH_SIZE	23	0x0	See AOS_0_FETCH_SIZE
AOS_8_FETCH_SIZE	24	0x0	See AOS_0_FETCH_SIZE
AOS_9_FETCH_SIZE	25	0x0	See AOS_0_FETCH_SIZE
AOS_10_FETCH_SIZE	26	0x0	See AOS_0_FETCH_SIZE
AOS_11_FETCH_SIZE	27	0x0	See AOS_0_FETCH_SIZE
AOS_12_FETCH_SIZE	28	0x0	See AOS_0_FETCH_SIZE

AOS_13_FETCH_SIZE	29	0x0	See AOS_0_FETCH_SIZE
AOS_14_FETCH_SIZE	30	0x0	See AOS_0_FETCH_SIZE
AOS_15_FETCH_SIZE	31	0x0	See AOS_0_FETCH_SIZE

VAP:VAP_VTX_SIZE · [R/W] · 32 bits · Access: 32 · MMReg:0x20b4

DESCRIPTION: *Vertex Size Specification Register*

Field Name	Bits	Default	Description
DWORDS_PER_VTX	6:0	0x0	This field specifies the number of DWORDS per vertex to expect when VAP_VF_CNTL.PRIM_WALK is set to Vertex Data (vertex data embedded in command stream). This field is not used for any other PRIM_WALK settings. This field replaces the usage of the VAP_VTX_FMT_0/1 for this purpose in prior implementations.

VAP:VAP_VTX_STATE_CNTL · [R/W] · 32 bits · Access: 32 · MMReg:0x2180

DESCRIPTION: *VAP Vertex State Control Register*

Field Name	Bits	Default	Description
COLOR_0_ASSEMBLY_CNTL	1:0	0x0	0 : Select Color 0 1 : Select User Color 0 2 : Select User Color 1 3 : Reserved
COLOR_1_ASSEMBLY_CNTL	3:2	0x0	0 : Select Color 1 1 : Select User Color 0 2 : Select User Color 1 3 : Reserved
COLOR_2_ASSEMBLY_CNTL	5:4	0x0	0 : Select Color 2 1 : Select User Color 0 2 : Select User Color 1 3 : Reserved
COLOR_3_ASSEMBLY_CNTL	7:6	0x0	0 : Select Color 3 1 : Select User Color 0 2 : Select User Color 1 3 : Reserved
COLOR_4_ASSEMBLY_CNTL	9:8	0x0	0 : Select Color 4 1 : Select User Color 0 2 : Select User Color 1 3 : Reserved
COLOR_5_ASSEMBLY_CNTL	11:10	0x0	0 : Select Color 5 1 : Select User Color 0 2 : Select User Color 1 3 : Reserved
COLOR_6_ASSEMBLY_CNTL	13:12	0x0	0 : Select Color 6 1 : Select User Color 0 2 : Select User Color 1

			3 : Reserved
COLOR_7_ASSEMBLY_CNTL	15:14	0x0	0 : Select Color 7 1 : Select User Color 0 2 : Select User Color 1 3 : Reserved
UPDATE_USER_COLOR_0_ENA	16	0x0	0 : User Color 0 State is NOT updated when User Color 0 is written. 1 : User Color 1 State IS updated when User Color 0 is written.
USE_ADDR_IND_TBL	18	0x0	0 : Use vertex state addresses directly to write to vertex state memory. 1 : Use Address Indirection table to write to vertex state memory for lower 64 DWORD addresses.

VAP:VAP_VTX_ST_BLND_WT_[0-3] · [R/W] · 32 bits · Access: 32 · MMReg:0x2430-0x243c			
Field Name	Bits	Default	Description
DATA_REGISTER	31:0	0x0	

VAP:VAP_VTX_ST_CLR_[0-7]_A · [R/W] · 32 bits · Access: 32 · MMReg:0x232c-0x239c			
Field Name	Bits	Default	Description
DATA_REGISTER	31:0	0x0	

VAP:VAP_VTX_ST_CLR_[0-7]_B · [R/W] · 32 bits · Access: 32 · MMReg:0x2328-0x2398			
Field Name	Bits	Default	Description
DATA_REGISTER	31:0	0x0	

VAP:VAP_VTX_ST_CLR_[0-7]_G · [R/W] · 32 bits · Access: 32 · MMReg:0x2324-0x2394			
Field Name	Bits	Default	Description
DATA_REGISTER	31:0	0x0	

VAP:VAP_VTX_ST_CLR_[0-7]_PKD · [W] · 32 bits · Access: 32 · MMReg:0x2470-0x248c			
Field Name	Bits	Default	Description
DATA_REGISTER	31:0	0x0	

VAP:VAP_VTX_ST_CLR_[0-7_R · [R/W] · 32 bits · Access: 32 · MMReg:0x2320-0x2390			
Field Name	Bits	Default	Description
DATA_REGISTER	31:0	0x0	

VAP:VAP_VTX_ST_DISC_FOG · [R/W] · 32 bits · Access: 32 · MMReg:0x2424			
Field Name	Bits	Default	Description
DATA_REGISTER	31:0	0x0	

VAP:VAP_VTX_ST_EDGE_FLAGS · [R/W] · 32 bits · Access: 32 · MMReg:0x245c			
Field Name	Bits	Default	Description
DATA_REGISTER	0	0x0	

VAP:VAP_VTX_ST_END_OF_PKT · [W] · 32 bits · Access: 32 · MMReg:0x24ac			
Field Name	Bits	Default	Description
DATA_REGISTER	31:0	0x0	

VAP:VAP_VTX_ST_NORM_0_PKD · [W] · 32 bits · Access: 32 · MMReg:0x2498			
Field Name	Bits	Default	Description
DATA_REGISTER	31:0	0x0	

VAP:VAP_VTX_ST_NORM_0_X · [R/W] · 32 bits · Access: 32 · MMReg:0x2310			
Field Name	Bits	Default	Description
DATA_REGISTER	31:0	0x0	

VAP:VAP_VTX_ST_NORM_0_Y · [R/W] · 32 bits · Access: 32 · MMReg:0x2314			
Field Name	Bits	Default	Description
DATA_REGISTER	31:0	0x0	

VAP:VAP_VTX_ST_NORM_0_Z · [R/W] · 32 bits · Access: 32 · MMReg:0x2318			
Field Name	Bits	Default	Description
DATA_REGISTER	31:0	0x0	

VAP:VAP_VTX_ST_NORM_1_X · [R/W] · 32 bits · Access: 32 · MMReg:0x2450			
Field Name	Bits	Default	Description
DATA_REGISTER	31:0	0x0	

VAP:VAP_VTX_ST_NORM_1_Y · [R/W] · 32 bits · Access: 32 · MMReg:0x2454			
Field Name	Bits	Default	Description

DATA_REGISTER	31:0	0x0	

VAP:VAP_VTX_ST_NORM_1_Z · [R/W] · 32 bits · Access: 32 · MMReg:0x2458			
Field Name	Bits	Default	Description
DATA_REGISTER	31:0	0x0	

VAP:VAP_VTX_ST_PNT_SPRT_SZ · [R/W] · 32 bits · Access: 32 · MMReg:0x2420			
Field Name	Bits	Default	Description
DATA_REGISTER	31:0	0x0	

VAP:VAP_VTX_ST_POS_0_W_4 · [R/W] · 32 bits · Access: 32 · MMReg:0x230c			
Field Name	Bits	Default	Description
DATA_REGISTER	31:0	0x0	

VAP:VAP_VTX_ST_POS_0_X_2 · [W] · 32 bits · Access: 32 · MMReg:0x2490			
Field Name	Bits	Default	Description
DATA_REGISTER	31:0	0x0	

VAP:VAP_VTX_ST_POS_0_X_3 · [W] · 32 bits · Access: 32 · MMReg:0x24a0			
Field Name	Bits	Default	Description
DATA_REGISTER	31:0	0x0	

VAP:VAP_VTX_ST_POS_0_X_4 · [R/W] · 32 bits · Access: 32 · MMReg:0x2300			
Field Name	Bits	Default	Description
DATA_REGISTER	31:0	0x0	

VAP:VAP_VTX_ST_POS_0_Y_2 · [W] · 32 bits · Access: 32 · MMReg:0x2494			
Field Name	Bits	Default	Description
DATA_REGISTER	31:0	0x0	

VAP:VAP_VTX_ST_POS_0_Y_3 · [W] · 32 bits · Access: 32 · MMReg:0x24a4			
Field Name	Bits	Default	Description
DATA_REGISTER	31:0	0x0	

VAP:VAP_VTX_ST_POS_0_Y_4 · [R/W] · 32 bits · Access: 32 · MMReg:0x2304			
Field Name	Bits	Default	Description
DATA_REGISTER	31:0	0x0	

VAP:VAP_VTX_ST_POS_0_Z_3 · [W] · 32 bits · Access: 32 · MMReg:0x24a8			
Field Name	Bits	Default	Description
DATA_REGISTER	31:0	0x0	

VAP:VAP_VTX_ST_POS_0_Z_4 · [R/W] · 32 bits · Access: 32 · MMReg:0x2308			
Field Name	Bits	Default	Description
DATA_REGISTER	31:0	0x0	

VAP:VAP_VTX_ST_POS_1_W · [R/W] · 32 bits · Access: 32 · MMReg:0x244c			
Field Name	Bits	Default	Description
DATA_REGISTER	31:0	0x0	

VAP:VAP_VTX_ST_POS_1_X · [R/W] · 32 bits · Access: 32 · MMReg:0x2440			
Field Name	Bits	Default	Description
DATA_REGISTER	31:0	0x0	

VAP:VAP_VTX_ST_POS_1_Y · [R/W] · 32 bits · Access: 32 · MMReg:0x2444			
Field Name	Bits	Default	Description
DATA_REGISTER	31:0	0x0	

VAP:VAP_VTX_ST_POS_1_Z · [R/W] · 32 bits · Access: 32 · MMReg:0x2448			
Field Name	Bits	Default	Description
DATA_REGISTER	31:0	0x0	

VAP:VAP_VTX_ST_PVMS · [R/W] · 32 bits · Access: 32 · MMReg:0x231c			
Field Name	Bits	Default	Description
DATA_REGISTER	31:0	0x0	

VAP:VAP_VTX_ST_SHININESS_0 · [R/W] · 32 bits · Access: 32 · MMReg:0x2428			
Field Name	Bits	Default	Description

DATA_REGISTER	31:0	0x0	

VAP:VAP_VTX_ST_SHININESS_1 · [R/W] · 32 bits · Access: 32 · MMReg:0x242c			
Field Name	Bits	Default	Description
DATA_REGISTER	31:0	0x0	

VAP:VAP_VTX_ST_TEX_[0-7]_Q · [R/W] · 32 bits · Access: 32 · MMReg:0x23ac-0x241c			
Field Name	Bits	Default	Description
DATA_REGISTER	31:0	0x0	

VAP:VAP_VTX_ST_TEX_[0-7]_R · [R/W] · 32 bits · Access: 32 · MMReg:0x23a8-0x2418			
Field Name	Bits	Default	Description
DATA_REGISTER	31:0	0x0	

VAP:VAP_VTX_ST_TEX_[0-7]_S · [R/W] · 32 bits · Access: 32 · MMReg:0x23a0-0x2410			
Field Name	Bits	Default	Description
DATA_REGISTER	31:0	0x0	

VAP:VAP_VTX_ST_TEX_[0-7]_T · [R/W] · 32 bits · Access: 32 · MMReg:0x23a4-0x2414			
Field Name	Bits	Default	Description
DATA_REGISTER	31:0	0x0	

VAP:VAP_VTX_ST_USR_CLR_A · [R/W] · 32 bits · Access: 32 · MMReg:0x246c			
Field Name	Bits	Default	Description
DATA_REGISTER	31:0	0x0	

VAP:VAP_VTX_ST_USR_CLR_B · [R/W] · 32 bits · Access: 32 · MMReg:0x2468			
Field Name	Bits	Default	Description
DATA_REGISTER	31:0	0x0	

VAP:VAP_VTX_ST_USR_CLR_G · [R/W] · 32 bits · Access: 32 · MMReg:0x2464			
Field Name	Bits	Default	Description
DATA_REGISTER	31:0	0x0	

VAP:VAP_VTX_ST_USR_CLR_PKD · [W] · 32 bits · Access: 32 · MMReg:0x249c			
Field Name	Bits	Default	Description
DATA_REGISTER	31:0	0x0	

VAP:VAP_VTX_ST_USR_CLR_R · [R/W] · 32 bits · Access: 32 · MMReg:0x2460			
Field Name	Bits	Default	Description
DATA_REGISTER	31:0	0x0	

1.11 Z Buffer Registers

ZB:ZB_BW_CNTL · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4f1c			
DESCRIPTION: *Z Buffer Band-Width Control*			
Field Name	Bits	Default	Description
HIZ_ENABLE	0	0x0	Enables hierarchical Z. POSSIBLE VALUES: 00 - Hierarchical Z Disabled 01 - Hierarchical Z Enabled
HIZ_MIN	1	0x0	POSSIBLE VALUES: 00 - Update Hierarchical Z with Max value 01 - Update Hierarchical Z with Min value
FAST_FILL	2	0x0	POSSIBLE VALUES: 00 - Fast Fill Disabled 01 - Fast Fill Enabled (ZB_DEPTHCLEARVALUE)
RD_COMP_ENABLE	3	0x0	Enables reading of compressed Z data from memory to the cache. POSSIBLE VALUES: 00 - Z Read Compression Disabled 01 - Z Read Compression Enabled
WR_COMP_ENABLE	4	0x0	Enables writing of compressed Z data from cache to memory, POSSIBLE VALUES: 00 - Z Write Compression Disabled 01 - Z Write Compression Enabled
ZB_CB_CLEAR	5	0x0	This bit is set when the Z buffer is used to help the CB in clearing a region. Part of the region is cleared by the color buffer and part will be cleared by the Z buffer. Since the Z buffer does not have any write masks in the cache, full micro-tiles need to be written. If a partial micro-tile is touched , then the un-touched part will be unknowns. The cache will operate in write-allocate mode and quads will be accumulated in the cache and then evicted to main memory. The color value is supplied through the ZB_DEPTHCLEARVALUE register. POSSIBLE VALUES: 00 - Z unit cache controller does RMW 01 - Z unit cache controller does cache-line granular Write only
FORCE_COMPRESSED_STENCIL_VALUE	6	0x0	Enabling this bit will force all the compressed stencil values to be equal to old_stencil_value&~ZB_STENCILREFMASK.stencilwritemask | ZB_STENCILREFMASK.stencilref&ZB_STENCILREFMASK.stencilwritemask. This should be enabled during stencil clears to avoid needless decompression. POSSIBLE VALUES: 00 - Do not force the compressed stencil value.

			01 - Force the compressed stencil value.

ZB:ZB_CNTL · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4f00			
DESCRIPTION: *Z Buffer Control*			
Field Name	Bits	Default	Description
STENCIL_ENABLE	0	0x0	Enables stenciling. POSSIBLE VALUES: 00 - Disabled 01 - Enabled
Z_ENABLE	1	0x0	Enables Z functions. POSSIBLE VALUES: 00 - Disabled 01 - Enabled
ZWRITEENABLE	2	0x0	Enables writing of the Z buffer. POSSIBLE VALUES: 00 - Disable 01 - Enable
ZSIGNED_COMPARE	3	0x0	Enable signed Z buffer comparison , for W-buffering. POSSIBLE VALUES: 00 - Disable 01 - Enable
STENCIL_FRONT_BACK	4	0x0	When STENCIL_ENABLE is set, setting STENCIL_FRONT_BACK bit to one specifies that stencilfunc/stencilfail/stencilzpass/stencilzfail registers are used if the quad is generated from front faced primitive and stencilfunc_bf/stencilfail_bf/stencilzpass_bf/stencilzfail_bf are used if the quad is generated from a back faced primitive. If the STENCIL_FRONT_BACK is not set, then stencilfunc/stencilfail/stencilzpass/stencilzfail registers determine the operation independent of the front/back face state of the quad. POSSIBLE VALUES: 00 - Disable 01 - Enable

ZB:ZB_DEPTHCLEARVALUE · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4f28			
DESCRIPTION: *Z Buffer Clear Value*			
Field Name	Bits	Default	Description
DEPTHCLEARVALUE	31:0	0x0	When a block has a Z Mask value of 0, all Z values in

			that block are cleared to this value. In 24bpp, the stencil value is also updated regardless of whether it is enabled or not.

ZB:ZB_DEPTHOFFSET · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4f20			
DESCRIPTION: *Z Buffer Address Offset*			
Field Name	Bits	Default	Description
DEPTHOFFSET	31:5	0x0	2K aligned Z buffer address offset for macro tiles.

ZB:ZB_DEPTHPITCH · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4f24			
DESCRIPTION: *Z Buffer Pitch and Endian Control*			
Field Name	Bits	Default	Description
DEPTHPITCH	13:2	0x0	Z buffer pitch in multiples of 4 pixels.
DEPTHMACROTILE	16	0x0	Specifies whether Z buffer is macro-tiled. macro-tiles are 2K aligned POSSIBLE VALUES: 00 - macro tiling disabled 01 - macro tiling enabled
DEPTHMICROTILE	18:17	0x0	Specifies whether Z buffer is micro-tiled. micro-tiles is 32 bytes POSSIBLE VALUES: 00 - 32 byte cache line is linear 01 - 32 byte cache line is tiled 02 - 32 byte cache line is tiled square (only applies to 16-bit pixels) 03 - Reserved
DEPTHENDIAN	20:19	0x0	Specifies endian control for the Z buffer. POSSIBLE VALUES: 00 - No swap 01 - Word swap 02 - Dword swap 03 - Half Dword swap

ZB:ZB_DEPTHXY_OFFSET · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4f60			
DESCRIPTION: *Depth buffer X and Y coordinate offset*			
Field Name	Bits	Default	Description
DEPTHX_OFFSET	11:1	0x0	X coordinate offset. multiple of 32 . Bits 4:0 have to be zero
DEPTHY_OFFSET	27:17	0x0	Y coordinate offset. multiple of 32 . Bits 4:0 have to be zero

ZB:ZB_FORMAT · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4f10			
DESCRIPTION: *Format of the Data in the Z buffer*			
Field Name	Bits	Default	Description
DEPTHFORMAT	3:0	0x0	Specifies the format of the Z buffer. POSSIBLE VALUES: 00 - 16-bit Integer Z 01 - 16-bit compressed 13E3 02 - 24-bit Integer Z, 8 bit Stencil (LSBs) 03 - RESERVED 04 - RESERVED 05 - RESERVED 06 - RESERVED 07 - RESERVED 08 - RESERVED 09 - RESERVED 10 - RESERVED 11 - RESERVED 12 - RESERVED 13 - RESERVED 14 - RESERVED 15 - RESERVED
INVERT	4	0x0	POSSIBLE VALUES: 00 - in 13E3 format , count leading 0`s 01 - in 13E3 format , count leading 1`s.
PEQ8	5	0x0	This bit is unused POSSIBLE VALUES: 00 - 7 bytes per plane equation, 1 byte for stencil 01 - 8 bytes per plane equation, no bytes for stencil

ZB:ZB_HIZ_DWORD · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4f4c			
DESCRIPTION: *Hierarchical Z Data*			
Field Name	Bits	Default	Description
HIZ_DWORD	31:0	0x0	This DWORD contains an 8-bit value for 4 4x4 blocks. The 4 blocks are organized as a 2x2 tile. The frame buffer coordinate (X,Y) corresponds to a particular 8-bit value for the 4x4 block within the DWORD as follows: BITPOS[4:0] = 16 * X[2] + 8 * Y[2] HIZ[7:0] = HIZDWORD[BITPOS+7:BITPOS]

ZB:ZB_HIZ_OFFSET · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4f44
DESCRIPTION: *Hierarchical Z Memory Offset*

Field Name	Bits	Default	Description
HIZ_OFFSET	16:2	0x0	DWORD offset into HiZ RAM. A DWORD can hold an 8-bit HiZ value for 4 blocks, so this offset is aligned on 4 4x4 blocks. In each pipe, the HIZ RAM DWORD address is generated from a pixel x[11:0] , y[11:0] as follows: HIZ_DWORD_ADDRESS[13:0] = HIZ_OFFSET[16:3] + Y[11:3] * HIZ_PITCH[13:5] + X[11:5].

ZB:ZB_HIZ_PITCH · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4f54			
DESCRIPTION: *Hierarchical Z Pitch*			
Field Name	Bits	Default	Description
HIZ_PITCH	13:4	0x0	Pitch used in HiZ address computation.

ZB:ZB_HIZ_RDINDEX · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4f50			
DESCRIPTION: *Hierarchical Z Read Index*			
Field Name	Bits	Default	Description
HIZ_RDINDEX	16:2	0x0	Read index into HiZ RAM. The index must start on a DWORD boundary. RDINDEX words much like WRINDEX. Every read from HIZ_DWORD will increment the register by 2.

ZB:ZB_HIZ_WRINDEX · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4f48			
DESCRIPTION: *Hierarchical Z Write Index*			
Field Name	Bits	Default	Description
HIZ_WRINDEX	16:2	0x0	Self-incrementing write index into the HiZ RAM. Starting write index must start on a DWORD boundary. Each time ZB_HIZ_DWORD is written, this index will increment by two DWORD, this due to the fact that there are 2 pipes and the data is broadcasted to both pipes. HIZ_OFFSET and HIZ_PITCH are not used to compute read/write address to HIZ ram, when it is accessed through WRINDEX and DWORD

ZB:ZB_STENCILREFMASK · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4f08			
DESCRIPTION: *Stencil Reference Value and Mask*			
Field Name	Bits	Default	Description
STENCILREF	7:0	0x0	Specifies the reference stencil value.
STENCILMASK	15:8	0x0	This value is ANDed with both the reference and the current stencil value prior to the stencil test.
STENCILWRITEMASK	23:16	0x0	Specifies the write mask for the stencil planes.

ZB:ZB_ZCACHE_CTLSTAT · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4f18			
DESCRIPTION: *Z Buffer Cache Control/Status*			
Field Name	Bits	Default	Description
ZC_FLUSH	0	0x0	Setting this bit flushes the dirty data from the Z cache. Unless ZC_FREE bit is also set, the tags in the cache remain valid. A purge is achieved by setting both ZC_FLUSH and ZC_FREE. This is a sticky bit and it clears itself at the end of the operation. POSSIBLE VALUES: 00 - No effect 01 - Flush and Free Z cache lines
ZC_FREE	1	0x0	Setting this bit invalidates the Z cache tags. Unless ZC_FLUSH bit is also set, the cachelines are not written to memory. A purge is achieved by setting both ZC_FLUSH and ZC_FREE. This is a sticky bit that clears itself at the end of the operation. POSSIBLE VALUES: 00 - No effect 01 - Free Z cache lines (invalidate)
ZC_BUSY	31	0x0	This bit is unused ... POSSIBLE VALUES: 00 - Idle 01 - Busy

ZB:ZB_ZPASS_ADDR · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4f5c			
DESCRIPTION: *Z Buffer Z Pass Counter Address*			
Field Name	Bits	Default	Description
ZPASS_ADDR	31:2	0x0	Writing this location with a DWORD address causes the value in ZB_ZPASS_DATA to be written to main memory at the location pointed to by this address. NOTE: R300 has 2 pixel pipes. Broadcasting this address causes both pipes to write their ZPASS value to the same address. There is no guarantee which pipe will write last. So when writing to this register, the GA needs to be programmed to send the write command to pipe 0. Then a different address needs to be written to pipe 1. Then both pipes should be enabled for further register writes.

ZB:ZB_ZPASS_DATA · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4f58			
DESCRIPTION: *Z Buffer Z Pass Counter Data*			
Field Name	Bits	Default	Description

ZPASS_DATA	31:0	0x0	Contains the number of passed Z components since the last write to this location. Writing this location resets the count to the value written.

ZB:ZB_ZSTENCILCNTL · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4f04			
DESCRIPTION: *Z and Stencil Function Control*			
Field Name	Bits	Default	Description
ZFUNC	2:0	0x0	Specifies the Z function. POSSIBLE VALUES: 00 - Never 01 - Less 02 - Less or Equal 03 - Equal 04 - Greater or Equal 05 - Greater Than 06 - Not Equal 07 - Always
STENCILFUNC	5:3	0x0	Specifies the stencil function. POSSIBLE VALUES: 00 - Never 01 - Less 02 - Less or Equal 03 - Equal 04 - Greater or Equal 05 - Greater 06 - Not Equal 07 - Always
STENCILFAIL	8:6	0x0	Specifies the stencil value to be written if the stencil test fails. POSSIBLE VALUES: 00 - Keep: New value = Old value 01 - Zero: New value = 0 02 - Replace: New value = STENCILREF 03 - Increment: New value++ (clamp) 04 - Decrement: New value-- (clamp) 05 - Invert new value: New value = !Old value 06 - Increment: New value++ (wrap) 07 - Decrement: New value-- (wrap)
STENCILZPASS	11:9	0x0	Same encoding as STENCILFAIL. Specifies the stencil value to be written if the stencil test passes and the Z test passes (or is not enabled).
STENCILZFAIL	14:12	0x0	Same encoding as STENCILFAIL. Specifies the stencil value to be written if the stencil test passes and the Z test fails.
STENCILFUNC_BF	17:15	0x0	Same encoding as STENCILFUNC. Specifies the stencil

			function for back faced quads , if STENCIL_FRONT_BACK = 1.
STENCILFAIL_BF	20:18	0x0	Same encoding as STENCILFAIL. Specifies the stencil value to be written if the stencil test fails for back faced quads, if STENCIL_FRONT_BACK = 1
STENCILZPASS_BF	23:21	0x0	Same encoding as STENCILFAIL. Specifies the stencil value to be written if the stencil test passes and the Z test passes (or is not enabled) for back faced quads, if STENCIL_FRONT_BACK = 1
STENCILZFAIL_BF	26:24	0x0	Same encoding as STENCILFAIL. Specifies the stencil value to be written if the stencil test passes and the Z test fails for back faced quads, if STENCIL_FRONT_BACK = 1

ZB:ZB_ZTOP · [R/W] · 32 bits · Access: 8/16/32 · MMReg:0x4f14			
Field Name	Bits	Default	Description
ZTOP	0	0x0	POSSIBLE VALUES: 00 - Z is at the bottom of the pipe, after the fog unit. 01 - Z is at the top of the pipe, after the scan unit.