

# Intel<sup>®</sup> Open Source HD Graphics Programmers' Reference Manual (PRM)

#### Volume 2, Part 3: Command Reference - Registers

For the 2014 Intel Atom<sup>™</sup> Processors, Celeron<sup>™</sup> Processors, and Pentium<sup>™</sup> Processors based on the "BayTrail" Platform (ValleyView graphics)

© April 2014, Intel Corporation



#### **Creative Commons License**

You are free to Share — to copy, distribute, display, and perform the work

#### Under the following conditions:

**Attribution.** You must attribute the work in the manner specified by the author or licensor (but not in any way that suggests that they endorse you or your use of the work).

No Derivative Works. You may not alter, transform, or build upon this work

#### **Notices and Disclaimers**

INFORMATION IN THIS DOCUMENT IS PROVIDED IN CONNECTION WITH INTEL® PRODUCTS. NO LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE, TO ANY INTELLECTUAL PROPERTY RIGHTS IS GRANTED BY THIS DOCUMENT. EXCEPT AS PROVIDED IN INTEL'S TERMS AND CONDITIONS OF SALE FOR SUCH PRODUCTS, INTEL ASSUMES NO LIABILITY WHATSOEVER AND INTEL DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY, RELATING TO SALE AND/OR USE OF INTEL PRODUCTS INCLUDING LIABILITY OR WARRANTIES RELATING TO FITNESS FOR A PARTICULAR PURPOSE, MERCHANTABILITY, OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT.

A "Mission Critical Application" is any application in which failure of the Intel Product could result, directly or indirectly, in personal injury or death. SHOULD YOU PURCHASE OR USE INTEL'S PRODUCTS FOR ANY SUCH MISSION CRITICAL APPLICATION, YOU SHALL INDEMNIFY AND HOLD INTEL AND ITS SUBSIDIARIES, SUBCONTRACTORS AND AFFILIATES, AND THE DIRECTORS, OFFICERS, AND EMPLOYEES OF EACH, HARMLESS AGAINST ALL CLAIMS COSTS, DAMAGES, AND EXPENSES AND REASONABLE ATTORNEYS' FEES ARISING OUT OF, DIRECTLY OR INDIRECTLY, ANY CLAIM OF PRODUCT LIABILITY, PERSONAL INJURY, OR DEATH ARISING IN ANY WAY OUT OF SUCH MISSION CRITICAL APPLICATION, WHETHER OR NOT INTEL OR ITS SUBCONTRACTOR WAS NEGLIGENT IN THE DESIGN, MANUFACTURE, OR WARNING OF THE INTEL PRODUCT OR ANY OF ITS PARTS.

Intel may make changes to specifications and product descriptions at any time, without notice. Designers must not rely on the absence or characteristics of any features or instructions marked "reserved" or "undefined". Intel reserves these for future definition and shall have no responsibility whatsoever for conflicts or incompatibilities arising from future changes to them. The information here is subject to change without notice. Do not finalize a design with this information.

The products described in this document may contain design defects or errors known as errata which may cause the product to deviate from published specifications. Current characterized errata are available on request.

Implementations of the I2C bus/protocol may require licenses from various entities, including Philips Electronics N.V. and North American Philips Corporation.

Intel and the Intel logo are trademarks of Intel Corporation in the U.S. and other countries.

\*Other names and brands may be claimed as the property of others.

**Copyright © 2014, Intel Corporation. All rights reserved.** 



#### **Table of Contents**

BBA_LEVEL2 - 2nd Level Batch Buffer Address	10
ARB_MODE - Arbiter Mode Control Register	11
ARB_MODE - Arbiter Mode Control register	15
3DPRIM_END_OFFSET - Auto Draw End Offset	
BB_ADDR - Batch Buffer Head Pointer Register	
BB_STATE - Batch Buffer State Register	
BCS_CXT_SIZE - BCS Context Sizes	
BCS_EIR - BCS Error Identity Register	
BCS_EMR - BCS Error Mask Register	
BCS_ESR - BCS Error Status Register	
BCS_EXCC - BCS Execute Condition Code Register	
BCS_HWSTAM - BCS Hardware Status Mask Register	
BCS_PWRCTX_MAXCNT - BCS IDLE Max Count	
BCS_INSTPM - BCS Instruction Parser Mode Register	
BCS_IMR - BCS Interrupt Mask Register	
BCS_MI_MODE - BCS Mode Register for Software Interface	
BCS_PP_DCLV - BCS PPGTT Directory Cacheline Valid Register	
BCS_TLBPEND_RDY0 - BCS Ready Bit Vector for TLBPEND Registers	
BCS_TIMESTAMP - BCS Reported Timestamp Count	
BCS_RNCID - BCS Ring Buffer Next Context ID Register	
BCS_TLBPEND_SEC0 - BCS Section 0 of TLBPEND entry	
BCS_TLBPEND_SEC1 - BCS Section 1 of TLBPEND entry	
BCS_PSMI_CTRL - BCS Sleep State and PSMI Control	
BCS_SWCTRL - BCS SW Control	43
BCSTLB_VA - BCS TLB Virtual Page Address Registers	44
BCS_TLBPEND_VLD0 - BCS Valid Bit Vector for TLBPEND registers	45
BCS_SYNC_FLIP_STATUS - BCS Wait for event and Display flip flags Register	46
BCS_CTR_THRSH - BCS Watchdog Counter Threshold	51
MFC_BITSTREAM_SE_BITCOUNT_SLICE - Bitstream Output Bit Count for the last Syntax Element Report Register	52
MFC_BITSTREAM_BYTECOUNT_SLICE - Bitstream Output Byte Count Per Slice Report Register	53
MFC_AVC_MINSIZE_PADDING_COUNT - Bitstream Output Minimal Size Padding Count Report Register	54
BLBTLB_VA - BLBTLB_VA Virtual page Address Registers	55
BLT_ENG_FR - Blitter Engine Fault Register	56
BLT_MODE - Blitter Mode Register	
BRSYNC - Blitter/Render Semaphore Sync Register	
BVSYNC - Blitter/Video Semaphore Sync Register	60
CACHE_MODE_0 - Cache Mode Register 0	61
CACHE_MODE_1 - Cache Mode Register 1	64
CASC_LRA_0 - CASC LRA 0	
CASC_LRA_1 - CASC LRA 1	70
CASC_LRA_2 - CASC LRA 2	72
CASC_LRA_2 - CASC LRA 2 CASC_LRA_3 - CASC LRA 3 CL_INVOCATION_COUNT - Clipper Invocation Counter	73



CL_PRIMITIVES_COUNT - Clipper Primitives Counter	76
CGMSG - Clock Gating Messages	
CZWMRK - Color/Depth Write FIFO Watermarks	
TS_GPGPU_THREADS_DISPATCHED - Count Active Channels Dispatched	79
CSPWRFSM - CS Power Management FSM	
CTXTLB_VA - CTXTLB_VA Virtual page Address Registers	
CCID - Current Context Register	
CEC0-0 - Customizable Event Creation 0-0	
CEC1-0 - Customizable Event Creation 1-0	
CEC2-0 - Customizable Event Creation 2-0	
CEC3-0 - Customizable Event Creation 3-0	
CVS_TLB_LRA_0 - CVS TLB LRA 0	95
CVS_TLB_LRA_1 - CVS TLB LRA 1	
CVS_TLB_LRA_2 - CVS TLB LRA 2	
ZSHR - Depth/Early Depth TLB Partitioning Register	
DS_INVOCATION_COUNT - DS Invocation Counter	
EIR - Error Identity Register	
EMR - Error Mask Register	
ESR - Error Status Register	
TD_PM_MODE_EUCOUNT - EU Mask Programming	
EXCC - Execute Condition Code Register	
GAB_ERR_REPORT - GAB Error Reporting Register	
GAB_HWSP_REG - GAB Hardware Status Page Address Register	
GAB_CTL_REG - GAB unit Control Register	
GAC_ARB_CTL_REG - GAC_GAB Arbitration Counters Register 1	
ARB_GAC_GAM_REQCNTS0 - GAC_GAM Arbitration Counters Register 0	
ARB_GAC_GAM_REQCNTS1 - GAC_GAM Arbitration Counters Register 1	
ARB_R_GAC_GAM0 - GAC_GAM R Arbitration Register 0	
ARB_R_GAC_GAM1 - GAC_GAM R Arbitration Register 1	
ARB_R_GAC_GAM2 - GAC_GAM R Arbitration Register 2	
ARB_R_GAC_GAM3 - GAC_GAM R Arbitration Register 3	
ARB_RO_GAC_GAM0 - GAC_GAM RO Arbitration Register 0	
ARB_RO_GAC_GAM1 - GAC_GAM RO Arbitration Register 1	
ARB_RO_GAC_GAM2 - GAC_GAM RO Arbitration Register 2	
ARB_RO_GAC_GAM3 - GAC_GAM RO Arbitration Register 3	
ARB_WR_GAC_GAM0 - GAC_GAM WR Arbitration Register 0	
ARB_WR_GAC_GAM1 - GAC_GAM WR Arbitration Register 1	
ARB_WR_GAC_GAM2 - GAC_GAM WR Arbitration Register 2	
ARB_WR_GAC_GAM3 - GAC_GAM WR Arbitration Register 3	
GAC_HWSP_REG - GAC Hardware Status Page Address Register	
DONE_REG - GAM Fub Done Lookup Register	
GAM_HWSP_REG - GAM Hardware Status Page Address Register	
GFX_PRIO_CTRL - GFX Arbiter Client Priority Control	
MIDARB_GOTOFIELD_HIT0 - Goto Field in Programmable Arbitration for Hit0	
MIDARB_GOTOFIELD_HIT1 - Goto Field in Programmable Arbitration for Hit1	

#### **Command Reference - Registers**



MIDARB_GOTOFIELD_HIT2 - Goto Field in Programmable Arbitration for Hit2	
MIDARB_GOTOFIELD_HIT3 - Goto Field in Programmable Arbitration for Hit3	136
MIDARB_GOTOFIELD_NP0 - Goto Field in Programmable Arbitration for Hit-NP0	
MIDARB_GOTOFIELD_NP1 - Goto Field in Programmable Arbitration for Hit-NP1	139
MIDARB_GOTOFIELD_NP2 - Goto Field in Programmable Arbitration for Hit-NP2	140
MIDARB_GOTOFIELD_NP3 - Goto Field in Programmable Arbitration for Hit-NP3	141
GPGPU_DISPATCHDIMX - GPGPU Dispatch Dimension X	142
GPGPU_DISPATCHDIMY - GPGPU Dispatch Dimension Y	143
GPGPU_DISPATCHDIMZ - GPGPU Dispatch Dimension Z	
GDRST - Graphics Device Reset Control	145
GFX_ENG_FR - Graphics Engine Fault Register	
FENCE - Graphics Memory Fence Table Register	
GFX_MODE - Graphics Mode Register	152
GS_INVOCATION_COUNT - GS Invocation Counter	
GS_PRIMITIVES_COUNT - GS Primitives Counter	
FLRCTLMSG - GT Function Level Reset Control Message	
GT_MODE - GT Mode Register	
HWSTAM - Hardware Status Mask Register	
HWS_PGA - Hardware Status Page Address Register	
HS_INVOCATION_COUNT - HS Invocation Counter	
IA_VERTICES_COUNT - IA Vertices Count	
INSTPM - Instruction Parser Mode Register	
IMR - Interrupt Mask Register	
L3_LRA_0 - L3 LRA 0	
L3_LRA_1 - L3 LRA 1	
3DPRIM_BASE_VERTEX - Load Indirect Base Vertex	
3DPRIM_INSTANCE_COUNT - Load Indirect Instance Count	
3DPRIM_START_INSTANCE - Load Indirect Start Instance	
3DPRIM_START_VERTEX - Load Indirect Start Vertex	
3DPRIM_VERTEX_COUNT - Load Indirect Vertex Count	
ERROR - Main Graphic Arbiter Error Report	
GFX_ARB_ERROR_RPT - Main Graphic Arbiter Error Report Register	179
GFX_PEND_TLB_0 - Max Outstanding Pending TLB Requests 0	
GFX_PEND_TLB_0 - Max Outstanding Pending TLB Requests 0	
GFX_PEND_TLB_1 - Max Outstanding Pending TLB Requests 1	
GFX_PEND_TLB_1 - Max Outstanding Pending TLB Requests 1	
MEDIA_MAX_REQ_COUNT - MAX Requests Allowed - CASC	
GFX_MAX_REQ_COUNT - MAX Requests Allowed - GAM	
GAC_ERROR - Media Arbiter Error Report Register	
MEDIA_ENG_FR - Media Engine Fault Register	
AVC_CABAC_INSERTION_COUNT - MFC_AVC_CABAC_INSERTION_COUNT	
MFC_VIN_AVD_ERROR_CNTR - MFC_AVC Bitstream Decoding Front-End Parsing Logic Error Counter	
MFC_IMAGE_STATUS_CONTROL - MFC Image Status Control	
MFC_IMAGE_STATUS_MASK - MFC Image Status Mask	
MFC_QUP_CT - MFC QP Status Count	



MFD_ERROR_STATUS - MFD Error Status	
MFD_PICTURE_PARAM - MFD Picture Parameter	
MFX_LAT_CT1 - MFX_Memory_Latency_Count1	
MFX_SE-BIN_CT - MFX Frame BitStream SE/BIN Count	204
MFX_MB_COUNT - MFX Frame Macroblock Count	
MFX_MISS_CT - MFX Frame Motion Comp Miss Count	
MFX_READ_CT - MFX Frame Motion Comp Read Count	
MFX_FRAME_PERFORMANCE_CT - MFX Frame Performance Count	
MFX_ROW-PER-BS_COUNT - MFX Frame Row-Stored/BitStream Read Count	
MFX_LAT_CT2 - MFX Memory Latency Count2	
MFX_LAT_CT3 - MFX Memory Latency Count3	
MFX_LAT_CT4 - MFX Memory Latency Count4	
MFX_STATUS_FLAGS - MFX Pipeline Status Flags	
MFX_SLICE_PERFORM_CT - MFX Slice Performance Count	
MISCCPCTL - Misc. Clocking / Reset Control Registers	
RSTCTL - Misc. Reset Control Register	
GAB_MODE - Mode Register for GAB	
GAC_MODE - Mode Register for GAC	
GAFS_MODE - Mode Register for GAFS	
MTTLB_VA - MT Virtual Page Address Registers	
NOPID - NOP Identification Register	
PAK_ERR - PAK_Stream-Out Report (Errors)	
PAK_WARN - PAK_Stream-Out Report (Warnings)	
PAK_REPORT_STAT - PAK Report Running Status	
PAT_INDEX - PAT Index	
PDTLB_VA - PDTLB_VA Virtual page Address Registers	
UHPTR - Pending Head Pointer Register	
PP_DCLV - PPGTT Directory Cacheline Valid Register	230
PP_PFD[0:31] - PPGTT Page Fault Data Registers	231
MI_PREDICATE_RESULT - Predicate Rendering Data Result	
MI_PREDICATE_DATA - Predicate Rendering Data Storage	233
MI_PREDICATE_SRC0 - Predicate Rendering Temporary Register0	234
MI_PREDICATE_SRC1 - Predicate Rendering Temporary Register1	235
IA_PRIMITIVES_COUNT - Primitives Generated By VF	236
MIDARB_PRIO_HIT_REGISTER - Priority Field in Programmable Arbitration for Hit	237
MIDARB_PRIO_NP_REGISTER - Priority Field in Programmable Arbitration for Hit-NP	238
MIDARB_PRIO_MISS_REGISTER - Priority Field in Programmable Arbitration for Miss	240
PRIV_PAT - Private PAT	241
PS_DEPTH_COUNT - PS Depth Count	242
PS_INVOCATION_COUNT - PS Invocation Count	243
RCGCTL1 - RAM Clock Gating Control 1	
RCGCTL2 - RAM Clock Gating Control 2	
RCC_LRA_0 - RCC LRA 0	
RCC_LRA_1 - RCC LRA 1	253
RCCTLB_VA - RCC Virtual page Address Registers	254



RCS_BB_STATE - RCS Batch Buffer State Register	255
RCZTLB_VA - RCZ Virtual Page Address Registers	256
TLBPEND_RDY0 - Ready Bit Vector 0 for TLBPEND registers	257
TLBPEND_RDY1 - Ready Bit Vector 1 for TLBPEND registers	258
RBSYNC - Render/Blitter Semaphore Sync Register	259
MI_MODE - Render Mode Register for Software Interface	260
RVSYNC - Render/Video Semaphore Sync Register	263
PR_CTR - Render Watchdog Counter	264
PR_CTR_THRSH - Render Watchdog Counter Threshold	265
MFC_BITSTREAM_SE_BITCOUNT_FRAME - Reported Bitstream Output Bit Count for Syntax Elements Only Register	266
MFC_BITSTREAM_BYTECOUNT_FRAME - Reported Bitstream Output Byte Count per Frame Register	267
MFC_AVC_CABAC_BIN_COUNT_FRAME - Reported Bitstream Output CABAC Bin Count Register	268
TIMESTAMP - Reported Timestamp Count	
RSTFCTLMSG - Reset Flow Control Messages	270
RING_BUFFER_HEAD_PREEMPT_REG - RING_BUFFER_HEAD_PREEMPT_REG	272
RING_BUFFER_CTL - Ring Buffer Control	273
BCS_RCCID - Ring Buffer Current Context ID Register	276
RING_BUFFER_HEAD - Ring Buffer Head	277
RING_BUFFER_START - Ring Buffer Start	279
RING_BUFFER_TAIL - Ring Buffer Tail	280
TLBPEND_SEC0 - Section 0 of TLBPEND Entry	281
TLBPEND_SEC1 - Section 1 of TLBPEND Entry	282
TLBPEND_SEC2 - Section 2 of TLBPEND Entry	284
SO_NUM_PRIMS_WRITTEN[0:3] - Stream Output Num Primitives Written Counter	285
SO_PRIM_STORAGE_NEEDED[0:3] - Stream Output Primitive Storage Needed Counters	
SO_WRITE_OFFSET[0:3] - Stream Output Write Offsets	287
FF_MODE - Thread Mode Register	288
TLB_RD_ADDR - TLB_RD_ADDRESS Register	291
TLB_RD_DATA - TLB_RD_DATA Register	292
TLB064_VA - TLB064_VA Virtual Page Address Registers	293
TLB132_VA - TLB132_VA Virtual Page Address Registers	294
TLB232_VA - TLB232_VA Virtual Page Address Registers	295
TLB304_VA - TLB304_VA Virtual Page Address Registers	296
GFX_PEND_TLB - TLBPEND Control Register	297
UCGCTL1 - Unit Level Clock Gating Control 1	299
UCGCTL2 - Unit Level Clock Gating Control 2	305
UCGCTL3 - Unit Level Clock Gating Control 3	
UCGCTL4 - Unit Level Clock Gating Control 4	317
CASCTLB_VLD_0 - Valid Bit Vector 0 for CASC	
CVSTLB_VLD_0 - Valid Bit Vector 0 for CVS	324
L3TLB_VLD_0 - Valid Bit Vector 0 for L3	
MTTLB_VLD0 - Valid Bit Vector 0 for MTTLB	326
VICTLB_VLD0 - Valid Bit Vector 0 for MTVICTLB	
RCCTLB_VLD_0 - Valid Bit Vector 0 for RCC	
RCCTLB_VLD0 - Valid Bit Vector 0 for RCCTLB	329



RCZTLB_VLD0 - Valid Bit Vector 0 for RCZTLB	
MTTLB064_VLD0 - Valid Bit Vector 0 for TLB064	
MTTLB132_VLD0 - Valid Bit Vector 0 for TLB132	
MTTLB232_VLD0 - Valid Bit Vector 0 for TLB232	
MTTLB304_VLD0 - Valid Bit Vector 0 for TLB304	
TLBPEND_VLD0 - Valid Bit Vector 0 for TLBPEND registers	
VLFTLB_VLD_0 - Valid Bit Vector 0 for VLF	
ZTLB_VLD_0 - Valid Bit Vector 0 for Z	
CASCTLB_VLD_1 - Valid Bit Vector 1 for CASC	
L3TLB_VLD_1 - Valid Bit Vector 1 for L3	
MTTLB_VLD1 - Valid Bit Vector 1 for MTTLB	
MTVICTLB_VLD1 - Valid Bit Vector 1 for MTVICTLB	
RCCTLB_VLD_1 - Valid Bit Vector 1 for RCC	
RCCTLB_VLD1 - Valid Bit Vector 1 for RCCTLB	
RCZTLB_VLD1 - Valid Bit Vector 1 for RCZTLB	
MTTLB064_VLD1 - Valid Bit Vector 1 for TLB064	
MTTLB132_VLD1 - Valid Bit Vector 1 for TLB132	
MTTLB232_VLD1 - Valid Bit Vector 1 for TLB232	
MTTLB304_VLD1 - Valid Bit Vector 1 for TLB304	
TLBPEND_VLD1 - Valid Bit Vector 1 for TLBPEND registers	
ZTLB_VLD_1 - Valid Bit Vector 1 for Z	
CASCTLB_VLD_2 - Valid Bit Vector 2 for CASC	
L3TLB_VLD_2 - Valid Bit Vector 2 for L3	
ZTLB_VLD_2 - Valid Bit Vector 2 for Z	
CASCTLB_VLD_3 - Valid Bit Vector 3 for CASC	
L3TLB_VLD_3 - Valid Bit Vector 3 for L3	
ZTLB_VLD_3 - Valid Bit Vector 3 for Z	
CASCTLB_VLD_4 - Valid Bit Vector 4 for CASC	
L3TLB_VLD_4 - Valid Bit Vector 4 for L3	
L3TLB_VLD_5 - Valid Bit Vector 5 for L3	
L3TLB_VLD_6 - Valid Bit Vector 6 for L3	
L3TLB_VLD_7 - Valid Bit Vector 7 for L3	
BCSTLB_VLD - Valid Bit Vector for BCS TLB	
BLBTLB_VLD - Valid Bit Vector for BLB TLB	
CTX_TLB_VLD - Valid Bit Vector for CTX TLB	
PDTLB_VLD - Valid Bit Vector for PD TLB	
VCS_CXT_SIZE - VCS Context Sizes	
VCS_CNTR - VCS Counter for the bit stream decode engine	
VCS_EIR - VCS Error Identity Register	
VCS_EMR - VCS Error Mask Register	
VCS_ESR - VCS Error Status Register	
VCS_EXCC - VCS Execute Condition Code Register	
VCS_HWSTAM - VCS Hardware Status Mask Register	
VCS_PWRCTX_MAXCNT - VCS IDLE Max Count	
VCS_INSTPM - VCS Instruction Parser Mode Register	





VCS_IMR - VCS Interrupt Mask Register	
VCS_MI_MODE - VCS Mode Register for Software Interface	
VCS_PP_DCLV - VCS PPGTT Directory Cacheline Valid Register	
VCS_TLBPEND_RDY0 - VCS Ready Bit Vector 0 for TLBPEND Registers	
VCS_TLBPEND_RDY1 - VCS Ready Bit Vector 1 for TLBPEND Registers	
VCS_TIMESTAMP - VCS Reported Timestamp Count	
VCS_RNCID - VCS Ring Buffer Next Context ID Register	
VCS_TLBPEND_SEC0 - VCS Section 0 of TLBPEND Entry	
VCS_TLBPEND_SEC1 - VCS Section 1 of TLBPEND Entry	
VCS_TLBPEND_SEC2 - VCS Section 2 of TLBPEND Entry	
VCS_THRSH - VCS Threshold for the counter of bit stream decode engine	
VCS_TLBPEND_VLD0 - VCS Valid Bit Vector 0 for TLBPEND Registers	
VCS_TLBPEND_VLD1 - VCS Valid Bit Vector 1 for TLBPEND Registers	
VFSKPD - VF Scratch Pad	
VICTLB_VA - VIC Virtual page Address Registers	
VBSYNC - Video/Blitter Semaphore Sync Register	
MFX_MODE - Video Mode Register	
VRSYNC - Video/Render Semaphore Sync Register	
VS_INVOCATION_COUNT - VS Invocation Counter	
SYNC_FLIP_STATUS - Wait For Event and Display Flip Flags Register	
SYNC_FLIP_STATUS_1 - Wait For Event and Display Flip Flags Register 1	
PR_CTR_CTL - Watchdog Counter Control	405
PR_CTR_THRSH - Watchdog Counter Threshold	
WR_WATERMARK - Write Watermark	
ZTLB_LRA_0 - ZTLB LRA 0	
ZTLB_LRA_1 - ZTLB LRA 1	



#### **BBA\_LEVEL2 - 2nd Level Batch Buffer Address**

	Format:		MBZ		
1:0	Reserved				
Pointer to the WA Batch Buffer Address.					
	Format:		U30		
31:2	WA Batch Buffer Address				
Bit	Description				
This register is to read the current value of the 2nd level batch buffer address. Since the 2nd level batch buffer logic is shared with the C6 work-around batch buffer, this also shows the work-around address when it is active.					
12144h					
1					
32					
R/W					
0x0000000					
VideoCS					
MMIO: 0/2/0					
	VideoCS 0x000000 R/W 32 1 12144h ad the curre the C6 work Bit 31:2	VideoCS 0x00000000 R/W 32 1 12144h ad the current value of the 2nd level back the C6 work-around batch buffer, this ac Bit 31:2 WA Batch Buffer Addre Format: Pointer to the WA Batch 1:0 Reserved	VideoCS 0x0000000 R/W 32 1 12144h ad the current value of the 2nd level batch buffer address. Sin the C6 work-around batch buffer, this also shows the work-ar Bit Description 31:2 WA Batch Buffer Address Format: Pointer to the WA Batch Buffer Address. 1:0 Reserved		



Register							
Default Value: 0x0000000							
Size (in b	oits):	32					
Address: 04030h-04033h							
ARB_MC	DDE -	Arbiter Mode Control register					
DWord	Bit		Description				
0	31:16	Mask Bits	l .				
		Default Value:	d00000000000000b				
		Access:	RO				
		Mask bits					
		Format: U16					
		Mask bits act as write enables for th	e bits in the lower bits of thi	s register			
	15	Reserved					
	14	GAM to Bypass GTT Translation					
		Default Value:		0b			
		Access:		R/W			
		GAM to bypass GTT translation and pass logical addresses through with 0's padded on the MSBs to form the physical address					
·	13	DC GDR					
		Default Value:		0b			
		Access:		R/W			
		DC GDR					
	12	HIZ GDR					
		Default Value:		0b			
		Access:		R/W			
		HIZ GDR					
·	11	STC GDR					
		Default Value:		0b			
		Access:		R/W			
		STC GDR					



10	BLB GDR	
	Default Value:	0b
	Access:	R/W
	BLB GDR	
9	GAM PD GDR	
5	Default Value:	0b
	Access:	R/W
	GAM PD GDR	
8	Extra Register	
0	Default Value:	0b
	Access:	R/W
	Reserve Bit	r/ w
7:6	Reserved	
5	Address Swizzling for Tiled Surfaces	
-	Default Value:	0b
		00
	Access:	
	Access: Address Swizzling for Tiled-Surfaces	R/W
	Address Swizzling for Tiled-Surfaces	R/W DRAM accesses. Driver M configuration
	Address Swizzling for Tiled-Surfaces Format: U1 This register location is updated via GFX Driver prior to enabling needs to obtain the need for memory address swizzling via DRA	R/W DRAM accesses. Driver M configuration
	Address Swizzling for Tiled-Surfaces Format: U1 This register location is updated via GFX Driver prior to enabling needs to obtain the need for memory address swizzling via DRA registers and set the following bits (in Display Engine and Rende	R/W DRAM accesses. Driver M configuration
4	Address Swizzling for Tiled-Surfaces Format: U1 This register location is updated via GFX Driver prior to enabling needs to obtain the need for memory address swizzling via DRA registers and set the following bits (in Display Engine and Rende 0: No address Swizzling	R/W DRAM accesses. Driver M configuration



	Access:	R/W			
	When this bit is set, Data requested from the VM	When this bit is set, Data requested from the VMC client will be generated by the GDR			
	algorithm				
3	Texture (MT) Cache GDR Enable bit				
	Default Value:	Ob			
	Access:	R/W			
	Texture Cache GDR enable bit				
	Format: U1				
	When this bit is set, Data requested from the Tex the GDR algorithm	xture Cache client will be generated by			
2	Depth (RCZ) Cache GDR Enable bit				
	Default Value:	Ob			
	Access:	R/W			
	Depth Cache GDR enable bit				
	Depth Cache GDR enable bit				
	Depth Cache GDR enable bit Format: U1				
	Format: U1				
	Format: U1 When this bit is set, Data requested from the De				
	Format: U1				
1	Format: U1 When this bit is set, Data requested from the De				
1	Format: U1 When this bit is set, Data requested from the De the GDR algorithm (See GDR algorithm in xxx see				
1	Format: U1 When this bit is set, Data requested from the De the GDR algorithm (See GDR algorithm in xxx sec Color (RCC) Cache GDR Enable bit	ction)			
1	Format: U1 When this bit is set, Data requested from the Dep the GDR algorithm (See GDR algorithm in xxx sec Color (RCC) Cache GDR Enable bit Default Value:	ction)			
1	Format: U1 When this bit is set, Data requested from the De the GDR algorithm (See GDR algorithm in xxx sec Color (RCC) Cache GDR Enable bit Default Value: Access: Color Cache GDR enable bit	ction)			
1	Format: U1 When this bit is set, Data requested from the De the GDR algorithm (See GDR algorithm in xxx sec Color (RCC) Cache GDR Enable bit Default Value: Access:	ction)			
1	Format: U1 When this bit is set, Data requested from the Dep the GDR algorithm (See GDR algorithm in xxx sec Color (RCC) Cache GDR Enable bit Default Value: Access: Color Cache GDR enable bit Format: U1	Ction)			
1	Format: U1 When this bit is set, Data requested from the De the GDR algorithm (See GDR algorithm in xxx sec Color (RCC) Cache GDR Enable bit Default Value: Access: Color Cache GDR enable bit	ob R/W			
1	Format: U1 When this bit is set, Data requested from the Dep the GDR algorithm (See GDR algorithm in xxx sec Color (RCC) Cache GDR Enable bit Default Value: Access: Color Cache GDR enable bit Format: U1 When this bit is set, Data requested from the Col	ob R/W			
	Format: U1 When this bit is set, Data requested from the Dep the GDR algorithm (See GDR algorithm in xxx sec Color (RCC) Cache GDR Enable bit Default Value: Access: Color Cache GDR enable bit Format: U1 When this bit is set, Data requested from the Col GDR algorithm (See GDR algorithm in xxx section	ob R/W			
	Format: U1 When this bit is set, Data requested from the Dep the GDR algorithm (See GDR algorithm in xxx sec Color (RCC) Cache GDR Enable bit Default Value: Access: Color Cache GDR enable bit Format: U1 When this bit is set, Data requested from the Col GDR algorithm (See GDR algorithm in xxx section GTT Accesses GDR	lor Cache client will be generated by t			
	Format: U1 When this bit is set, Data requested from the Dep the GDR algorithm (See GDR algorithm in xxx sec Color (RCC) Cache GDR Enable bit Default Value: Access: Color Cache GDR enable bit Format: U1 When this bit is set, Data requested from the Col GDR algorithm (See GDR algorithm in xxx section GTT Accesses GDR Default Value:	lor Cache client will be generated by t			
	Format: U1 When this bit is set, Data requested from the Dep the GDR algorithm (See GDR algorithm in xxx sec Color (RCC) Cache GDR Enable bit Default Value: Access: Color Cache GDR enable bit Format: U1 When this bit is set, Data requested from the Col GDR algorithm (See GDR algorithm in xxx section GTT Accesses GDR Default Value: Access:	lor Cache client will be generated by t			



When this bit is enabled along with the Client's GDR bit, PPGTT and GGTT requests for this memory access will also be tagged as GDR to SQ.



				-	
Register	Space	MMIO: 0/2/0			
Source:		RenderCS			
Default <b>`</b>	Value:				
Size (in l	oits):				
Trusted Type: 1					
Address	:	04030h			
DWord Bit			Description		
0	31:16	Mask Bits			
		Default Value:	000000000000000000b		
		Access:	RO		
		Format:	U16		
		Mask bits act as write enables for the bits in the lower bits of this register			
	15	Reserved			
	14	GAM to Bypass GTT Translation (G	AM2BGTTT)	_	
		Default Value:		0b	
		Access:		R/W	
		Format:		MBZ	
		GAM to bypass GTT translation and to form the physical address.	pass logical addresses through w	vith 0's padded on the MSBs	
	13	DC GDR (DC_GDR)			
		Default Value:		0b	
		Access:		R/W	
		DC GDR			
	12	HIZ GDR (HIZ_GDR)			
		Default Value:		0b	
		Access:		R/W	
		HIZ GDR			
	11	STC GDR (STC_GDR)			
		Default Value:		0b	
		Access:		R/W	
		Format:		U1	
		STC GDR			
	10	BLB GDR (STC_GDR)			



Access:         BLB GDR         9       GAM PD GDR (GAMPD_GDR)         Default Value:         Access:         GAM PD GDR         8       Color/Depth Port Share Bit (CDPS)         Default Value:         Access:         Format:         Color/Depth port share bit         This bit is used to force Color and Depth Caches to si         (Bit = 0) the Color Cache will NOT share the read requined         7:6       Reserved         5       Address Swizzling for Tiled Surfaces (AS4TS)         Access:       Format:         Address Swizzling for Tiled-Surfaces. This register loce         enabling DRAM accesses. Driver needs to obtain the top         DRAM configuration registers and set the following be         access streams).       Value         Value       N         0b       No address Swizzling         1b       Address bit[6] needs to be swizzled for ti         4       VMC GDR Enable (VMC_GDR_EN)         Access:       When this bit is set, Data requested from the VMC cliver         3       Texture Cache GDR Enable bit (TCGDREN)	R/W				
9       GAM PD GDR (GAMPD_GDR)         Default Value:       Access:         GAM PD GDR       GAM PD GDR         8       Color/Depth Port Share Bit (CDPS)         Default Value:       Access:         Access:       Format:         Color/Depth port share bit       This bit is used to force Color and Depth Caches to si (Bit = 0) the Color Cache will NOT share the read requination of the color Cache will NOT share the read requination of the color Cache will NOT share the read requination of the color cache shows and set the following between the configuration registers and set the following between the configuration registers and set the following between the color cache shows and set the following between the color cache shows and set the following between the color cache shows and set the following between the color cache shows and set the following between the color cache shows and set the following between the color cache shows and set the following between the cache shot cache shows and set the following between the color cache shows and set the following between the cache shot cache shows and set the following between the cache shot cache shows and set the following between the cache shot cache shows and set the following between the cache shot c					
Default Value:         Access:         GAM PD GDR         8       Color/Depth Port Share Bit (CDPS)         Default Value:         Access:         Format:         Color/Depth port share bit         This bit is used to force Color and Depth Caches to si         (Bit = 0) the Color Cache will NOT share the read requined         7:6       Reserved         5       Address Swizzling for Tiled Surfaces (AS4TS)         Access:       Format:         Address Swizzling for Tiled-Surfaces. This register loc         enabling DRAM accesses. Driver needs to obtain the to         DRAM configuration registers and set the following be         access streams).         Value       N         0b       No address Swizzling         1b       Address bit[6] needs to be swizzled for ti         4       VMC GDR Enable (VMC_GDR_EN)         Access:       When this bit is set, Data requested from the VMC clive         3       Texture Cache GDR Enable bit (TCGDREN)					
Access:         GAM PD GDR         8       Color/Depth Port Share Bit (CDPS)         Default Value:         Access:         Format:         Color/Depth port share bit         This bit is used to force Color and Depth Caches to si         (Bit = 0) the Color Cache will NOT share the read requination         7:6       Reserved         5       Address Swizzling for Tiled Surfaces (AS4TS)         Access:       Format:         Address Swizzling for Tiled-Surfaces. This register loce         enabling DRAM accesses. Driver needs to obtain the to         DRAM configuration registers and set the following biaccess streams).         Value       N         0b       No address Swizzling         1b       Address bit[6] needs to be swizzled for ti         4       VMC GDR Enable (VMC_GDR_EN)         Access:       When this bit is set, Data requested from the VMC clive         3       Texture Cache GDR Enable bit (TCGDREN)					
GAM PD GDR         8       Color/Depth Port Share Bit (CDPS)         Default Value:         Access:         Format:         Color/Depth port share bit         This bit is used to force Color and Depth Caches to sit         (Bit = 0) the Color Cache will NOT share the read requination         7:6       Reserved         5       Address Swizzling for Tiled Surfaces (AS4TS)         Access:       Format:         Address Swizzling for Tiled-Surfaces. This register loce         enabling DRAM accesses. Driver needs to obtain the orticle of the port of t	0b				
8       Color/Depth Port Share Bit (CDPS)         Default Value:       Access:         Access:       Format:         Color/Depth port share bit       This bit is used to force Color and Depth Caches to sit (Bit = 0) the Color Cache will NOT share the read requined         7:6       Reserved         5       Address Swizzling for Tiled Surfaces (AS4TS)         Access:       Format:         Address Swizzling for Tiled-Surfaces. This register loce enabling DRAM accesses. Driver needs to obtain the to DRAM configuration registers and set the following be access streams).         Value       N         0b       No address Swizzling         1b       Address Swizzling         4       VMC GDR Enable (VMC_GDR_EN)         Access:       When this bit is set, Data requested from the VMC clients of the top	R/W				
Default Value:         Access:         Format:         Color/Depth port share bit         This bit is used to force Color and Depth Caches to sit         (Bit = 0) the Color Cache will NOT share the read requination         7:6         Reserved         5         Address Swizzling for Tiled Surfaces (AS4TS)         Access:         Format:         Address Swizzling for Tiled-Surfaces. This register loce         enabling DRAM accesses. Driver needs to obtain the read DRAM configuration registers and set the following be         access streams).         Value         0b       No address Swizzling         1b       Address bit[6] needs to be swizzled for tit         4       VMC GDR Enable (VMC_GDR_EN)         Access:       When this bit is set, Data requested from the VMC clive         3       Texture Cache GDR Enable bit (TCGDREN)					
Access:         Format:         Color/Depth port share bit         This bit is used to force Color and Depth Caches to si         (Bit = 0) the Color Cache will NOT share the read requination         7:6         Reserved         5         Address Swizzling for Tiled Surfaces (AS4TS)         Access:         Format:         Address Swizzling for Tiled-Surfaces. This register loce         enabling DRAM accesses. Driver needs to obtain the to         DRAM configuration registers and set the following be         access streams).         Value       N         0b       No address Swizzling         1b       Address bit[6] needs to be swizzled for ti         4       VMC GDR Enable (VMC_GDR_EN)         Access:       When this bit is set, Data requested from the VMC clive         3       Texture Cache GDR Enable bit (TCGDREN)					
Format:         Color/Depth port share bit         This bit is used to force Color and Depth Caches to si         (Bit = 0) the Color Cache will NOT share the read requination         7:6 <b>Reserved</b> 5 <b>Address Swizzling for Tiled Surfaces (AS4TS)</b> Access:         Format:         Address Swizzling for Tiled-Surfaces. This register lock         enabling DRAM accesses. Driver needs to obtain the read need to be contain the read configuration registers and set the following beaccess streams).         Value       N         0b       No address Swizzling         1b       Address Swizzling         1b       Address Swizzling         3       Texture Cache GDR Enable bit (TCGDREN)         Access:       Access:	00b				
Color/Depth port share bit         This bit is used to force Color and Depth Caches to sit         (Bit = 0) the Color Cache will NOT share the read requination         7:6       Reserved         5       Address Swizzling for Tiled Surfaces (AS4TS)         Access:       Format:         Address Swizzling for Tiled-Surfaces. This register lock         enabling DRAM accesses. Driver needs to obtain the read requination registers and set the following beaccess streams).         Value       N         0b       No address Swizzling         1b       Address bit[6] needs to be swizzled for till         4       VMC GDR Enable (VMC_GDR_EN)         Access:       When this bit is set, Data requested from the VMC cline         3       Texture Cache GDR Enable bit (TCGDREN)	R/W				
This bit is used to force Color and Depth Caches to sit (Bit = 0) the Color Cache will NOT share the read required         7:6       Reserved         5       Address Swizzling for Tiled Surfaces (AS4TS)         Access:       Format:         Address Swizzling for Tiled-Surfaces. This register loce         enabling DRAM accesses. Driver needs to obtain the red         DRAM configuration registers and set the following be         access streams).         Value       N         0b       No address Swizzling         1b       Address bit[6] needs to be swizzled for tild         4       VMC GDR Enable (VMC_GDR_EN)         Access:       When this bit is set, Data requested from the VMC clied         3       Texture Cache GDR Enable bit (TCGDREN)	U1				
Access:         Format:         Address Swizzling for Tiled-Surfaces. This register loc         enabling DRAM accesses. Driver needs to obtain the r         DRAM configuration registers and set the following b         access streams).         Value         0b       No address Swizzling         1b       Address bit[6] needs to be swizzled for ti         4       VMC GDR Enable (VMC_GDR_EN)         Access:       When this bit is set, Data requested from the VMC cline         3       Texture Cache GDR Enable bit (TCGDREN)         Access:       Access:					
Format:         Address Swizzling for Tiled-Surfaces. This register loc enabling DRAM accesses. Driver needs to obtain the of DRAM configuration registers and set the following be access streams).         Value       N         0b       No address Swizzling         1b       Address bit[6] needs to be swizzled for ti         4       VMC GDR Enable (VMC_GDR_EN)         Access:       When this bit is set, Data requested from the VMC client         3       Texture Cache GDR Enable bit (TCGDREN)         Access:       Access:	Address Swizzling for Tiled Surfaces (AS4TS)				
Address Swizzling for Tiled-Surfaces. This register loc         enabling DRAM accesses. Driver needs to obtain the n         DRAM configuration registers and set the following b         access streams).         Value         0b       No address Swizzling         1b       Address bit[6] needs to be swizzled for ti         4       VMC GDR Enable (VMC_GDR_EN)         Access:       When this bit is set, Data requested from the VMC clive         3       Texture Cache GDR Enable bit (TCGDREN)         Access:       Access:	R/W				
<ul> <li>enabling DRAM accesses. Driver needs to obtain the in DRAM configuration registers and set the following be access streams).</li> <li>Value</li> <li>No address Swizzling</li> <li>1b</li> <li>Address bit[6] needs to be swizzled for ti</li> <li>4</li> <li>VMC GDR Enable (VMC_GDR_EN)</li> <li>Access:</li> <li>When this bit is set, Data requested from the VMC client</li> <li>3</li> <li>Texture Cache GDR Enable bit (TCGDREN)</li> <li>Access:</li> </ul>					
0b       No address Swizzling         1b       Address bit[6] needs to be swizzled for ti         4       VMC GDR Enable (VMC_GDR_EN)         Access:       When this bit is set, Data requested from the VMC clive         3       Texture Cache GDR Enable bit (TCGDREN)         Access:       Access:	eed for memory address swizzling				
1b       Address bit[6] needs to be swizzled for ti         4       VMC GDR Enable (VMC_GDR_EN)         Access:       When this bit is set, Data requested from the VMC clive         3       Texture Cache GDR Enable bit (TCGDREN)         Access:       Access:	me				
<ul> <li>4 VMC GDR Enable (VMC_GDR_EN)         <ul> <li>Access:</li> <li>When this bit is set, Data requested from the VMC clivit</li> <li>3 Texture Cache GDR Enable bit (TCGDREN)</li></ul></li></ul>					
Access: When this bit is set, Data requested from the VMC clives 3 Texture Cache GDR Enable bit (TCGDREN) Access:	ed surfaces				
When this bit is set, Data requested from the VMC clives a set of the VMC cliv	1				
3 Texture Cache GDR Enable bit (TCGDREN) Access:	R/W				
Access:	nt will be generated by the GDR alo				
	1				
Format	R/W				
	U1				
Texture Cache GDR enable bit When this bit is set, Da will be generated by the GDR algorithm (See GDR alg	•				



	Access:	R/W		
	Format:	U1		
	When this bit is set, Data requested for algorithm (See the GDR algorithm see	•	be generated by the GDR	
1	Color Cache GDR enable bit(CCGDF	REN)		
	Access:	R/W		
	Format:	U1		
	When this bit is set, Data requested from the Color Cache client will be generated by the GDR algorithm (See GDR algorithm in xxx section)			
0	GTT Accesses GDR (GTTAGDR )			
	Default Value:		0b	
	Access:		R/W	
	Format:	U1		
	When this bit is enabled along with the memory access will also be tagged as		GTT requests for this	



#### 3DPRIM\_END\_OFFSET - Auto Draw End Offset

Register Space:		e: MMIO: 0/2/0		
Source:		RenderCS		
Default '	Value:	0x0000000		
Access:		R/W		
Size (in l	oits):	32		
Address: 02420h-02423h				
DWord	Bit		Description	
0	31:0	End Offset		
		U32		
	This register is used to store the end offset value used by the Vertex Fetch to determine whe stop processing the 3D_PRIMITIVE command. This register is valid when the End Offset Enable set in the 3D_PRIMITIVE command.			



## **BB\_ADDR - Batch Buffer Head Pointer Register**

Register	Space	e: MMIO: 0/2,	MMIO: 0/2/0				
Source:		BSpec	BSpec				
Default '	Value:	0x0000000	0				
Access:		RO					
Size (in l	bits):	32					
Address	:	02140h					
Name:		RCS Batch	Buffer Head Pointer Register				
ShortNa	ime:	RCS_BB_AD	DDR				
Address	:	12140h					
Name:		VCS Batch	Buffer Head Pointer Register				
ShortNa	ime:	VCS_BB_AD	DDR				
Address	:	22140h					
Name:		BCS Batch I	Buffer Head Pointer Register				
ShortNa	ime:	BCS_BB_AD	DDR				
This reg	gister (	contains the current	t DWord Graphics Memory Address of the last-initiated batch buffer.				
			Programming Notes				
<b>Progra</b> only.	mmin	ig Restriction: This	register should NEVER be programmed by driver. This is for HW internal use				
_	mmin Bit	ng Restriction: This	register should NEVER be programmed by driver. This is for HW internal use Description				
only.	1		Description				
only. DWord	Bit		Description				
only. DWord	Bit	Batch Buffer Head	Description d Pointer				
only. DWord	Bit	Batch Buffer Head	Description d Pointer BlitterCS, VideoCS				
only. DWord	Bit	Batch Buffer Head Source: Exists If: Format: This field specifies	Description         d Pointer         BlitterCS, VideoCS         //BCS, VCS         GraphicsAddress[31:3]         s the DWord-aligned Graphics Memory Address where the last initiated Batch fetching commands. If no batch buffer is currently active, the Valid bit will be 0				
only. DWord	<b>Bit</b> 31:3	Batch Buffer Head Source: Exists If: Format: This field specifies Buffer is currently f	Description         d Pointer         BlitterCS, VideoCS         //BCS, VCS         GraphicsAddress[31:3]         s the DWord-aligned Graphics Memory Address where the last initiated Batch fetching commands. If no batch buffer is currently active, the Valid bit will be 0 be meaningless.				
only. DWord	<b>Bit</b> 31:3	Batch Buffer Head Source: Exists If: Format: This field specifies Buffer is currently f and this field will b	Description         d Pointer         BlitterCS, VideoCS         //BCS, VCS         GraphicsAddress[31:3]         s the DWord-aligned Graphics Memory Address where the last initiated Batch fetching commands. If no batch buffer is currently active, the Valid bit will be 0 be meaningless.				
only. DWord	<b>Bit</b> 31:3	Batch Buffer Head Source: Exists If: Format: This field specifies Buffer is currently f and this field will b Batch Buffer Head	Description         d Pointer         BlitterCS, VideoCS         //BCS, VCS         GraphicsAddress[31:3]         is the DWord-aligned Graphics Memory Address where the last initiated Batch fetching commands. If no batch buffer is currently active, the Valid bit will be 0 be meaningless.         d Pointer				
only. DWord	<b>Bit</b> 31:3	Batch Buffer Head Source: Exists If: Format: This field specifies Buffer is currently f and this field will b Batch Buffer Head Source:	Description         d Pointer         BlitterCS, VideoCS         //BCS, VCS         GraphicsAddress[31:3]         is the DWord-aligned Graphics Memory Address where the last initiated Batch fetching commands. If no batch buffer is currently active, the Valid bit will be 0 be meaningless.         d Pointer         RenderCS				
only. DWord	<b>Bit</b> 31:3	Batch Buffer Head         Source:         Exists If:         Format:         This field specifies         Buffer is currently frand this field will b         Batch Buffer Head         Source:         Exists If:         Format:         This field specifies         This field specifies         This field specifies	Description         d Pointer         BlitterCS, VideoCS         //BCS, VCS         GraphicsAddress[31:3]         is the DWord-aligned Graphics Memory Address where the last initiated Batch fetching commands. If no batch buffer is currently active, the Valid bit will be 0 or meaningless.         d Pointer         RenderCS         //RCS         GraphicsAddress[31:2]         is the DWord-aligned Graphics Memory Address where the last initiated Batch fetching commands. If no batch buffer is currently active, the Valid bit will be 0				
only. DWord	<b>Bit</b> 31:3	Batch Buffer Head         Source:         Exists If:         Format:         This field specifies         Buffer is currently fand this field will b         Batch Buffer Head         Source:         Exists If:         Format:         This field specifies         Buffer is currently fand this field will b	Description         d Pointer         BlitterCS, VideoCS         //BCS, VCS         GraphicsAddress[31:3]         is the DWord-aligned Graphics Memory Address where the last initiated Batch fetching commands. If no batch buffer is currently active, the Valid bit will be 0 or meaningless.         d Pointer         RenderCS         //RCS         GraphicsAddress[31:2]         is the DWord-aligned Graphics Memory Address where the last initiated Batch fetching commands. If no batch buffer is currently active, the Valid bit will be 0				



## **BB\_ADDR - Batch Buffer Head Pointer Register**

	Exists If:	//BCS, VCS	//BCS, VCS			
	Format:	MBZ				
1 Reserved						
	Format:			MBZ		
0 Valid						
	Format:				U1	
	Value	Name			Description	
	0h	Invalid <b>[Default]</b>	В	atch buffer	Invalid	
	1h	Valid	В	atch buffer	Valid	



## **BB\_STATE - Batch Buffer State Register**

Register Spa	ace	ce: MMIO: 0/2/0						
Source:		Blitt	erCS, VideoCS					
Default Value: 0x0000000								
Access:		RO						
Size (in bits)	):	32						
Address:		121	10h					
Name:		VCS	Batch Buffer State Register					
ShortName:	:	VCS	BB_STATE					
Address:		221	10h					
Name:		BCS	Batch Buffer State Register					
ShortName:	:	BCS	_BB_STATE					
This registe	er co	ontains the	attributes of the current batch buffe	er initiated	d from the Ring Buffer.			
DWord Bi	it		Desc	cription				
0 31	.:7 I	Reserved						
		Format:			MBZ			
6	5	2nd Level	Buffer Security Indicator					
		Source:		VideoCS				
		Exists If: //VCS						
		If set, VCS is fetching 2nd level batch commands from a PPGTT address space. If clear, this batch						
			cure and will be accessed via the GG	TT. Note:				
		Value	Name		Description			
		0h	MIBUFFER_SECURE [Default]		Located in GGTT memory			
		1h	MIBUFFER_NONSECURE		Located in PPGTT memory			
6	5	Reserved						
6	l r	Reserved						
		Source:		BlitterCS				
		Exists If:		//BCS				
		Format:		MBZ				
5	; ;		Suffer Security Indicator					
		Format: MI_1stBufferSecurityType						
			5		m a PPGTT address space. If clear, this			
		Value	er is secure and will be accessed Name	via the G	Description			
		0h			-			
	ŀ		MIBUFFER_SECURE [Default]		Located in GGTT memory			
		1h	MIBUFFER_NONSECURE		Located in PPGTT memory			
4	ł	Reserved						



## **BB\_STATE - Batch Buffer State Register**

	Source:	BlitterCS
	Exists If:	//BCS
	Format:	MBZ
4	Reserved	
3:0	Reserved	
	Format:	MBZ



### **BCS\_CXT\_SIZE - BCS Context Sizes**

Register Space:	MMIO: 0/2/0
Source:	BlitterCS
Default Value:	0x0000402
Access:	Read/32 bit Write Only
Size (in bits):	32
Address:	221A8h

DWord	Bit	Description			
0	31:13	Reserved			
		Format:		MBZ	2
	12:8	BCS Context Size			
		Format:			U5
					1
		Value	Name		
		4h	[Default]		
	7:5	Reserved		1	
		Format: Reserved		MBZ	7
	4:0				



#### **BCS\_EIR - BCS Error Identity Register**

Register Space:	MMIO: 0/2/0
Source:	BlitterCS
Default Value:	0x0000000
Access:	R/WC
Size (in bits):	32
Address:	220B0h

The EIR register contains the persistent values of Hardware-Detected Error Condition bits. Any bit set in this register will cause the Master Error bit in the ISR to be set. The EIR register is also used by software to clear detected errors (by writing a '1' to the appropriate bit(s) except for the unrecoverable bits described).).

DWord	Bit	Description					
0	31:16	Reserved					
		Format: MBZ					
	15:0	Error Iden	tity Bits				
		Format:	Array of Error condition bits See Ta	ble 1 5. Har	dware-Detected Error	r Bits	
		the EMR r Master Er must first	ter contains the persistent values register. The logical OR of all (def ror bit of the Interrupt Status Reg clear the error by writing a '1' to should then proceed to clear the Name	ined) bits i gister. To c the appro Master Err	n this register is rep lear an error condit priate bit(s) in this f	ported in the ion, software	
		0h	[Default]				
		1h	Error occurred	Error occu	rred		
		) Muriting of		nming Note		the Instruction	
			'1' to a set bit will cause that error of Bit 0) cannot be cleared except by re				



BCS_EMR -	<b>BCS Error</b>	Mask	Register

Register Spa	ace:	MMIO: 0/2/	MMIO: 0/2/0						
Source:		BlitterCS							
Default Valu	ie:	0x0000FFFF							
Access:		R/W							
Size (in bits)	):	32							
Address:		220B4h							
The EMR register is used by software to control which Error Status Register bits are "masked" or "unmasked". "Unmasked" bits will be reported in the EIR, thus setting the Master Error ISR bit and possibly triggering a CPU interrupt, and will persist in the EIR until cleared by software. "Masked" bits will not be reported in the EIR and therefore cannot generate Master Error conditions or CPU interrupts.									
DWord	Bit	ed bits in the Hardware Detected Error Bit Table will always return a read value of '1'							
0	31:16	Description Reserved							
0	51.10	Default Val	110.		0000h				
		Format:	uc.		MBZ				
	15:0	Error Mask							
			Format: Array of error condition mask bits See Table 1 5. Hardware-Detected Error Bits						
			This register contains a bit mask that selects which error condition bits (from the ESR) are						
		Value	reported in the EIR. Value Name Description						
		0000h	Not Masked	Will be reported ir	-				
		FFFFh	Masked [Default]	Will not be reported in					
				will not be report					



#### **BCS\_ESR - BCS Error Status Register**

Register Space:	MMIO: 0/2/0
Source:	BlitterCS
Default Value:	0x0000000
Access:	RO
Size (in bits):	32
Address:	220B8h

The ESR register contains the current values of all Hardware-Detected Error condition bits (these are all by definition "persistent"). The EMR register selects which of these error conditions are reported in the persistent EIR (i.e., set bits must be cleared by software) and thereby causing a Master Error interrupt condition to be reported in the ISR.

DWord	Bit		Description					
0	31:16	Reserved	Reserved					
		Format:	Format: MBZ					
	15:0	Error Sta	Error Status Bits					
		Format:	Format: Array of error condition bits See Table 1 5. Hardware-Detected Error Bits					
		This regis bits.	This register contains the non-persistent values of all hardware-detected error condition bits.					
		Value	Name		Description			
		0h	[Default]					
		1h	Error Condition Detected	Error Con	dition detected			



#### **BCS\_EXCC - BCS Execute Condition Code Register**

Register Space:	MMIO: 0/2/0
Source:	BlitterCS
Default Value:	0x0000000
Access:	R/W,RO
Size (in bits):	32
Trusted Type:	1
Address:	22028h

This register contains user defined and hardware generated conditions that are used by MI\_WAIT\_FOR\_EVENT commands. An MI\_WAIT\_FOR\_EVENT instruction excludes the executing ring from arbitration if the selected event evaluates to a 1, while instruction is discarded if the condition evaluates to a 0. Once excluded, a ring is enabled into arbitration when the selected condition evaluates to a 0. This register also contains control for the invalidation of indirect state pointers on context restore.

DWord	Bit		Description					
0	31:16	Mask Bits						
		Format: Mask[15:0]						
		These bits serves as a write enable for b	oits 15:0.					
		If this register is written with any of the						
		corresponding bit in the field 15:0 will	not be modified.					
		Reading these bits always returns 0s.						
	15	Reserved						
		Format:	MB	MBZ				
	14:12	Reserved						
		Format:	MB	ΒZ				
	11:5	Reserved						
		Format:	MB	ΒZ				
	4:0	User Defined Condition Codes						
		Format:			U5			
		The software may signal a Stream Sema to match the bit field specified in a WA			5 5			



# **BCS\_HWSTAM - BCS Hardware Status Mask Register**

				_				
Register Space:	M	MIO: 0/2/0	O: 0/2/0					
Source:	Bli	tterCS	rCS					
Default Value:	0xl	FFFFFFF						
Access:	R/\	N						
Size (in bits):	32							
Trusted Type:	1							
Address:	22	098h						
Access: RO for	Reserved	Control bits						
"mask" bits that prevent the corresponding bits in the Interrupt Status Register from generating a "Hardware Status Write" (PCI write cycle). Any unmasked interrupt bit (HWSTAM bit set to 0) will allow the Interrupt Status Register to be written to the ISR location (within the memory page specified by the Hardware Status Page Address Register) when that Interrupt Status Register bit changes state.								
Programming Notes								
To write the interrupt to the HWSP, the corresponding IMR bit must also be clear (enabled). At most 1 bit can be unmasked at any given time.								
DWord	Bit		Description					
0	31:0	Hardware Status Mask Regi	ister					
		Default Value:		FFFFFFFh				
		Format:		Array of Masks				
		refer to Table 5-1 in Interrupt	efer to Table 5-1 in Interrupt Control Register section for bit definitions					



#### **BCS\_PWRCTX\_MAXCNT - BCS IDLE Max Count**

Register	Space:	MMIO:	MMIO: 0/2/0							
Source:		BlitterC	BlitterCS							
Default \	/alue:	0x0000	0x0000040							
Access:		R/W	R/W							
Size (in b	oits):	32	32							
Trusted <sup>*</sup>	Туре:	1	1							
Address:		22054h								
This regi IDLE	ster co	ntains the tim	e in 0.64us to wa	it before tellir	ng power mana	gement hardware the rend	ler pipe is			
DWord	Bit		Description							
0	31:20	Reserved								
		Format:				MBZ				
	19:0	Blitter IDLE	Nait Time							
		Format:		М	lax Count					
		•	long the comma jement hardware		ould wait befo	re ensuring the pipe is IDL	E and to let			
		Value	Name		Descrip	otion				
		00040h	[Default]	0x00040 * 0.	.64us ~ 41us wa	ait time				
Programming Notes										
Program										
		• This is	s only useable if <b>b</b>	oit 0 of the PC	_PSMI_CTRL is	clear.				
		• The v	alue in this field r	nust be great	er than 1.					



B	CS_	INSTPM - BCS	Instruction Pars	er M	lode Register					
Register	Space:	MMIO: 0/2/0								
Source:		BlitterCS								
Default \	/alue:	0x0000000								
Access:		R/W								
Size (in b	oits):	32								
Trusted <sup>-</sup>	Туре:	1								
Address:		220C0h								
Desc										
DWord	Bit		Description							
0	31:16	Mask Bits								
		Format:	Mask[15:0]							
		Must be set to modify corr	responding bit in Bits 15:0. (All i	mplemer	nted bits)					
	15:11	Reserved								
	10.11	Format:		MBZ						
	10	Reserved								
	10	Format:		MBZ						
	9	TLB Invalidate								
	9	Format:			U1					
		If set, this bit allows the command stream engine to invalidate the blitter TLBs. This bit is valid only with the Sync flush enable Note: GFX soft resets do not invalidate TLBs, it is up to GFX driver to explicitly invalidate TLBs post reset.								
	8:7	Reserved								
		Format:		MBZ						
	6	Memory Sync Enable								
		Format:			U1					
		This set, this bit allows the blitter decode engine to write out the data from the local caches to memory. This bit is not persistent. S/W must define this bit each time a sync flush is requested								
	5	Sync Flush Enable								
		Format:	U1							
		Format:	Enable Cleared by HW							
		-	st a Sync Flush operation. The d eration. See Sync Flush (Progran		-					
	4:0	Reserved								
		Reserved Format: MBZ								





## **BCS\_IMR - BCS Interrupt Mask Register**

Register Spa	gister Space: MMIO: 0/2/0							
Source:		BlitterCS						
Default Valu	ie:	0xFFFFFFFF						
Access:		R/W						
Size (in bits)	):	32						
Address:		220A8h						
"Unmasked	" bits wi	s used by software to control which Interrupt Status Register bits are "masked" or "unmasked". will be reported in the IIR, possibly triggering a CPU interrupt, and will persist in the IIR until re. "Masked" bits will not be reported in the IIR and therefore cannot generate CPU interrupts.						
0	31:0	Interrupt Mask Bits						
		Format: Array o section This field contain	Format:       Array of interrupt mask bits Refer to Table 5-1 in Interrupt Control Register section for bit definitions         This field contains a bit mask which selects which interrupt bits (from the ISR) are reported in					
			he IIR.					
		Value	Name	Description				
		FFFF FFFFh	[Default]					
		0h	Not Masked	Will be reported in the IIR				
		1h	Masked	Will not be reported in the IIR				



# **BCS\_MI\_MODE - BCS Mode Register for Software Interface**

					-				
Register	Space:	MMIO: 0/2/0							
Source:		Blitt	erCS						
Default <b>'</b>	Value:	0x0	0000200						
Access:		R/W	/						
Size (in bits): 32									
Address	:	220	9Ch-2209Fh						
The MI_l	MODE	register co	ntains informat	tion	that				
	1	are interfac	e aspects of th	e co	ommand parser.				
DWord	Bit		Description						
0	31:16	<ul> <li>Masks</li> <li>A 1 in a bit in this field allows</li> <li>the modification of the corresponding bit in Bits 15:0</li> </ul>							
	15	Suspend							
		Value	Name		D	escription			
		0h	No Delay		HW will not delay flush, this b will get cleared by MI_SUSPE				
		1h	Delay Flush		Suspend flush is active				
14:12 <b>Reserved</b> Read/Write									
	11								
	10	Reserved							
		Format:				MBZ			
	9	Ring Idle (Read Only Status Bit) Writes to this bit are not allowed.							
		١	/alue		1	Name			
		0		Par	ser not idle	er not idle			
		1 Parser idle [Default]							
	8	<b>Stop Ring</b> Software must set this bit to force the Ring and Command Parser to Idle. Software must read a 1 in Ring Idle bit after setting this bit to ensure that the hardware is idle. <i>Software must clear this bit for Ring to resume normal operation</i> .							
			Value			Name			
		0		N	ormal Operation				
		1		Pa	arser is turned off				
	7:2	Reserved							



## **BCS\_MI\_MODE - BCS Mode Register for Software Interface**

	Read/Write					
1	<b>Bypass Fence Write</b> If set, this bit will bypass all writes during flushes, independent of programming. This includes post-sync op bits, the implicit TLB invalidate write (set in GFX_MODE[13]), and sync flush fences. <i>Note this is only intended for work-arounds</i>					
	Value	Name				
	0	Normal Operation				
	1	Bypass				
0	Reserved					
	Read/Write					



### BCS\_PP\_DCLV - BCS PPGTT Directory Cacheline Valid Register

Register Space:	MMIO: 0/2/0
Source:	BlitterCS
Default Value:	0x00000000, 0x00000000
Access:	R/W
Size (in bits):	64
Address:	22220h

Default Value = 0h

This register controls update of the on-chip PPGTT Directory Cache during a context restore. Bits that are set will trigger the load of the corresponding 16 directory entry group. This register is restored with context (prior to restoring the on-chip directory cache itself). This register is also restored when switching to a context whose LRCA matches the current CCID if the Force PD Restore bit is set in the context descriptor.

The context image of this register must be updated and maintained by SW; SW should not normally need to read this register.

This register can also effectively be used to limit the size of a processes' virtual address space. Any access by a process that requires a PD entry in a set that is not enabled in this register will cause a fatal error, and no fetch of the PD entry will be attempted.

DWord	Bit	Description			
0	63:32	Reserved			
		Format:		MBZ	
	31:0	PPGTT Directory Cache Restore	-		
		Format:	Enable[32]		
	[132] 16 entries				
		If set, the [1st32nd] 16 entries of the directory cache are considered valid and will be brought in on context restore. If clear, these entries are considered invalid and fetch of these entries will not be attempted.			



BCS_TLBPEND_RDY0 - BCS Ready Bit Vector for TLBPEND Registers					
Register Space: MMIO: 0/		2/0			
Source: BlitterCS					
Default Value:	0x0000000				
Access:	RO				
Size (in bits):	32				
Trusted Type:	1				
Address: 24708h-2		470Bh			
This register contains the ready bits for entries 0-31 of TLBPEND structure (Cycles pending TLB translation).					
DWord		Bit	Description		
0		31:0	Ready bits per entry		



# **BCS\_TIMESTAMP - BCS Reported Timestamp Count**

Register Space:	MMI	MMIO: 0/2/0			
Source:	Blitte	rCS			
Default Value:	0x00	000000, 0x00000000			
Access:	RO. T	his register is not set by	the context restore.		
Size (in bits):	64				
Address:	2235	3h			
graphics reset. I Note: This time	This register provides an elapsed real-time value that can be used as a timestamp. This register is not reset l graphics reset. It will maintain its value unless a full chipset reset is performed. Note: This timestamp register reflects the value of the PCU TSC. The PCU TSC counts 10ns increments; this timestamp reflects bits 38:3 of the TSC (i.e. 80ns granularity, rolling over every 1.5 hours).			ounts 10ns increments; this	
DWord	Bit	Description			
0	63:36	3:36 Reserved			
		Format:		MBZ	
	35:0	5:0 <b>Timestamp Value</b>			
		Format:		U36	
		This register toggles e	very 80 ns. The upper 28 bits	are zero.	



#### **BCS\_RNCID - BCS Ring Buffer Next Context ID Register**

Register Space: N	MMIO: 0/2/0			
Source: B	litterCS			
Default Value: 0	x00000000, 0x0	000000		
Access: R	/W			
Size (in bits): 6	4			
Address: 2	2198h-2219Fh			
This register contains th	ne <i>next</i> ring con	text ID associated with the ring buffer.		
Programming Notes				
event. Note that the	The current context (RCCID) register can be updated indirectly from this register on a context switch event. Note that the only time a context switch can occur is when MI_ARB_CHECK enables preemption or the current context runs dry (head pointer becomes equal to tail pointer).			
DWord	Bit Description			
0	63:0	Unnamed		
		See Context Descriptor for BCS		



# **BCS\_TLBPEND\_SEC0 - BCS Section 0 of TLBPEND entry**

Register Space: MMIO: 0/2/0				
urce: BlitterCS				
'alue:	0x0000000			
	RO			
its):	32			
ype:	1			
Address: 24400h-24403h				
ster is d	lirectly mapped to the TLBPEND Array in the Graphic Arbiter.			
Bit	Description			
31:28	GTT bits			
	Bits 3:0 of the GTT entry used to translate the Virtual Address. 000 if translation is pending.			
27:0	<b>Current Address</b> The value of this field depends on the stage of the TLB translation for this entry: VA - bits 27:20 = 00, bits 19:0 = Bits 31:12 of the Virtual Address of the cycle.			
	/alue: its): [ype: ster is c Bit 31:28			



Register	Space:	MMIO: 0/2/0					
Source:		BlitterCS					
Default <b>\</b>	Value:	0x0000000					
Access:		RO					
Size (in b	oits):	32					
Trusted	Type:	1					
Address	:	24500h-24503h					
This regi	ister is	directly mapped to the current Virtua	al Addresses i	n the MTTLB (Texture and constant cache TLB).			
DWord	Bit		Descri	iption			
0	31	<b>vtstatus</b> This bit will be used in conjunction with the ready bit to determine the stage of the translation See table in section 0 register.					
	30:28	Reserved					
	27:24	<b>PAT entry</b> Location of Physical Address in Phys	sical Address	Table.			
	23:22	Reserved					
	21:20	Surface format					
		Value		Name			
		0xb	l	linear			
		10b	٦	Tile X			
		11b	1	Гile Y			
	19:14	Cache line offset in page					
	13:10	Cacheability Control Bits					
	9	ZLR bit indicates a zero length read					
	8:2	TAG					
	1:0	<b>SRC ID</b> 00/01=BCS; 10/11= BLB					



# BCS\_PSMI\_CTRL - BCS Sleep State and PSMI Control

		-	_				
Register	Space:	Ν	1MIO: 0/2/0				
Source:	Source: BlitterCS						
Default Value: 0x0000000							
Access:		R	/W				
Size (in b	bits):	3	2				
Trusted	Type:	1					
Address	:	2	2050h				
This regi	ister is	to be us	ed to control a	all aspects of PSMI and power saving	functi	ions	
DWord	Bit			Description			
0	31:16	Mask B	lits				
		Format	t:	Mask[15:0]			
		Must be	e set to modify	y corresponding bit in Bits 15:0. (All in	mplem	nented bits)	
	15	Reserve	ed				
		Format	t:		MBZ		
	14:13	Reserve	ed				
		Format	t:		MBZ		
	12	Reserve	ed				
		Format	t:		MBZ		
	11:8	Reserve	ed				
		Format	t:		MBZ		
	7	Reserved					
		Format	Format:			MBZ	
	6:5	Reserve	ed				
		Format:			MBZ		
	4	GO Ind	icator				
		Access	:			RO	
		Format:				GO	
		This is a read only field. Writing to this bit is undefined. To simplify power saving and					
		soft reset flows, the power management hardware has the ability to block all pending					
		memory cycles of the render pipe. When GO=0, all cycles are blocked. All CPD enter/exit					
				has this bit set to 0.			
		Value	Name		riptio		
		0h	Disable [ <b>Default]</b>	No cycles allowed coming out of cycles are complete. No new cyc context or PSMI cycles.			
		1h	Enable	Normal execution			



# BCS\_PSMI\_CTRL - BCS Sleep State and PSMI Control

3	IDLE Indicator				
	Default Value:	0h Render is assumed NOT IDLE coming out of reset			
	Access: RO				
	Format:	IDLE	IDLE		
	This is a read only field. Writing to this bit is undefined. This indicates what power management thinks what state the render pipe is in. That is, if set, the full handshake between render and power management has occured and most likely the render clocks are currently turned off.				
2	Reserved				
1	Reserved				
	Format: MBZ				
0	Reserved				



#### **BCS\_SWCTRL - BCS SW Control**

Register	Space:	MMIO: 0/2/0					
Source:		BlitterCS					
Default Value: 0x0000000							
Access: R/W							
Size (in b	oits):	32					
Trusted <sup>-</sup>	Туре:	1					
Address:		22200h					
DWord	Bit	Description					
0	31:16	Masks					
		Format:	U16				
		A "1" in a bit in this field allows the modification of the cor	responding bit in bits 15:0.				
-							
	15:4	Reserved					
		Format:	MBZ				
	3:2	Reserved					
		Format:	MBZ				
	1	Tile Y Destination					
		Format:	U1				
		Programming this bit makes the HW treat all destination surfaces as Tile Y. This bit over-rides the					
		setting of the destination format in the packet provided to the blitter command streamer. SW is					
		required to flush the HW before changing the polarity of this bit. This bit is part of the context					
		save/restore.					
	0	Tile Y Source					
		Format: U1					
		Programming this bit makes the HW treat all source surfaces as Tile Y. This bit over-rides the					
		setting of the source format in the packet provided to the blitter command streamer. SW is					
		required to flush the HW before changing the polarity of this bit. This bit is part of the context					
		save/restore.					



# **BCSTLB\_VA - BCS TLB Virtual Page Address Registers**

Register Space:	MMIO: 0/2,	MMIO: 0/2/0			
Source:	BlitterCS				
Default Value:	0x0000000	)			
Access:	RO				
Size (in bits):	32				
Trusted Type:	1				
Address:	24800h-248	303h			
This register is direc	tly mapped to	o the current Virtu	al Addresses in the BCS TLB.		
DWord	Bit		Description		
0	31:12	ADDRESS			
		Format:	GraphicsAddress[31:12]		
		PAGE VIRTUAL ADDRESS.			
	11:0 <b>RESERVED</b>				
		Format:		MBZ	



BCS_TLBPEND_VLD0 - BCS Valid Bit Vector for TLBPEND registers				
Register Space:	MMIO: 0/2	2/0		
Source:	BlitterCS			
Default Value:	0x000000	0		
Access:	RO			
Size (in bits):	32			
Trusted Type:	1			
Address:	Address: 24700h-24703h			
This register contains the valid bits for entries 0-31 of TLBPEND structure(Cycles pending TLB translation).				
DWord		Bit	Description	
0		31:0	Valid bits per entry	



#### BCS\_SYNC\_FLIP\_STATUS - BCS Wait for event and Display flip flags Register

Register Space:	MMIO: 0/2/0
Source:	BlitterCS
Default Value:	0x0000000
Access:	RO
Size (in bits):	32
Address:	222D0h

This register is the saved value of what wait for events are still valid. This register is part of context save and restore for RC6 feature.

DWord	Bit	t Description					
0	31	Reserved					
		Format:	MBZ				
	30	Display Plane A Asynchronous Display Flip Pending					
		Format:	Enable				
		This field enables a wait for the duration of a Display Plane A "Flip Pending" condition. If a flip request is pending, the parser will wait until the flip operation has completed (i.e., the new front buffer address has now been loaded into the active front buffer registers). See Display Flip Pending Condition (in the Device Programming Interface chapter of MI Functions.					
	29	Display Plane A Synchronous Flip Di	splay Pending				
		Format:	Enable				
		This field enables a wait for the duration of a Display Plane A "Flip Pending" condition. If a flip request is pending, the parser will wait until the flip operation has completed (i.e., the new front buffer address has now been loaded into the active front buffer registers). See Display Flip Pending Condition (in the Device Programming Interface chapter of MI Functions.					
	28	Display Sprite A Synchronous Flip Display Pending					
		Format:	Enable				
-		request is pending, the parser will wait buffer address has now been loaded ir	on of a Display Sprite A "Flip Pending" condition. If a flip until the flip operation has completed (i.e., the new front to the active front buffer registers). See Display Flip amming Interface chapter of MI Functions.				
	27	Reserved					
		Format:	MBZ				
	26	Display Plane B Asynchronous Displ	ay Flip Pending				
		Format:	Enable				
		This field enables a wait for the durati	on of a Display Plane B "Flip Pending" condition. If a flip				



BCS	5_S\	NC_FLIP_STATUS - BCS Wa flip flags Regis				
		request is pending, the parser will wait until the flip buffer address has now been loaded into the active Pending Condition (in the Device Programming Inte	e front buffer registers). See Display Flip			
	25	Display Plane B Synchronous Flip Display Pending				
		Format:	Enable			
		This field enables a wait for the duration of a Display Plane B "Flip Pending" condition. If a flip request is pending, the parser will wait until the flip operation has completed (i.e., the new front buffer address has now been loaded into the active front buffer registers). See Display Flip Pending Condition (in the Device Programming Interface chapter of MI Functions.				
	24	Display Sprite B Synchronous Flip Display Pendi	ng			
		Format:	Enable			
		This field enables a wait for the duration of a Displorequest is pending, the parser will wait until the flip buffer address has now been loaded into the active Pending Condition in the Device Programming Inte	operation has completed (i.e., the new front front buffer registers). See Display Flip			
	23	Reserved				
		Format:	MBZ			
	22	Display Plane A Asynchronous Flip Pending Wai	t Enable			
		Format:	Enable			
		This field enables a wait for the duration of a Displorequest is pending, the parser will wait until the flip buffer address has now been loaded into the active Pending Condition (in the Device Programming Inte	operation has completed (i.e., the new front front buffer registers). See Display Flip			
	21	Display Plane A Synchronous Flip Pending Wait	Enable			
		Format:	Enable			
		This field enables a wait for the duration of a Displa				
		request is pending, the parser will wait until the flip	• • •			
		buffer address has now been loaded into the active Pending Condition (in the Device Programming Inte				
	20	Display Sprite A Synchronous Flip Pending Wait	Enable			
		Format:	Enable			
		This field enables a wait for the duration of a Displorequest is pending, the parser will wait until the flip buffer address has now been loaded into the active Pending Condition in the Device Programming Interview	operation has completed (i.e., the new front front buffer registers). See Display Flip			



Instruction         MBZ           14         Display Plane B Asynchronous Flip Pending Wait Enable         Enable           14         Display Plane B Asynchronous Flip Pending Wait Enable         Enable           14         Display Plane B Asynchronous Flip Pending Wait Enable         Enable           15         Format:         Enable           16         Pending, the parser will wait until the flip operation has completed (i.e., the new free buffer address has now been loaded into the active front buffer registers). See Display Flip           13         Display Plane B Synchronous Flip Pending Wait Enable           Format:         Enable           This field enables a wait for the duration of a Display Plane B "Flip Pending" condition. If a flip           request is pending, the parser will wait until the flip operation has completed (i.e., the new free buffer address has now been loaded into the active front buffer registers). See Display Flip           Pending Condition (in the Device Programming Interface chapter of MI Functions.           12         Display Sprite B Synchronous Flip Pending Wait Enable           Format:         Enable           This field enables a wait for the duration of a Display Sprite B "Flip Pending" condition. If a flip request is pending, the parser will wait until the flip operation has completed (i.e., the new free buffer address has now been loaded into the active front buffer registers). See Display Flip Pending           Format:         Enable      <	19.15	Reserved					
14       Display Plane B Asynchronous Flip Pending Wait Enable         Format:       Enable         This field enables a wait for the duration of a Display Plane B "Flip Pending" condition. If a flip request is pending, the parser will wait until the flip operation has completed (i.e., the new from buffer address has now been loaded into the active front buffer registers). See Display Flip Pending Condition (in the Device Programming Interface chapter of MI Functions.         13       Display Plane B Synchronous Flip Pending Wait Enable         Format:       Enable         This field enables a wait for the duration of a Display Plane B "Flip Pending" condition. If a flip request is pending, the parser will wait until the flip operation has completed (i.e., the new from buffer address has now been loaded into the active front buffer registers). See Display Flip Pending Condition (in the Device Programming Interface chapter of MI Functions.         12       Display Sprite B Synchronous Flip Pending Wait Enable         Format:       Enable         This field enables a wait for the duration of a Display Sprite B "Flip Pending" condition. If a flip request is pending, the parser will wait until the flip operation has completed (i.e., the new from buffer address has now been loaded into the active front buffer registers). See Display Flip Pending         Pormat:       Enable         This field enables a wait for the duration of a Display Sprite B "Flip Pending" condition. If a flip request is pending, the parser will wait until the flip operation has completed (i.e., the new from buffer address has now been loaded into the active front buffer registers). See Display Fli	19.19		MBZ				
Format:       Enable         This field enables a wait for the duration of a Display Plane B "Flip Pending" condition. If a fli         request is pending, the parser will wait until the flip operation has completed (i.e., the new fre         buffer address has now been loaded into the active front buffer registers). See Display Flip         Pending Condition (in the Device Programming Interface chapter of MI Functions.         13       Display Plane B Synchronous Flip Pending Wait Enable         Format:       Enable         This field enables a wait for the duration of a Display Plane B "Flip Pending" condition. If a fli         request is pending, the parser will wait until the flip operation has completed (i.e., the new fre         buffer address has now been loaded into the active front buffer registers). See Display Flip         Pending Condition (in the Device Programming Interface chapter of MI Functions.         12       Display Sprite B Synchronous Flip Pending Wait Enable         Format:       Enable         This field enables a wait for the duration of a Display Sprite B "Flip Pending" condition. If a fli         request is pending, the parser will wait until the flip operation has completed (i.e., the new fre         buffer address has now been loaded into the active front buffer registers). See Display Flip         Pending Condition in the Device Programming Interface chapter of MI Functions.         11       Display Plane C Asynchronous Display Flip Pending	14		Flip Pending Wait Enable				
request is pending, the parser will wait until the flip operation has completed (i.e., the new from buffer address has now been loaded into the active front buffer registers). See Display Flip Pending Condition (in the Device Programming Interface chapter of MI Functions.         13       Display Plane B Synchronous Flip Pending Wait Enable         Format:       Enable         This field enables a wait for the duration of a Display Plane B "Flip Pending" condition. If a flip request is pending, the parser will wait until the flip operation has completed (i.e., the new from buffer address has now been loaded into the active front buffer registers). See Display Flip Pending Condition (in the Device Programming Interface chapter of MI Functions.         12       Display Sprite B Synchronous Flip Pending Wait Enable         Format:       Enable         This field enables a wait for the duration of a Display Sprite B "Flip Pending" condition. If a flip request is pending, the parser will wait until the flip operation has completed (i.e., the new from buffer address has now been loaded into the active front buffer registers). See Display Flip Pending Condition in the Device Programming Interface chapter of MI Functions.         11       Display Plane C Asynchronous Display Flip Pending         Format:       Enable         This field enables a wait for the duration of a Display Plane C "Flip Pending" condition. If a flip request is pending, the parser will wait until the flip operation has completed (i.e., the new from buffer address has now been loaded into the active front buffer registers). See Display Flip Pending Condition (in the Device Programming Interface chapter of MI Functions.							
Format:       Enable         This field enables a wait for the duration of a Display Plane B "Flip Pending" condition. If a flir request is pending, the parser will wait until the flip operation has completed (i.e., the new from buffer address has now been loaded into the active front buffer registers). See Display Flip Pending Condition (in the Device Programming Interface chapter of MI Functions.         12       Display Sprite B Synchronous Flip Pending Wait Enable         Format:       Enable         This field enables a wait for the duration of a Display Sprite B "Flip Pending" condition. If a flir request is pending, the parser will wait until the flip operation has completed (i.e., the new from buffer address has now been loaded into the active front buffer registers). See Display Flip Pending Condition in the Device Programming Interface chapter of MI Functions.         11       Display Plane C Asynchronous Display Flip Pending         Format:       Enable         This field enables a wait for the duration of a Display Plane C "Flip Pending" condition. If a flir request is pending, the parser will wait until the flip operation has completed (i.e., the new from buffer address has now been loaded into the active front buffer registers). See Display Flip Pending Condition. If a flir request is pending, the parser will wait until the flip operation has completed (i.e., the new from buffer address has now been loaded into the active front buffer registers). See Display Flip Pending Condition (in the Device Programming Interface chapter of MI Functions.         10       Display Plane C Synchronous Display Flip Pending Format:       Enable         This field enables a wait for the duratio		request is pending, the parser wil buffer address has now been load	l wait until the flip operation has completed (i.e., the new frod ded into the active front buffer registers). See Display Flip				
Format:       Enable         This field enables a wait for the duration of a Display Plane B "Flip Pending" condition. If a fli       request is pending, the parser will wait until the flip operation has completed (i.e., the new fro         buffer address has now been loaded into the active front buffer registers). See Display Flip         Pending Condition (in the Device Programming Interface chapter of MI Functions.         12       Display Sprite B Synchronous Flip Pending Wait Enable         Format:       Enable         This field enables a wait for the duration of a Display Sprite B "Flip Pending" condition. If a fli         request is pending, the parser will wait until the flip operation has completed (i.e., the new fro         buffer address has now been loaded into the active front buffer registers). See Display Flip         Pending Condition in the Device Programming Interface chapter of MI Functions.         11       Display Plane C Asynchronous Display Flip Pending         Format:       Enable         This field enables a wait for the duration of a Display Plane C "Flip Pending" condition. If a fli         request is pending, the parser will wait until the flip operation has completed (i.e., the new fro         buffer address has now been loaded into the active front buffer registers). See Display Flip         Pending Condition (in the Device Programming Interface chapter of MI Functions.         10       Display Plane C Synchronous Display Flip Pending         Format:	13	Display Plane B Synchronous F	ip Pending Wait Enable				
request is pending, the parser will wait until the flip operation has completed (i.e., the new from buffer address has now been loaded into the active front buffer registers). See Display Flip Pending Condition (in the Device Programming Interface chapter of MI Functions.         12       Display Sprite B Synchronous Flip Pending Wait Enable         Format:       Enable         This field enables a wait for the duration of a Display Sprite B "Flip Pending" condition. If a fli request is pending, the parser will wait until the flip operation has completed (i.e., the new from buffer address has now been loaded into the active front buffer registers). See Display Flip Pending Condition in the Device Programming Interface chapter of MI Functions.         11       Display Plane C Asynchronous Display Flip Pending         Format:       Enable         This field enables a wait for the duration of a Display Plane C "Flip Pending" condition. If a flir request is pending, the parser will wait until the flip operation has completed (i.e., the new from buffer address has now been loaded into the active front buffer registers). See Display Flip Pending Condition (in the Device Programming Interface chapter of MI Functions.         10       Display Plane C Synchronous Display Flip Pending [Format:       Enable         This field enables a wait for the duration of a Display Plane C "Flip Pending" condition. If a flir request is pending, the parser will wait until the flip operation has completed (i.e., the new from buffer address has now been loaded into the active front buffer registers). See Display Flip Pending [Format:         10       Display Plane C Synchronous Display Flip Pending [Format:							
Format:       Enable         This field enables a wait for the duration of a Display Sprite B "Flip Pending" condition. If a flirequest is pending, the parser will wait until the flip operation has completed (i.e., the new from buffer address has now been loaded into the active front buffer registers). See Display Flip Pending Condition in the Device Programming Interface chapter of MI Functions.         11 <b>Display Plane C Asynchronous Display Flip Pending</b> Format:       Enable         This field enables a wait for the duration of a Display Plane C "Flip Pending" condition. If a flir request is pending, the parser will wait until the flip operation has completed (i.e., the new from buffer address has now been loaded into the active front buffer registers). See Display Flip Pending Condition (in the Device Programming Interface chapter of MI Functions.         10 <b>Display Plane C Synchronous Display Flip Pending</b> Format:       Enable         11 <b>Display Plane C Synchronous Display Flip Pending</b> 12 <b>Display Plane C Synchronous Display Flip Pending</b> 13 <b>Display Plane C Synchronous Display Flip Pending</b> 14 <b>Display Plane C Synchronous Display Flip Pending</b> 15       Format:       Enable         16 <b>Display Plane C Synchronous Display Flip Pending</b> Format:         17       Enable       This field enables a wait for the duration of a Display Plane C "Flip Pending" condition. If a flir request is pending, the parser will wait un		request is pending, the parser wil buffer address has now been load	l wait until the flip operation has completed (i.e., the new fro ded into the active front buffer registers). See Display Flip				
Prival       Prival         This field enables a wait for the duration of a Display Sprite B "Flip Pending" condition. If a flirequest is pending, the parser will wait until the flip operation has completed (i.e., the new from buffer address has now been loaded into the active front buffer registers). See Display Flip Pending Condition in the Device Programming Interface chapter of MI Functions.         11       Display Plane C Asynchronous Display Flip Pending         Format:       Enable         This field enables a wait for the duration of a Display Plane C "Flip Pending" condition. If a flir request is pending, the parser will wait until the flip operation has completed (i.e., the new from buffer address has now been loaded into the active front buffer registers). See Display Flip Pending Condition (in the Device Programming Interface chapter of MI Functions.         10       Display Plane C Synchronous Display Flip Pending         Format:       Enable         11       This field enables a wait for the duration of a Display Plane C "Flip Pending" condition. If a flir request is pending, the parser will wait until the flip operation has completed (i.e., the new from buffer address has now been loaded into the active front buffer registers). See Display Flip Pending         10       Display Plane C Synchronous Display Flip Pending         11       Format:       Enable         12       This field enables a wait for the duration of a Display Plane C "Flip Pending" condition. If a flir request is pending, the parser will wait until the flip operation has completed (i.e., the new from buffer address has now been loaded into the active fron	12	Display Sprite B Synchronous F	lip Pending Wait Enable				
request is pending, the parser will wait until the flip operation has completed (i.e., the new from buffer address has now been loaded into the active front buffer registers). See Display Flip Pending Condition in the Device Programming Interface chapter of MI Functions.         11       Display Plane C Asynchronous Display Flip Pending         Format:       Enable         This field enables a wait for the duration of a Display Plane C "Flip Pending" condition. If a flir request is pending, the parser will wait until the flip operation has completed (i.e., the new from buffer address has now been loaded into the active front buffer registers). See Display Flip Pending Condition (in the Device Programming Interface chapter of MI Functions.         10       Display Plane C Synchronous Display Flip Pending         Format:       Enable         This field enables a wait for the duration of a Display Plane C "Flip Pending" condition. If a flir request is pending, the parser will wait until the flip operation has completed (i.e., the new from buffer address has now been loaded into the active front buffer registers). See Display Flip Pending         10       Display Plane C Synchronous Display Flip Pending         Image: Format:       Enable         This field enables a wait for the duration of a Display Plane C "Flip Pending" condition. If a flir request is pending, the parser will wait until the flip operation has completed (i.e., the new from buffer address has now been loaded into the active front buffer registers). See Display Flip Pending Condition (in the Device Programming Interface chapter of MI Functions.         9       Display Sprite C Synchronous Flip Displa		Format:	Enable				
Format:       Enable         This field enables a wait for the duration of a Display Plane C "Flip Pending" condition. If a flir request is pending, the parser will wait until the flip operation has completed (i.e., the new from buffer address has now been loaded into the active front buffer registers). See Display Flip Pending Condition (in the Device Programming Interface chapter of MI Functions.         10       Display Plane C Synchronous Display Flip Pending         Format:       Enable         This field enables a wait for the duration of a Display Plane C "Flip Pending" condition. If a flip request is pending, the parser will wait until the flip operation has completed (i.e., the new from buffer address has now been loaded into the active front buffer registers). See Display Flip Pending         9       Display Sprite C Synchronous Flip Display Pending		request is pending, the parser wil buffer address has now been load	l wait until the flip operation has completed (i.e., the new fro ded into the active front buffer registers). See Display Flip				
<ul> <li>This field enables a wait for the duration of a Display Plane C "Flip Pending" condition. If a flip request is pending, the parser will wait until the flip operation has completed (i.e., the new from buffer address has now been loaded into the active front buffer registers). See Display Flip Pending Condition (in the Device Programming Interface chapter of MI Functions.</li> <li>Display Plane C Synchronous Display Flip Pending         <ul> <li>Format:</li> <li>Enable</li> <li>This field enables a wait for the duration of a Display Plane C "Flip Pending" condition. If a flip request is pending, the parser will wait until the flip operation has completed (i.e., the new from buffer address has now been loaded into the active front buffer registers). See Display Flip Pending</li> </ul> </li> <li>9 Display Sprite C Synchronous Flip Display Pending</li> </ul>	11	Display Plane C Asynchronous	Display Flip Pending				
<ul> <li>request is pending, the parser will wait until the flip operation has completed (i.e., the new from buffer address has now been loaded into the active front buffer registers). See Display Flip Pending Condition (in the Device Programming Interface chapter of MI Functions.</li> <li>Display Plane C Synchronous Display Flip Pending         <ul> <li>Format:</li> <li>Enable</li> <li>This field enables a wait for the duration of a Display Plane C "Flip Pending" condition. If a flip request is pending, the parser will wait until the flip operation has completed (i.e., the new from buffer address has now been loaded into the active front buffer registers). See Display Flip Pending Condition (in the Device Programming Interface chapter of MI Functions.</li> </ul> </li> <li>Display Sprite C Synchronous Flip Display Pending</li> </ul>		Format:	Enable				
Format:       Enable         This field enables a wait for the duration of a Display Plane C "Flip Pending" condition. If a flip request is pending, the parser will wait until the flip operation has completed (i.e., the new from buffer address has now been loaded into the active front buffer registers). See Display Flip Pending Condition (in the Device Programming Interface chapter of MI Functions.         9       Display Sprite C Synchronous Flip Display Pending		request is pending, the parser wil buffer address has now been load	l wait until the flip operation has completed (i.e., the new fro ded into the active front buffer registers). See Display Flip				
This field enables a wait for the duration of a Display Plane C "Flip Pending" condition. If a fli request is pending, the parser will wait until the flip operation has completed (i.e., the new fro buffer address has now been loaded into the active front buffer registers). See Display Flip Pending Condition (in the Device Programming Interface chapter of MI Functions. 9 <b>Display Sprite C Synchronous Flip Display Pending</b>	10	Display Plane C Synchronous D	isplay Flip Pending				
<ul> <li>request is pending, the parser will wait until the flip operation has completed (i.e., the new from buffer address has now been loaded into the active front buffer registers). See Display Flip Pending Condition (in the Device Programming Interface chapter of MI Functions.</li> <li>Display Sprite C Synchronous Flip Display Pending</li> </ul>		Format:	Enable				
		request is pending, the parser wil buffer address has now been load	l wait until the flip operation has completed (i.e., the new fro ded into the active front buffer registers). See Display Flip				
	9	Display Sprite C Synchronous F	lip Display Pending				
		request is pending, the parser wi	I wait until the flip operation has completed (i.e., the new fro				



			w been loaded into the active front buffer registers). See Display Flip the Device Programming Interface chapter of MI Functions.		
8	Display Pl	lane C Async	chronous Flip Pending Wait Enable		
	Format:		Enable		
	request is buffer add	pending, the Iress has now	ait for the duration of a Display Plane C "Flip Pending" condition. If a f e parser will wait until the flip operation has completed (i.e., the new fr w been loaded into the active front buffer registers). See Display Flip the Device Programming Interface chapter of MI Functions.		
7	Display Pl	lane C Synch	hronous Flip Pending Wait Enable		
	Format:	-	Enable		
	buffer add	lress has now	e parser will wait until the flip operation has completed (i.e., the new fr w been loaded into the active front buffer registers). See Display Flip the Device Programming Interface chapter of MI Functions.		
6	Display Sp	prite C Synch	hronous Flip Pending Wait Enable		
	Format:		Enable		
	This field enables a wait for the duration of a Display Sprite C "Flip Pending" condition. If a flip request is pending, the parser will wait until the flip operation has completed (i.e., the new from buffer address has now been loaded into the active front buffer registers). See Display Flip Pending Condition in the Device Programming Interface chapter of MI Functions.				
			• · · · ·		
5			the Device Programming Interface chapter of MI Functions.		
5	Pending C		• · · · ·		
5 4:0	Pending Control Reserved Format: Condition This field of enable selection	<b>Code Wait</b> enables a wai ect one of 15 tion-code in t	the Device Programming Interface chapter of MI Functions		
	Pending Control Reserved Format: Condition This field of enable selection	ondition in the condition is the condition of the conditi	the Device Programming Interface chapter of MI Functions MBZ Select ait for the duration that the corresponding condition code is active. The 5 condition codes in the EXCC register, that cause the parser to wait u		
	Pending Control Reserved Format: Condition This field of enable select that condition Value	<b>Code Wait</b> enables a wai ect one of 15 tion-code in t	MBZ  Select ait for the duration that the corresponding condition code is active. Th condition codes in the EXCC register, that cause the parser to wait u the EXCC is cleared.  Description		
	Pending Control Reserved Format: Condition This field of enable select that condition Value Oh	<b>Code Wait S</b> enables a wai ect one of 15 tion-code in t Name	MBZ         Select         ait for the duration that the corresponding condition code is active. The secondition codes in the EXCC register, that cause the parser to wait une the EXCC is cleared.         Description		
	Pending Condition         Format:         Condition         This field of enable selection         that condition         Value         Oh         1h-5h	<b>Code Wait S</b> enables a wai ect one of 15 tion-code in t Name Not Enabled	Image: Select       MBZ         Scondition codes in the corresponding condition code is active. The scondition codes in the EXCC register, that cause the parser to wait up the EXCC is cleared.         Image: Description         Condition Code Wait not enabled		





# BCS\_CTR\_THRSH - BCS Watchdog Counter Threshold

Register Space:		e: MMIO: 0/2/0							
Source:		BlitterCS							
Default Value:		0x00150000							
Access:		R/W							
Size (in b	oits):	32							
Address:		2217Ch							
DWord	Bit		Description						
0	21.0	Country to all Thursday I.d.	er logic Threshold						
	51.0	Counter logic Threshold							
	51.0	Default Value:	00150000h						
	51.0		00150000h U32						



# MFC\_BITSTREAM\_SE\_BITCOUNT\_SLICE - Bitstream Output Bit Count for the last Syntax Element Report Register

Register Space	ce:	MMIO: 0/2/0				
Source:		VideoCS				
Default Value:		0x0000000				
Access:		RO				
Size (in bits):		32				
Trusted Type	:	1				
Address:		124D4h				
Name:		VDBOX1				
bit count is b	efore the	ne count of number of bits in the bitstream for the last syntax element before padding. The e byte-aligned alignment padding insertion, but includes the stop-one-bit. This register is ve and restore.				
DWord Bit		Description				
0	31:0	<b>MFC Bitstream Syntax Element Bit Count</b> Total number of bits in the bitstream output before padding. This count is updated each time the internal counter is incremented.				



#### MFC\_BITSTREAM\_BYTECOUNT\_SLICE - Bitstream Output Byte Count Per Slice Report Register

Register Space:		MMIO: 0/2/0			
Source:		VideoCS			
Default Value:		0x0000000			
Access:		RO			
Size (in bits	5):	32			
Trusted Typ	be:	1			
Address:		124D0h			
This regist	er store	es the count of bytes of the bitstream output. This register is part of the context save and			
restore.					
DWord	Bit	Description			
0	31:0	MFC Bitstream Byte Count			
		Total number of bytes in the bitstream output from the encoder. This count is updated for			
		every time the internal bitstream counter is incremented.			



#### MFC\_AVC\_MINSIZE\_PADDING\_COUNT - Bitstream Output Minimal Size Padding Count Report Register

Register Sp	bace:	MMIO: 0/2/0		
Source:		VideoCS		
Default Value:		0x0000000		
Access:		RO		
Size (in bit	s):	32		
Trusted Ty	pe:	1		
Address:		12414h		
Name:		VDBOX1		
5		es the count in bytes of <b>minimal size padding insertion. It is primarily provided for</b> <b>athering.</b> This register is part of the context save and restore.		
DWord	Bit	Description		
0	31:0	MFC AVC MinSize Padding Count		
		Total number of bytes in the bitstream output contributing to minimal size padding operation. This count is updated each time when the padding count is incremented.		



# **BLBTLB\_VA - BLBTLB\_VA Virtual page Address Registers**

Register Space:	MMIO: 0/2/	0			
Source:	BlitterCS				
Default Value:	0x0000000	)			
Access:	RO				
Size (in bits):	32				
Trusted Type:	1				
Address:	24900h-249	903h			
This register is dire	ectly mapped	d to the current	Virtual Addresses in the BLB	TLB.	
DWord	Bit		Description		
0	31:12	ADDRESS			
		Format:	GraphicsAddress[31:12]		
		PAGE VIRTUAI	_ ADDRESS		
	11:0	RESERVED	RESERVED		
		Format:		MBZ	



#### **BLT\_ENG\_FR - Blitter Engine Fault Register**

Register Space:MMIO: 0/2/0Default Value:0x00000000Size (in bits):32

Address: 04294h-04297h

Blitter Engine Fault Register

Word	Bit	Description				
0	31:12	Virtual Address of Page Fault				
		Default Value:	00000h			
		Access: R/W				
		This is the original Address of the Page that gene	erated the First	fault for this engine.		
		This value is locked and not updated on subsequent faults, until the valid bit of this register is cleared by SW				
	11	Blitter GTTSEL				
		Default Value:		0b		
		Access:	R/W			
				19.00		
		This bit indicates if the valid bit happened while u GGTT	using PPGTT or			
		This bit indicates if the valid bit happened while u	5	GGTT: 0 - PPGTT, 1 -		
	10:3	This bit indicates if the valid bit happened while u GGTT This value is locked and not updated on subsequ	5	GGTT: 0 - PPGTT, 1 -		
	10:3	This bit indicates if the valid bit happened while u GGTT This value is locked and not updated on subsequ register is cleared by SW	5	GGTT: 0 - PPGTT, 1 -		
	10:3	This bit indicates if the valid bit happened while u GGTT This value is locked and not updated on subsequ register is cleared by SW SRCID of Fault	5	GGTT: 0 - PPGTT, 1 - the valid bit of this		
	10:3	This bit indicates if the valid bit happened while u GGTT This value is locked and not updated on subsequ register is cleared by SW SRCID of Fault Default Value:	ent faults, until	GGTT: 0 - PPGTT, 1 - the valid bit of this 00h R/W		
	10:3	This bit indicates if the valid bit happened while u GGTT This value is locked and not updated on subsequ register is cleared by SW SRCID of Fault Default Value: Access: This is the Source ID of the unit that requested th	ent faults, until	GGTT: 0 - PPGTT, 1 - the valid bit of this 00h R/W nerated the First Page		
	2:1	This bit indicates if the valid bit happened while u GGTT This value is locked and not updated on subseque register is cleared by SW SRCID of Fault Default Value: Access: This is the Source ID of the unit that requested the fault for this engine. This value is locked and not updated on subseque	ent faults, until	GGTT: 0 - PPGTT, 1 - the valid bit of this 00h R/W nerated the First Page		



	Access:	R/W			
	Type of Fault recorded:				
	00 - Page Fault.				
	01 - Invalid PD Fault				
	10 - Unloaded PD Fault				
	11 - Invalid and Unloaded PD fault				
	11 - Invalid and Unloaded PD fault				
	<ul><li>11 - Invalid and Unloaded PD fault</li><li>This value is locked and not updated on subsequent fa</li><li>register is cleared by SW</li></ul>	ults, until the valid bit of th			
0	This value is locked and not updated on subsequent fa	ults, until the valid bit of th			
0	This value is locked and not updated on subsequent faregister is cleared by SW	ults, until the valid bit of th			
0	This value is locked and not updated on subsequent faregister is cleared by SW           Valid Bit				



# **BLT\_MODE - Blitter Mode Register**

Register Space:	MMIO: 0/2/0	
Source:	BlitterCS	
Default Value:	0x0000000	
Access:	R/W	
Size (in bits):	32	
Trusted Type:	1	
Address:	2229Ch	

DWord	Bit				[	Description		
0	31:16	Mask B	Bits					
		Forma	t:		N	Mask[15:0]		
		Must b	e set to m	odify o	corresponding bit in	Bits 15:0. (All in	nplemented bits)	
	15:14	Reserv	ed					
		Forma	t:				MBZ	
	13:10	Reserv	ed					
		Forma	t:				MBZ	
	9	Per-Pro	ocess GTT	Enab	le			
		Forma	t:	Enabl	e Per-Process GTT BS Mode Enable			
		Value	Name		Description			
		0h	0h PPGTT Disable [Default]		When clear, the Global GTT will be used to translate memory access from designated commands and for commands that select the PPGTT as their translation space.			
		1h	PPGTT Er	nable		ommands and f	to translate memory access or commands that select the	
	8	Reserv	/ed					
	7:4	Reserved						
		Format: MBZ						
	3:1	Reserved						
		Forma	t:				MBZ	
	0	Reserv	ed					
		Forma	t:				MBZ	



BRS	<b>BRSYNC - Blitter/Render Semaphore Sync Register</b>							
Register Space:	MM	IO: 0/2/0						
Source:	Blitte	erCS						
Default Value:	0x00	000000						
Access:	R/W							
Size (in bits):	32							
Trusted Type:	1							
Address:	2204	40h						
This register is	written b	y CS, read by BCS.						
DWord	Bit	Description						
0	31:0	Semaphore Data						
		Semaphore data for synchronization between blitter engine and render engine.						

Г



٦

	<b>BVSYNC - Blitter/Video Semaphore Sync Register</b>							
Register S	pace:	MMIO: 0/2/0						
Source:		BlitterCS						
Default Va	alue:	0x0000000						
Access:		R/W						
Size (in bi	ts):	32						
Trusted T	/pe:	1						
Address:		22044h						
This regis	ter is v	written by VCS, read by BCS.						
DWord	Bit	Description						
0	0 31:0 <b>Semaphore Data</b> Semaphore data for synchronization between blitter engine and video codec engine.							



Γ

		CAC	HE_N	NO	DE_0 - Cacl	he Mod	e Re	gister 0
Register	Space:	M	MIO: 0/2,	/0				
Source:		Re	nderCS					
Default \	/alue:	0x0	0000004	4				
Access:		R/	W					
Size (in b	oits):	32						
Address:		07	000h					
	Description							
reserve Before	d bits chang	are impl ing the v	emented alue of	d as r this r	operation of the Re ead/write. egister, GFX pipelin as part of Context.	ie must be id		
-	-	MMIO_S				,		
DWord	Bit					Description		
0	31:16	Masks						
		Format:			M	lask[15:0]		
		A 1 in a b	oit in this	field	allows the modificati	on of the corre	espondin	g bit in Bits 15:0.
	15	Sampler	L2 Disal	ble				
		Format:				Disable		
		Value	Nam	e		Des	cription	
		0h	[Defau	lt]	Sampler L2 Cache En			
		1h	-	Sampler L2 Cache Disabled. All accesses are treated as misses.				
	14:12	Reserved						
		Format: MBZ						
	11	Reserved						
		Format:					MBZ	
	10	Reserved	d					
		Format:					MBZ	
	9	Sampler	L2 TLB	Prefe	tch Enable	Γ		
Value					Name		De	escription
		0h		[Def	ault]	TLB Prefetch I	Disabled	
		1h				TLB Prefetch I	Enabled	
	8	Reserve	d					
	7:6	Sampler	L2 Requ	iest A	rbitration			
		Format:						U2



# CACHE\_MODE\_0 - Cache Mode Register 0

		1.							
		Valu	ie N	lame	Description				
		00b			Round Robin				
		01b			Fetch are Highest Priority				
		10b			Constants are Highest Priority				
		11b	11b Reserved						
-	5	STC Eviction Policy							
		Format: Disable							
		this bit is		cates that	ave LRA as replacement policy. The default value i.e. (when non-LRA eviction policy. This bit must be reset. LRA ported.				
				1	Programming Notes				
		If this bit	t is set to "1"	, bit 4 of 0>	7010h must also be set to "1".				
-	4	RCC Evic	tion Policy						
		Format:			Disable				
		If this bit is set, RCCunit will have LRA as replacement policy. The default value i.e. (when this bit is reset) indicates that non-LRA eviction policy. This bit must be reset. LRA replacement policy is not supported.							
		Programming Notes							
		If this bit is set to "1", bit 7 of 0x7010h must also be set to "1".							
	3	Reserved							
	2	Hierarchical Z RAW Stall Optimization Disable							
		Format: U1							
		The Hierarchical Z RAW Stall Optimization allows non-overlapping polygons in the same 8x4 pixel/sample area to be processed without stalling waiting for the earlier ones to write to Hierarchical Z buffer.							
		Value	Nar	ne	Description				
		0h	Enable		Enables the hierarchical Z RAW Stall Optimization.				
		1h	Disable <b>[De</b>	fault]	Disables the hierarchical Z RAW Stall Optimization.				
-	1	Disable o	lock gating	in the pix	el backend				
		Format:			Disable				
		MCL rela	ated clock g	ating is di	sabled in the pixel backend. Before setting this bit to 1, the				
		instruction/state caches must be invalidated.							
	0	Render C	Cache Opera	tional Flus	sh Enable				
		Format:			Enable				
			N						
		Value	Name		Description				
		Oh D	Disable	Operation	hal Flush Disabled (recommended for performance when not				

Т



C	CACH	E_MOD	E_0 - (	Cache	Mode	Register	0	
1	1	l I						

	[Default]	rendering to the front buffer). Restriction: This bit must be set to '0' (Disable).
1h		Operational Flush Enabled (required when rendering to the front buffer). Restriction: Do not use this value; this bit must be 0.



#### **CACHE\_MODE\_1** - Cache Mode Register 1 **Register Space:** MMIO: 0/2/0 Source: RenderCS Default Value: 0x00000180 Access: Read/32 bit Write Only Size (in bits): 32 Address: 07004h **Description** RegisterType: MMIO\_SVL Before changing the value of this register, GFX pipeline must be idle; i.e., full flush is required. This Register is saved and restored as part of Context. DWord Bit Description 0 31:16 Mask Bits for 15:0 Format: Mask[15:0] Must be set to modify corresponding data bit. Reads to this field returns zero. 15 Reserved Format: MBZ Reserved 14 Format: MBZ Reserved 13 12 **HIZ Eviction Policy** U1 Format: If this bit is set, Hizunit will have LRA as replacement policy. The default value i.e. (when this bit is reset) indicates the non-LRA eviction policy. For performance reasons, this bit must be reset. Value Name Description 0h [Default] Non-LRA eviction Policy LRA eviction Policy 1h **Programming Notes** If this bit is set to "1", bit 3 of 0x7010h must also be set to "1" 11 Reserved Format: MBZ 10 Reserved 9 Reserved 8:7 Sampler Cache Set XOR selection Format: U2 These bits have an impact only when the Sampler cache is configured in 16 way set associative



# CACHE\_MODE\_1 - Cache Mode Register 1

Value	Name	Description
00b	None	No XOR.
01b	Scheme 1	New_set_mask[3:0] = Tiled_address[16:13].
		New_set[3:0] less than or = New_set_mask[3:0] ^Old_set[3:0].
		Rationale: These bits can distinguish among 16 different equivalent classes of virtual pages. These bits also represen the lsb for tile rows ranging from a pitch of 1 tile to 16 tiles
10b	Scheme 2	New_set_mask[3] = Tiled_address[17] ^ Tiled_address[16].
		New_set_mask[2] = Tiled_address[16] ^ Tiled_address[15].
		New_set_mask[1] = Tiled_address[15] ^ Tiled_address[14].
		New_set_mask[0] = Tiled_address[14] ^ Tiled_address[13].
		New_set[3:0] less than or = New_set_mask[3:0] ^ Old_set[3:0].
		Rationale: More bits on each XOR can give better statistica uniformity on sets and since two lsbs are taken for each til row size, it reduces the chance of aliasing on sets.
11b	Scheme 3 [ <b>Default]</b>	New_set_mask[3] = Tiled_address[22] ^ Tiled_address[21] Tiled_address[20] ^ Tiled_address[19].
		New_set_mask[2] = Tiled_address[18] ^ Tiled_address[17] Tiled_address[16].
		New_set_mask[1] = Tiled_address[15] ^ Tiled_address[14].



		CA	CHE_N	IODE_1 - Cache Mode Register 1					
				New_set_mask[0] = Tiled_address[13].					
				New_set[3:0] less than or = New_set_mask[3:0] ^ Old_set[3:0].					
				Rationale: More bits on each XOR can give better statistical uniformity on sets and since each XOR has different bits, it reduces the chance of aliasing on sets even more.					
	6	Pixel B	ackend sub	o-span collection Optimization Disable					
		Format	t:	Disable					
		Value	Name	Description					
		0h	[Default]	Enables two contiguous quads to be collected as 4X2 access for RCZ interface. This allows for less bank collision and less RAM power on RCZ.					
		1h		Disables this optimization and therefore only one valid sub-span is sent to RCZ on the 4X2 interface.					
		Programming Notes							
	_	This bit must be set.							
	5	MCS Cache Disable Format: Disable							
		For Programming restrictions please refer to the 3D Pipeline.							
		Value	Name	Description					
		0h	[Default]	MCS cache enabled. It allows RTs with MCS buffer enabled to be rendered using either MSAA compression for MSRT OR with color clear feature for non MSRT.					
		1h		MCS cache is disabled. Hence no MSAA compression for MSRT and no color clear for non-MSRT.					
	4	Reserve	ed						
		Format	t:	MBZ					
	3	Depth	Read Hit W	/rite-Only Optimization Disable					
		Format	t:	Disable					
				Description					
		This bi	t must alwa	ys be reset to "0".					
		Value	Name	Description					
		0h		Read Hit Write-only optimization is enabled in the Depth cache					



# CACHE\_MODE\_1 - Cache Mode Register 1

[Default] (RCZ).						
	1h		Read Hit Write-only optimization is dis (RCZ).	abled in the Depth cache		
2	Reserv	Reserved				
1	Reserved					
	Forma	Format: MBZ				
0	Reserv	ed				



#### CASC\_LRA\_0 - CASC LRA 0

Register Space:	MMIC	D: 0/2/0						
Default Value:	0x1F1	100F00						
Size (in bits):	32							
Address:	04060	)h-04063h						
CASC LRA 0								
DWord	Bit		Description					
0	31:24	CASC LRA1 Max						
		Default Value:	00011111b					
		Access:	R/W					
		CASC LRA1 Max	I					
		Format: U6						
		Maximum value of programmable LRA1						
		Maximum Allow Value: 159						
	23:16	CASC LRA1 Min						
		Default Value:	00010000b					
		Access:	R/W					
		CASC LRA1 Min						
		Format: U6						
		Minimum value of programmable LRA1						
	15:8	CASC LRA0 Max						
		Default Value:	00001111b					
		Access:	R/W					
		CASC LRA0 Max						
		Format: U6						
		Maximum value of programmable LRA0						
		Maximum Allow Value: 159						
	7:0	CASC LRA0 Min						
		Default Value:	0000000b					
		Access:	R/W					
		CASC LRA0 Min						



Г

CASC_LRA_0 - CASC LRA 0				
	Format: U6			
	Minimum value of programmable LRA1			



#### CASC\_LRA\_1 - CASC LRA 1

		·····						
Register Space:	MMIC	D: 0/2/0						
Default Value:	0x3F3	02F20						
Size (in bits):	32							
Address:	04064	łh-04067h						
CASC LRA 1								
DWord	Bit		Description					
0	31:24	CASC LRA3 Max						
		Default Value:	0011111b					
		Access:	R/W					
		CASC LRA3 Max						
		Format: U6						
		Maximum value of programmable LRA3						
	23:16	CASC LRA3 Min						
		Default Value:	00110000b					
		Access:	R/W					
		CASC LRA3 Min						
		Format: U6						
		Minimum value of programmable LRA3						
	15:8	CASC LRA2 Max						
		Default Value:	00101111b					
		Access:	R/W					
		CASC LRA2 Max						
		Format: U6						
-		Maximum value of programmable LRA2						
	7:0	CASC LRA2 Min						
		Default Value:	0010000b					
		Access:	R/W					
		CASC LRA2 Min						
		Format: U6						
		Minimum value of program	mable LRA2					





		CASC_LRA_2 - CASC LF	RA 2	
Register Space: Default Value: Size (in bits):	MMIO: 0/2/0 0x00009F40 32			
Address:	04068h-0406Bh			
CASC LRA 2				
DWord	Bit	Description		
0	31:16	Reserved		
		Default Value:	0000h	
		Access:	RO	
		Reserved Format: MBZ		
	15:8	CASC LRA4 Max		
	15.8	Default Value:	10011111b	
		Access:	R/W	
		CASC LRA4 Max	Γ./ ¥¥	
		Format: U6 Maximum value of programmable LRA4 Maximum Allow Value: 159		
	7:0	CASC LRA4 Min		
	7.0	Default Value:	0100000b	
		Access:	R/W	
		CASC LRA4 Min		
		Format: U6		
		Minimum value of programmable LRA4		



### CASC\_LRA\_3 - CASC LRA 3

Register Space:	MMIO: 0/2/0					
Default Value:	0x000014	E4				
size (in bits):	32					
Address:	0406Ch-0	406Fh				
CASC LRA 3						
DWord	Bit		Description			
0	31:18	Reserved				
		Default Value:	00000000000000000000000000000000000000			
		Access:	RO			
		Reserved				
		Format: MBZ				
	17:15	BCS LRA				
		Default Value:	000b			
		Access:	R/W			
		BCS LRA				
		Format: U6				
		Which LRA should use				
	14:12	BLB LRA				
		Default Value:	001b			
		Access: R/W				
		BLB LRA				
		Format: U6				
		Which LRA should use				
	11:9	VCS LRA				
		Default Value:	010b			
		Access: R/W				
		VCS LRA				
		Format: U6				
		Which LRA should use				
	8:6	VMX LRA				
		Default Value:	011b			



### CASC\_LRA\_3 - CASC LRA 3

	Access:	R/W
	VMX LRA	
	Format: U6	
	Which LRA should use	
5:3	VMC LRA	
	Default Value:	100b
	Access:	R/W
	VMC LRA	
	Format: U6	
	Which LRA should use	
2:0	VCR LRA	
	Default Value:	100b
	Access:	R/W
	VCR LRA	
	Format: U6	
	Which LRA should use	



# **CL\_INVOCATION\_COUNT** - **Clipper Invocation Counter**

Register	Space	MMIO: 0/2/0			
Source:		RenderCS			
Default \	/alue:	0x0000000, 0x0000000			
Access:		R/W			
Size (in b	oits):	64			
Trusted <sup>-</sup>	Туре:	1			
Address:	Address: 02338h				
This reg restore.	his register stores the count of objects entering the Clipper stage. This register is part of the context save and estore.				
DWord	Bit	Description			
0	63:0	CL Invocation Count Report			
		Number of objects entering the clipper stage. Updated only when Statistics Enable is set in CLIP_STATE (see the Clipper Chapter in the 3D Volume.)			



# **CL\_PRIMITIVES\_COUNT - Clipper Primitives Counter**

Register	Space	MMIO: 0/2/0				
Source:		RenderCS				
Default \	/alue:	0x0000000, 0x0000000				
Access:		R/W				
Size (in b	oits):	64				
Trusted <sup>-</sup>	Type:	1				
Address:		02340h				
5		reflects the total number of primitives that have been output by the clipper. This register is part of ve and restore.				
DWord	Bit	Description				
0	63:0	oped Primitives Output Count				
		otal number of primitives output by the clipper stage. This count is updated for every primitive utput by the clipper stage, as long as Statistics Enable is set in SF_STATE (see the Clipper and SF hapters in the 3D Volume.)				



# **CGMSG - Clock Gating Messages**

Register	Space:	MMIO: 0/2/0				
Default V	/alue:	0x0000000				
Size (in b	its):	32				
Address:		08104h				
Clock Ga	ting M	essages Register				
DWord	Bit	Description				
0	31:16	Message Mask				
		Access:	RO			
		Message Mask				
		In order to write to bits 15:0, the corresponding message mask bits must be written.				
		For example, for bit 14 to be set, bit 30 needs to be 3	1: 40004000			
	15:2	Reserved				
		Access:	RO			
		Reserved	·			
	1	Media 1 Clock gating control message				
		Access:	R/W			
		Gate Media 1 (2nd Vbox) Clock Message :				
		'0': Media 1 Clock Un-gate Request (un-gates the cmclk clock in the 2n Media block)				
		'1': Media 1 Clock Gate Request (gates the cmclk clockin the 2nd Media block)				
	0	Row Clock Gating Control Message				
		Access:	R/W			
		Gate Row Clocks Message :				
		'0': Row Clock Un-gate Request (un-gates the crclk a				
		'1': Row Clock Gate Request (gates the crclk and cr2>	(clk clocks)			



# **CZWMRK - Color/Depth Write FIFO Watermarks**

Register	Space	MMIO: 0/2/0				
Source: RenderCS		RenderCS				
Default V						
Access:		R/W				
Size (in b	oits):	32				
Trusted	Туре:	1				
Address	:	04060h				
This regi	ster is	directly mapped to the current Virtual Addresses in	n the MTTLB (Texture and constant cache TLB).			
DWord	Bit	Descri	ption			
0	31:24	Reserved				
		Format:	MBZ			
	23:18	<b>Color Wr Burst Size</b> This is the maximum size of the requests burst, from the last High Watermark trip, before reevaluating the High Watermark again.				
	17:16	Reserved				
		Format:	MBZ			
	15:12	<b>Color Wr FIFO High Watermark</b> This is the number of accumulated Color writes that will trigger a Burst of Z Writes.				
	11:6	<b>Z Wr Burst Size</b> This is the maximum size of the requests burst, fro reevaluating the High Watermark again.	om the last High Watermark trip, before			
	5:4	Reserved				
		Format:	MBZ			
	3:0	<b>Z Wr FIFO High Watermark</b> This is the number of accumulated Depth writes t	hat will trigger a Burst of Z Writes.			



### TS\_GPGPU\_THREADS\_DISPATCHED - Count Active Channels Dispatched

Register Space: MMIO: 0/2/0					
Source: RenderCS					
Default Value: 0x0000000, 0x0000000					
Access: R/W					
Size (in bits)	):	64			
Trusted Typ	e:	1			
Address:	Address: 02290h				
active bits in	This register is used to count the number of active channels that TS sends for dispatch. For each dispatch the active bits in the execution mask are summed and added to this register. This register is reset when a write occurs to 2290h				
DWord	Bit	Description			
0	63:0	GPGPU_THREADS_DISPATCHED			
		Format:	U64		
		This count is increased by the number of active bits in the execution mask each time the TS sends a GPGPU dispatch.			



Register Space:	MMIO: 0/2/0
Source:	RenderCS
Default Value:	0x0000000
Access:	RO
Size (in bits):	32
Address:	022ACh

This register contains the state code of the Power Management FSM, FBC Forward FSM, CSSTDT FSM, CSARB FSM, CSBUPDATE FSM. Decoding the contents of this register will indicate what the state of the corresponding state machine.

DWord	Bit	Description					
0	31:30	Reserved					
		Format:			MBZ		
	29:28	CSFBCSLICE0					
		Format:				U2	
		FBC message forward FSM state					
		Value			Name		
		Oh		CSFBCIDLE_0			
		1h		CSFBCMODIFY_0			
		2h		CSFBCCLEAN_0			
		3h		CSFBCDONE_0			
	27:24	Reserved			1		
		Format:		MBZ			
	23:21	CS ARB				1	
		Format:				U3	
			mmar	nd streamer. Describes what st	ate CS is	in	
		Value		Name			
		0h		BIDLE_s		_	
		1h	-	RNG_s CS		_	
		2h	-	BATCH_s		_	
		3h	_	ВСНК		_	
		4h	-	BCHK1		_	
		5h		XOP_s		_	
		6h-7h	Res	served			
	20	Reserved					
		Format:			MBZ		
	19:17	CSSWITCH			T	1	
		Format:				U3	



	Value	Name			
	0h	SWIDLE_s			
	1h	SWITCH_s			
	2h	ASREQ_s			
	3h	DMACHK_s			
	4h	ARBWAIT_s			
	5h	FIFORECFG_s			
	6h-7h	Reserved			
16:13	CSCSBUPDATE				
	Format:	U4			
	CS Power Management CS	BLOCK FSM state			
	Value	Name			
	0h	CSBIDLE			
	1h	CSQ			
	2h	WRPTR			
	3h	SEMA1			
	4h	SEMA2			
	5h	TS1			
	6h	TS2			
	7h	TS3			
	8h	TS4			
	9h	DUMMYREQ			
	Ah	DUMMYWT			
	Bh	INTWT			
	Ch-Fh	Reserved			
12:11	R2MWRREQ				
	Format:	U2			
	CSSTDT memory request F	SM state			
	Value	Name			
	0h	WRIDLE			
	1h	WRREQ_HW1			
	2h	WRREQ_HW2			
	3h	WRRD			
10	Reserved				



9:7	LOADARB				_
	Format:	Format:			U3
	CSSTDT arbiter FSM state				
	Value				Name
	0h			LDIDLE	
	1h			LDAUTO	
	2h			LDPRSR	
	3h			LDCTX	
	4h			LDFLSH	
	5h			LDREG	
	6h			LDSHR1	
	7h			Reserved	
			Dre	aramming Notos	
		oc poods 4		ogramming Notes	its have been manned on
		M is the sta			its have been mapped on bits mapped, LDLRM/LDID
6:4					
0.1	Format:				U3
	CS Power Management CSBLOCK FSM state				
	Value			Name	
	0h	CSBLO	СК		
	1h	CSCTX	ARB		
	2h	CSUNE	BLOCKRES	TODE	
				TORE	
	3h	CSUNE		TORE	
	3h 4h				
			BLOCK P4BLOCK		
3:0	4h	CSPRE	BLOCK P4BLOCK	IORE	
3:0	4h 5h-7h	CSPRE	BLOCK P4BLOCK		U4
3:0	4h 5h-7h CSIDLE	CSPRE Reserv	BLOCK P4BLOCK red		U4
3:0	4h 5h-7h <b>CSIDLE</b> Format:	CSPRE Reserv	BLOCK P4BLOCK red		1
3:0	4h 5h-7h <b>CSIDLE</b> Format: CS Power Managem	CSPRE Reserv	BLOCK P4BLOCK red	ate	1
3:0	4h 5h-7h CSIDLE Format: CS Power Managem Value	CSPRE Reserv	BLOCK P4BLOCK /ed PCK FSM st	ate	1
3:0	4h 5h-7h <b>CSIDLE</b> Format: CS Power Managem Value Oh	CSPRE Reserv	BLOCK P4BLOCK red OCK FSM st SBUSY	ate	1
3:0	4h 5h-7h CSIDLE Format: CS Power Managem Value Oh 1h	CSPRE Reserv	BLOCK P4BLOCK /ed PCK FSM st SBUSY NTWT	ate	1
3:0	4h 5h-7h CSIDLE Format: CS Power Managem Value Oh 1h 2h	CSPRE Reserv	BLOCK P4BLOCK red OCK FSM st SBUSY NTWT LSHREQ	ate	1



	6h	PMTURNOFF
	7h	PMIDLEWT
	8h	IDLE
	9h	PMTURNON
	Ah	PMBUSYWT
	Bh	DOPFFCGREQ
	Ch	DOPFFCGWAIT
	Dh	DOPFFCG
	Eh	DOPFFCUGREQ
	Fh	DOPFFCUGWAIT



# **CTXTLB\_VA - CTXTLB\_VA Virtual page Address Registers**

Register Space:	MMIO: 0/2/	MMIO: 0/2/0				
Source:	BlitterCS					
Default Value:	0x0000000	1				
Access:	RO					
Size (in bits):	32					
Trusted Type:	1					
Address:	24A00h-24A	A03h				
This register is dir	ectly mapped	I to the current	Virtual Addresses in the CTX	(TLB.		
DWord	Bit		Description			
0	31:12	ADDRESS				
		Format:	GraphicsAddress[31:12]			
		PAGE VIRTUAL	_ ADDRESS			
	11:0 RESERVED					
		Format:		MBZ		



# **CCID - Current Context Register**

Register S	Space:	MMIO: 0/2/0							
Source:		RenderCS							
Default V	alue:	0x0000000							
Access:		R/W							
Size (in bi	bits): 32								
Address:		02180h							
		ains the current logical rendering context address associated with the ring buffer in ring buffer ng. This register contents are not valid in Exec-List mode of scheduling.							
		Programming Notes							
(i.e., the l	Ring But	r must not be written directly (via MMIO) unless the Command Streamer is completely idle fer is empty and the pipeline is idle). Note that, under normal conditions, the CCID register pdated from the command stream using the MI_SET_CONTEXT command.							
DWord	Bit	Description							
0	31:12	Logical Render Context Address (LRCA)							
		Default Value: 0h							
		Format: GraphicsAddress[31:12]							
		This field contains the 4 KB-aligned Graphics Memory Address of the current Logical Rendering Context. Bit 11 MBZ.							
	11:10	Reserved							
		Format: MBZ							
	9	Reserved							
	8	Reserved							
	8	Reserved       Format:     Must Be One							
	8								
		Format: Must Be One							
		Format: Must Be One Reserved							
	7:4	Format:     Must Be One       Reserved       Format:     MBZ							
	7:4	Format:     Must Be One       Reserved       Format:     MBZ       Extended State Save Enable							
	7:4	Format:       Must Be One         Reserved       MBZ         Format:       MBZ         Extended State Save Enable       Enable         Format:       Enable         If set, the extended state identified in the Logical Context Data section of the Memory Data							
	7:4	Format:       Must Be One         Reserved       MBZ         Format:       MBZ         Extended State Save Enable       Enable         Format:       Enable         If set, the extended state identified in the Logical Context Data section of the Memory Data Formats chapter, is saved as part of switching away from this logical context.							
	7:4	Format:       Must Be One         Reserved       MBZ         Format:       MBZ         Extended State Save Enable       Enable         Format:       Enable         If set, the extended state identified in the Logical Context Data section of the Memory Data Formats chapter, is saved as part of switching away from this logical context.         Extended State Restore Enable							
	7:4	Must Be One         Reserved         Format:       MBZ         Extended State Save Enable         Format:       Enable         If set, the extended state identified in the Logical Context Data section of the Memory Data         Formats chapter, is saved as part of switching away from this logical context.         Extended State Restore Enable         Format:       Enable         If set, the extended state identified in the Logical Context Data section of the Memory Data         Format:       Enable         If set, the extended state identified in the Logical Context Data section of the Memory Data							



0	Valid	Valid					
	Format	t:		U1			
	Value	Value Name Description					
	0h	Invalid [ <b>Default]</b>	The other fields of this register are invalid. context will not invoke a context save oper	3			
1h Valid	Valid	The other fields of this register are valid, an will invoke the normal context save/restore					



#### **CEC0-0 - Customizable Event Creation 0-0**

Register	Space:	Ν	/MIO: 0/2/0	)					
Source:		В	Spec						
Default \	/alue:	0	x00000000						
Access:		R	x/W						
Size (in b	oits):	3	2						
Address:		0	2390h						
-			define custe Event Cour		er event 0, bit definitions in this register refer to the CEC block on.	(			
DWord	Bit				Description				
0	31:21	Reserv	ed						
		Forma	t:		MBZ				
	20:19	Source	Select						
		Forma			U2				
			Selects the input signals to the Boolean event definition logic (see block diagram in the Custom						
			ounters sec	tion).	Description				
		Value	Name	<b>D</b>	Description				
		00b	Reserved	Reserved					
		01b	Prev Event		e conditioned/flopped input from the last CEC block as bus to CEC0 block				
		10b	Reserved	Reserved					
		11b	Reserved						
	18:3	Compare Value							
		Forma	t:		U16				
		the co comp function	omparator arison that on is true,	(see block is done i then the s	ompared the 16-bit conditioned input bus that are fed in k diagram in the Custom Event Counters section). The ty s controlled by the Compare Function. When the compa signal for the custom event is asserted. This signal in tur formance counter or fed into other CEC blocks.	vpe of are			
	2:0	Compa	re Functio	n					
		Forma	t:		U3				
		to the v			used by the CEC0 comparator when comparing the compare C0 conditioned input bus (see block diagram in the Custom E				
		Value	Na	me	Description				
		000b	Any Are Ec	qual	Compare and assert output if any are equal (Can be used as function)	OR			



#### **CEC0-0 - Customizable Event Creation 0-0**

	001b	Greater Than	Compare and assert output if greater than	
	010b	Equal	Compare and assert output if equal to (Can also be used as AND function)	
	011b	Greater Than or Equal	Compare and assert output if greater than or equal	
	100b	Less Than	Compare and assert output if less than	
	101b	Not Equal	Compare and assert output if not equal	
	110b	Less Than or Equal	Compare and assert output if less than or equal	
	111b	Reserved		



#### **CEC1-0 - Customizable Event Creation 1-0**

Register	Space:	Ν	1MIO: 0/2/0	)					
Source:		В	Spec						
Default \	/alue:	0	x00000000						
Access:		R	/W						
Size (in b	oits):	3	2						
Address:		0	2398h						
-			define custo Event Cour		r event 1, bit definitions in on.	this regis	ter refer to the CEC blo	ock	
DWord	Bit	Description							
0	31:21	Reserve	ed						
		Format	t:			MBZ			
	20:19	Source	Select				1		
		Format	t:				U2		
			the input si ounters sec		ne Boolean event definition	logic (see	e block diagram in the	Custom	
		Value	Name		Descript	ion		_	
		00b	Reserved	Reserved					
		01b	Prev Event		e conditioned/flopped inpu out bus to this CEC block	it from th	e previous CEC block		
		10b	Reserved	Reserved					
		11b	Reserved						
	18:3	Compare Value							
		Format	t:			U16	5		
		the co compa functio	mparator arison that on is true,	(see block is done is then the s	ompared the 16-bit cond < diagram in the Custom s controlled by the Comp signal for the custom eve formance counter or fed	Event Co bare Fund nt is ass	ounters section). The ction. When the com erted. This signal in t	type of pare	
	2:0	Compa	re Functio	n					
		Format					U3		
		to the v			used by the CEC comparat C conditioned input bus (se				
		Value	Nai	me		Descrip	tion		
		000b	Any Are Ec	qual	Compare and assert outpu function)	t if any a	re equal (Can be used	as OR	
		001b	Greater Th	an	Compare and assert output	t if great	er than		



#### **CEC1-0 - Customizable Event Creation 1-0**

	010b	Equal	Compare and assert output if equal to (Can also be used as AND function)
	011b	Greater Than or Equal	Compare and assert output if greater than or equal
	100b	Less Than	Compare and assert output if less than
	101b	Not Equal	Compare and assert output if not equal
	110b	Less Than or Equal	Compare and assert output if less than or equal
	111b	Reserved	



#### **CEC2-0 - Customizable Event Creation 2-0**

Register	Space:	Ν	1MIO: 0/2/0	)					
Source:		В	Spec						
Default \	/alue:	0	x00000000						
Access:		R	/W						
Size (in b	oits):	3	2						
Address:		0	23A0h						
0			define custo Event Cour		er event 2, bit definitions in this register refer to the CEC bloc on.	k			
DWord	Bit				Description				
0	31:21	Reserve	ed						
		Format	t:		MBZ				
	20:19	Source	Select						
		Format	t:		U2				
			Selects the input signals to the Boolean event definition logic (see block diagram in the Custom Event Counters section).						
		Value	Name		Description				
		00b	Reserved	Reserved					
		01b	Prev Event		e conditioned/flopped input from the previous CEC block but bus to this CEC block	ck			
		10b	Reserved	Reserved					
		11b	Reserved						
	18:3	Compa	re Value						
	10.5	Compare Value       Format:     U16							
		the co compa functio	mparator arison that on is true,	(see blocl is done i then the s	ompared the 16-bit conditioned input bus that are fed k diagram in the Custom Event Counters section). The t s controlled by the Compare Function. When the comp signal for the custom event is asserted. This signal in tu formance counter or fed into other CEC blocks.	ype of are			
	2:0	Compa	re Functio	n					
		Format			U3				
		to the v			used by the CEC comparator when comparing the compare C conditioned input bus (see block diagram in the Custom E				
		Value	Na	me	Description				
		000b	Any Are Ec	qual	Compare and assert output if any are equal (Can be used as function)	s OR			



#### **CEC2-0 - Customizable Event Creation 2-0**

	001b	Greater Than	Compare and assert output if greater than
	010b	Equal	Compare and assert output if equal to (Can also be used as AND function)
	011b	Greater Than or Equal	Compare and assert output if greater than or equal
	100b	Less Than	Compare and assert output if less than
	101b	Not Equal	Compare and assert output if not equal
	110b	Less Than or Equal	Compare and assert output if less than or equal
	111b	Reserved	



#### **CEC3-0 - Customizable Event Creation 3-0**

Register	Space:	Ν	/MIO: 0/2/0	)				
Source:		В	Spec					
Default \	/alue:	0	x00000000					
Access:		R	./W					
Size (in b	oits):	3	32					
Address:		0	023A8h					
5			define custo Event Cour		r event 3, bit definitions in this register refer to the CEC block on.			
DWord	Bit				Description			
0	31:21	Reserv	ed					
		Forma	t:		MBZ			
	20:19	Source	Select					
		Forma	t:		U2			
			Selects the input signals to the Boolean event definition logic (see block diagram in the Custom Event Counters section).					
		Value	Name		Description			
		00b	Reserved	Reserved				
		01b	Prev Event		elects the conditioned/flopped input from the previous CEC block as the nput bus to this CEC block			
		10b	Reserved	Reserved				
		11b	Reserved					
	18:3	Compare Value						
	-010	Forma			U16			
		the co compa functio	omparator arison that on is true,	(see block is done i then the s	ompared the 16-bit conditioned input bus that are fed into c diagram in the Custom Event Counters section). The type of s controlled by the Compare Function. When the compare signal for the custom event is asserted. This signal in turn can formance counter or fed into other CEC blocks.			
	2:0	Compare Function Format: U3						
		to the v			used by the CEC comparator when comparing the compare value C conditioned input bus (see block diagram in the Custom Event			
		Value	Nai	ne	Description			
		000b	Any Are Ec	jual	Compare and assert output if any are equal (Can be used as OR function)			



#### **CEC3-0 - Customizable Event Creation 3-0**

	001b	Greater Than	Compare and assert output if greater than	
	010b	Equal	Compare and assert output if equal to (Can also be used as AND function)	
	011b	Greater Than or Equal	Compare and assert output if greater than or equal	
	100b	Less Than	Compare and assert output if less than	
	101b	Not Equal	Compare and assert output if not equal	
	110b	Less Than or Equal	Compare and assert output if less than or equal	
	111b	Reserved		

Look Inside."

#### CVS\_TLB\_LRA\_0 - CVS TLB LRA 0

Register Space:	MMI	D: 0/2/0	
Default Value:	0x1F0	080700	
Size (in bits):	32		
Address:	04044	4h-04047h	
CVS TLB LRA 0			
DWord	Bit		Description
0	31:29	Reserved	
		Default Value:	000b
		Access:	RO
		Reserved	
		Format: MBZ	
Γ	28:24	CVS LRA1 Max	
		Default Value:	11111b
		Access:	R/W
		CVS LRA1 Max	
		Format: MBZ	
		Maximum value of programmat	ble LRA1
	23:21	Reserved	
		Default Value:	000b
		Access:	RO
		Reserved	
		Format: MBZ	
	20:16	CVS LRA1 Min	
		Default Value:	01000b
		Access:	R/W
		CVS LRA1 Min	
		Format: U6	
		Minimum value of programmable LRA1	
	15:13	Reserved	
		Default Value:	000b
		Access:	RO
		Reserved	



### CVS\_TLB\_LRA\_0 - CVS TLB LRA 0

		Format: MBZ				
	12:8	CVS LRA0 Max				
		Default Value:	0011	l1b		
		Access:	R/W			
		CVS LRA0 Max				
		Format: MBZ				
		Maximum value of programmable LRA0				
-	7:5	Reserved				
		Default Value:		000b		
		Access:		RO		
		Reserved				
		Format: MBZ				
	4:0	CVS LRA0 Min				
		Default Value:	0000	00b		
		Access:	R/W			
		CVS LRA0 Min				
		Format: U6				
		Minimum value of programmable LRA0				

### CVS\_TLB\_LRA\_1 - CVS TLB LRA 1

Register Space:	MMIC	D: 0/2/0				
Default Value:		)01F18				
Size (in bits):	32					
Address:						
	04048	3h-0404Bh				
CVS TLB LRA 1						
DWord	Bit			Description		
0	31:13	Reserved				
		Default Value	:	000000000000000000000000000000000000000	00b	
		Access:		RO		
		Reserved				
_		Format:	MBZ			
	12:8	CVS LRA2 Ma				
		Default Value	:		111111	b
		Access:			R/W	
		CVS LRA2 Ma	ах			
		Format:	MBZ			
		Maximum value of programmable LRA2				
	7:5	Reserved				
	7.5	Default Value	·		(	000b
		Access:	•			RO
		Reserved				
		Format:	MBZ			
	4:0	CVS LRA2 Mi	n			
		Default Value	:		11000	b
		Access:			R/W	
		CVS LRA2 Mi	in			
		Format:	U6			
		Minimum va	lue of program	mable LRA2		



		CVS_T	LB_LR	A_2 - CVS TLB LRA 2					
Register Space: Default Value: Size (in bits):		MMIO: 0/2/0 0x00000005 32							
Address:		0404Ch-0404Fl	h						
CVS TLB LR	A 2								
DWord	Bit		Description						
0	31:6	Reserved							
		Default Value:		00000000000000000000000000000000000000					
		Access:		RO					
		Reserved							
		Format:	MBZ						
	5:4	CS LRA							
		Default Value	:		00b				
		Access:			R/W				
		CS LRA Format: Which LRA sh	U6 nould CS use	e					
	3:2	VF LRA							
		Default Value	:		01b				
		Access:			R/W				
		VF LRA							
		Format:	U1	_					
	1.0	Which LRA should VF use							
	1:0	SO LRA Default Value			01b				
			•		R/W				
		Access: SO LRA			R/ W				
		Format: MBZ							
		Which LRA sh	nould SO us	e					



# **ZSHR - Depth/Early Depth TLB Partitioning Register**

Register Space: MMIO: 0/2/0						
Source:		RenderCS				
Default Value	:	0x0000020				
Access:		R/W				
Size (in bits):		32				
Trusted Type:		1				
Address:	dress: 04050h					
J		o determine the number of TLB entries from the total of TLB. The rest of the entries are used for the Early Depth	5			
DWord	Bit	Description				
0	31:6	Reserved				
		Format:	MBZ			
	5:0	Number of TLB Entries Out of 64 used for Depth TLB				
		Default Value:	32			
		The rest are be used for Early Depth/Stencil TLB. Default value is 32.				



#### **DS\_INVOCATION\_COUNT - DS Invocation Counter**

oace:	MMIO: 0/2/0		
	RenderCS		
lue:	0x0000000, 0x0000000		
	R/W		
s):	64		
pe:	1		
	02308h		
This register stores the number of domain points shaded by the DS threads. Domain points which hit in the I cache will not cause this register to increment. Note that the spawning of a DS thread which shades two dom points will cause this counter to increment by two. This register is part of the context save and restore.			
Bit	Description		
63:0	<b>DS Invocation Count</b> Number of domain points shaded by the DS threads. Updated only when DS Function Enable and Statistics Enable are set in 3DSTATE DS		
	lue: s): pe: not cau cause t <b>Bit</b>		



#### **EIR - Error Identity Register**

Register Space:	MMIO: 0/2/0
Source:	RenderCS
Default Value:	0x0000000
Access:	R/W,RO
Size (in bits):	32
Address:	020B0h

The EIR register contains the persistent values of Hardware-Detected Error Condition bits. Any bit set in this register will cause the Master Error bit in the ISR to be set. The EIR register is also used by software to clear detected errors (by writing a 1 to the appropriate bit(s)), except for the unrecoverable bits described.)

DWord	Bit	Description							
0	31:16	Reserved	Reserved						
		Format:	Format: MBZ						
	15:0	Error Ide	Error Identity Bits						
		Format:	Array of Error condit	ion bits See the table titled	Hardware-Detected Error Bits.				
		EMR regis bits in this clear an e in this fiel	his register contains the persistent values of ESR error status bits that are unmasked via the IR register. (See Table Table 3-3. Hardware-Detected Error Bits). The logical OR of all (defined) is in this register is reported in the Master Error bit of the Interrupt Status Register. In order to ear an error condition, software must first clear the error by writing a 1 to the appropriate bit(s) this field. If required, software should then proceed to clear the Master Error bit of the IIR. served bits are RO.						
			Value		Name				
1h Error occurred									
		Programming Notes							
		0	or bit (Bit 4) nor the I		e cleared. However, neither the Page an be cleared except by reset (i.e., it is				



EMR - Error Mask Register							
Register Space: MMIO: 0/2/0							
Source:		RenderCS					
Default Valu	ie:	0x000000FI	:				
Access:		R/W,RO					
Size (in bits)	:	32					
Address:		020B4h					
Unmasked bits will b interrupt, and will pe		be reported ir ersist in the El	used by software to control which Error Status Register bits are masked or unmasked. e reported in the EIR, thus setting the Master Error ISR bit and possibly triggering a CPU rsist in the EIR until cleared by software. Masked bits will not be reported in the EIR and merate Master Error conditions or CPU interrupts. Reserved bits are RO.				
DWord	Bit			Description			
0	31:8	Reserved					
		Format:		Must Be One			
		Programming		Programming Notes			
		These bits a	re not implemented in H	IW and must be set to '1'			
	7:0	Error Mask I	Bits				
		Format: Array of error condition mask bits See the table titled Hardware-Detected Erro Bits.					
		5	This register contains a bit mask that selects which error condition bits (from the ESR) are eported in the EIR.				
		Value	Name	Description			
		FFh	[Default]				
		0h	Not Masked	Will be reported in the EIR			
		1h	Masked	Will not be reported in the EIR			



#### **ESR - Error Status Register**

Register Space:	MMIO: 0/2/0
Source:	RenderCS
Default Value:	0x0000000
Access:	RO
Size (in bits):	32
Address:	020B8h

The ESR register contains the current values of all Hardware-Detected Error condition bits (these are all by definition persistent). The EMR register selects which of these error conditions are reported in the persistent EIR (i.e., set bits must be cleared by software) and thereby causing a Master Error interrupt condition to be reported in the ISR.

DWord	Bit	Description					
0	31:16	Reserved	Reserved				
		Format:	Format: MBZ				
	15:0	Error Status Bits	Error Status Bits				
		Format: Array of error condition bits See the table titled Hardware-Detected Error Bits.					
		This register contains the non-persistent values of all hardware-detected error condition					
		bits.					
		Value Name					
		1h	1h Error Condition Detected				



### **TD\_PM\_MODE\_EUCOUNT - EU Mask Programming**

Register Space:	MMIO: 0/2/0
Source:	BSpec
Default Value:	0x00000000
Access:	WO
Size (in bits):	32

DWord	Bit		Description	
0	31:24	Reserved		
		Format:	MBZ	
	23	SubSlice 2 EU 7 Enable		
		Format:	Enable	
	22	SubSlice 2 EU 6 Enable		
		Format:	Enable	
	21	SubSlice 2 EU 5 Enable		
		Format:	Enable	
	20	SubSlice 2 EU 4 Enable		
		Format:	Enable	
	19	SubSlice 2 EU 3 Enable		
		Format:	Enable	
	18	SubSlice 2 EU 2 Enable		
		Format:	Enable	
	17	SubSlice 2 EU 1 Enable		
		Format:	Enable	
	16	SubSlice 2 EU 0 Enable		
		Format:	Enable	
	15	SubSlice 1 EU 7 Enable		
		Format:	Enable	
	14	SubSlice 1 EU 6 Enable		
		Format:	Enable	
	13	SubSlice 1 EU 5 Enable		
		Format:	Enable	
	12	SubSlice 1 EU 4 Enable		
		Format:	Enable	
	11	SubSlice 1 EU 3 Enable		
		Format:	Enable	
	10	SubSlice 1 EU 2 Enable		



# TD\_PM\_MODE\_EUCOUNT - EU Mask Programming

Format:Enable9SubSlice 1 EU 1 EnableFormat:Enable8SubSlice 1 EU 0 EnableFormat:Enable7SubSlice 0 EU 7 Enable7SubSlice 0 EU 7 Enable6SubSlice 0 EU 6 Enable6SubSlice 0 EU 6 Enable5SubSlice 0 EU 5 Enable5SubSlice 0 EU 5 Enable6SubSlice 0 EU 4 Enable
Format:Enable8SubSlice 1 EU 0 EnableFormat:Enable7SubSlice 0 EU 7 EnableFormat:Enable6SubSlice 0 EU 6 EnableFormat:Enable5SubSlice 0 EU 5 EnableFormat:Enable
8       SubSlice 1 EU 0 Enable         Format:       Enable         7       SubSlice 0 EU 7 Enable         Format:       Enable         6       SubSlice 0 EU 6 Enable         Format:       Enable         5       SubSlice 0 EU 5 Enable         Format:       Enable
Format:Enable7SubSlice 0 EU 7 EnableFormat:Enable6SubSlice 0 EU 6 EnableFormat:Enable5SubSlice 0 EU 5 EnableFormat:Enable
SubSlice 0 EU 7 Enable         Format:       Enable         6       SubSlice 0 EU 6 Enable         Format:       Enable         5       SubSlice 0 EU 5 Enable         Format:       Enable
Format:     Enable       6     SubSlice 0 EU 6 Enable       Format:     Enable       5     SubSlice 0 EU 5 Enable       Format:     Enable
6     SubSlice 0 EU 6 Enable       Format:     Enable       5     SubSlice 0 EU 5 Enable       Format:     Enable
Format:     Enable       5     SubSlice 0 EU 5 Enable       Format:     Enable
5 SubSlice 0 EU 5 Enable Format: Enable
Format: Enable
4 SubSlice 0 EU 4 Enable
Format: Enable
3 SubSlice 0 EU 3 Enable
Format: Enable
2 SubSlice 0 EU 2 Enable
Format: Enable
1 SubSlice 0 EU 1 Enable
Format: Enable
0 SubSlice 0 EU 0 Enable
Format: Enable



Register S	Space:	MMIO: 0/2/0			
Source:		RenderCS			
Default Value:		0x0000000			
Access:		R/W,RO			
Size (in bi	ts):	32			
Trusted T	ype:	1			
Address:		02028h			
MI_WAIT from arb evaluates evaluates	FOR_E itration to a 0 to a 0	ntains user defined and hardware generated conditions that are used by EVENT commands. An MI_WAIT_FOR_EVENT instruction excludes the executing ring in if the selected event evaluates to a 1, while instruction is discarded if the condition 0. Once excluded a ring is enabled into arbitration when the selected condition 0. 10.			
DWord	Bit	Description			
0 31:16 Mask Bits Format: Mask[15:0] These bits serves as a write enable for bits 15:0. If this register is written with any of the clear the corresponding bit in the field 15:0 will not be modified. Reading these bits returns 0s.					
	Reserved				
		Format: MBZ			
	11	Pending Indirect State Dirty Bit			
		Access:       RO         This field keeps track of whether or not an indirect state pointer command has been parsed in the current context. Clears either on a context save or explicitly through a flush command. This bit is Read Only.			
	10:7	<b>Pending Indirect State Counter</b> This field keeps track of the maximum number of indirect state pointers pending in the system. When the register is saved/restored, it saves either a value of 1 or 0. This field is Read-Only.			
	6:5	Reserved			
		Format: MBZ			
	4:0	<b>User Defined Condition Codes</b> The software may signal a Stream Semaphore by setting the Mask bit and Signal Bit together to match the bit field specified in a WAIT_FOR_EVENT (Semaphore).			



# **GAB\_ERR\_REPORT - GAB Error Reporting Register**

Register Space:	r Space: MMIO: 0/2/0			
Source:	Blitt	erCS		
Default Value:	0x00	000000		
Access:	RO			
Size (in bits):	32			
Trusted Type:	1			
Address:	2409	94h		
This register is directly mapped for the Error Reporting Register.				
DWord	Bit	Description		
0	31:8	Reserved		
	7	HWSP GGTT fetch yields an invalid entry		
	6	Reserved		
	5	Reserved		
	4	PD fetch yields an invalid entry		
	3	PD fetch for entry marked as invalid by BCS		
	2	GTT fetch yields an invalid entry		
		Page Fault occurred in one of the GTT translations.		
	1	Reserved		



### **GAB\_HWSP\_REG - GAB Hardware Status Page Address** Register

Register Space:	MMIO: 0/2/0
Default Value:	0x00000000
Size (in bits):	32
Address:	04280h-04283h

Address:

This register is used to program the 4 KB-aligned System Memory address of the Hardware Status Page used to report hardware status into (typically cacheable) System Memory. This address in this register is translated using the Global GTT in memory. The mapping type of the GTT entry determines the snoop nature of the transaction to memory.

DWord	Bit	Description	
0	31:12	GAB HWSP Register	
		Default Value: 00	000h
		Access: R/	W
	11:0	Reserved	
		Default Value:	000h
		Access:	RO



## **GAB\_CTL\_REG - GAB unit Control Register**

Register Space:			MMIO: 0/2/0							
BlitterC										
/alue:		0x000000	3F							
		R/W								
oits):		32								
		24000h								
alue=	FF0000	BFh Truste	d Type = 1							
Bit			Description							
31:9	Reserv	ed								
8	Contin	ue after P	age Fault							
	Value	Name	Description							
							1	GAB Set	Ipon receiving a page fault when requesting an addres set address bit 39 to 1 and continue.	s translation, GAB will
					0	GAB Hang	GAB will hang on a page fault. Default = b0.			
7:6	PPGTT	BCS TLB	LRA MIN							
	Default Value: 10b			10b						
	TLB Depth Partitioning Register In PP GTT Mode.									
5:4	GAB w	rite reque	st priority signal value used in GAC arbitration							
	Default Value: 11b			11b						
3:2	GAB re	ad only re	equest priority signal value used in GAC arbitration							
	Defaul									
1:0	GAB re	read request priority signal value used in GAC arbitration								
	Default Value: 11b									
	/alue: bits): alue= Bit 31:9 8 7:6 5:4 3:2	<pre>/alue: bits): alue=FF0000 Bit 31:9 Reserv 8 Contin Value 1 0 7:6 PPGTT 0efaul 7:6 PPGTT 0efaul 7:8 GAB w 0efaul 3:2 GAB re 0efaul 1:0 GAB re</pre>	Alue:       BlitterCS         /alue:       0x000000         R/W       R/W         oits):       32         alue=FF0000BFh Truste       24000h         alue:       FF0000BFh Truste         Bit	Image: Addition of the second seco						



## GAC\_ARB\_CTL\_REG - GAC\_GAB Arbitration Counters Register 1

Register	Space	MMIO: 0/2/0					
Source:		VideoCS					
Default Value:		0x00400002					
Access:		R/W					
Size (in l	bits):	32					
Trusted	Type:	1					
Address	:	14050h					
GAC_GA	B R/RC	)/W Arbitration Control Register					
DWord	Bit	Description					
0	31	GAC write request Limit Enable					
		Format:	U1				
		present and no GAB, let meda and vice versa). If both are present, start counting and switch when programmable no of request is expired. Allow only One GAB request and reset the counter. Counter only counts while we service a particular client and another client is present, else counter will reset.					
	30	VLF Final write Limit Enable					
		Format: MBZ					
		As long as there is no conflict Between VCS MFD and VLF Final V shows up (if VLF present and no VCSMFD, Let VLF and vice versa counting and when programmable no of request is expired. Allo And counter will reset. Counter only counts while we service a pr is present, else counter will reset.	a). If both are present, Start w only One VCSMFD request				
	29:24	Write Reg Limit Count					
		Format:	U6				
		The value programmed determines the number of GAC/VLF Wri	tes will allow for Each time.				
	23	GAC/GAB Cascaded Read Only Limit Enable					
		Format:	U1				
		As long as there is no conflict between GAC and GAB Read Requishows up (if GAC present and no GAB, Let GAC and vice versa). I counting and switch when programmable no of request from end counter when switch). Counter only counts while we service a part present, else counter will reset.	f both are present, Start her side is expired (reset the				
	22	shows up (if GAC present and no GAB, Let GAC and vice versa). I counting and switch when programmable no of request from eit counter when switch). Counter only counts while we service a pa	f both are present, Start her side is expired (reset the				



GAC_ARB_C	TL_REG - GAC	_GAB Arbitration Counters
	Regi	ster 1

	Once programmable counter is disabled, GAC uses the fixed arbitration setting given in this register setting.					
	Value	Name				
	0	GAC				
	1 GAB [Default]					
21	Reserved					
	Format:		MBZ			
20:16	GAC/GAB Read Only Limit Co	ounter Value				
	Format:		U5			
	This is the Maximum number c	of Read requests Allowed fro	om Each Cascaded Agent. Default 0			
15	GAC/GAB Cascaded Read Lin	nit Enable				
	Format:		U1			
	the counter when switch). Counter only counts while we service a particular client and other client is present, else counter will reset. Default 0					
14	Default priority 0-GAC, 1-GAB					
	Format:		MBZ			
	Default 0					
13	Reserved					
	Format:		MBZ			
12:8	GAC/GAB Read Limit Counter Value					
	Format: U5					
	This is the Maximum number of Read requests allowed from Each Cascaded Agent.					
7:6	Reserved					
	Format:		MBZ			
5:0	No of Global GTT Entries Valid in PPGTT mode in TLB064					
	Default Value:		000010b			
	Format:		U6			
	Minimum value the PPGGTT LRA can have (effectively partitioning the TLB between PPGTT and GGTT). Currently, only 2 entries are allocated to GGTT in ASmodel. TLB64 is shared by GGTT and PPGTT entries, are 2 LRAs, the GGTT one running from 0 up to PPGTT_MIN -1 (which is 2, but					



## GAC\_ARB\_CTL\_REG - GAC\_GAB Arbitration Counters Register 1

could be changed if needed), and the PPGTT one running from PPGTT\_MIN up to 63.



#### ARB\_GAC\_GAM\_REQCNTS0 - GAC\_GAM Arbitration Counters Register 0

Register Space:		MMIO: 0/2/0	
Source:		RenderCS	
Default Value	e:	0x0000000	
Access:		R/W	
Size (in bits):		32	
Trusted Type:		1	
Address:		043A8h	
DWord	Bit	Description	
0	31:22	Reserved	
	21:16	Number of GAC WR requests to be accumulated before applying the arbitration	
15:14		Reserved	
	13:8	Number of GAC R requests to be accumulated before applying the arbitration	
7:6		Reserved	
	5:0	Number of GAC RO requests to be accumulated before applying the arbitration	



## ARB\_GAC\_GAM\_REQCNTS1 - GAC\_GAM Arbitration Counters Register 1

Register Space:		MMIO: 0/2/0		
Source:		RenderCS		
Default Value	e:	0x0000000		
Access:		R/W		
Size (in bits):		32		
Trusted Type	9:	1		
Address:		043ACh		
DWord	Bit	Description		
0	31:22	Reserved		
	21:16	Number of GAC WR requests to be accumulated before applying the arbitration		
	15:14	Reserved		
	13:8	Number of GAC R requests to be accumulated before applying the arbitration		
	7:6	Reserved		
	5:0	Number of GAC RO requests to be accumulated before applying the arbitration		



## **ARB\_R\_GAC\_GAM0 - GAC\_GAM R Arbitration Register 0**

Register Space:	MMI	D: 0/2/0
Source:	Rend	erCS
Default Value:	0x000	00000
Access:	R/W	
Size (in bits):	32	
Trusted Type:	1	
Address:	043E0	Dh
DWord	Bit	Description
0	31:28	Reserved
	27	Priority for entry 1
	26:24	Goto field for entry 1 when request vector is 11b
	23:21	Goto field for entry 1 when request vector is 10b
	20:18	Goto field for entry 1 when request vector is 01b
	17:15	Goto field for entry 1 when request vector is 00b
	14:13	Reserved
	12	Priority for entry 0
	11:9	Goto field for entry 0 when request vector is 11b
	8:6	Goto field for entry 0 when request vector is 10b
	5:3	Goto field for entry 0 when request vector is 01b
	2:0	Goto field for entry 0 when request vector is 00b



#### **ARB\_R\_GAC\_GAM1 - GAC\_GAM R Arbitration Register 1**

Register Space:	MMI	O: 0/2/0
Source:	Rend	erCS
Default Value:	0x000	000000
Access:	R/W	
Size (in bits):	32	
Trusted Type:	1	
Address:	043E4	4h
DWord	Bit	Description
0	31:28	Reserved
	27	Priority for entry 3
	26:24	Goto field for entry 3 when request vector is 11b
	23:21	Goto field for entry 3 when request vector is 10b
	20:18	Goto field for entry 3 when request vector is 01b
	17:15	Goto field for entry 3 when request vector is 00b
	14:13	Reserved
	12	Priority for entry 2
	11:9	Goto field for entry 2 when request vector is 11b
	8:6	Goto field for entry 2 when request vector is 10b
	5:3	Goto field for entry 2 when request vector is 01b
	2:0	Goto field for entry 2 when request vector is 00b



## **ARB\_R\_GAC\_GAM2 - GAC\_GAM R Arbitration Register 2**

Register Space:	MMI	O: 0/2/0
Source:	Rend	lerCS
Default Value:	0x00	000000
Access:	R/W	
Size (in bits):	32	
Trusted Type:	1	
Address:	043E	8h
DWord	Bit	Description
0	31:28	Reserved
	27	Priority for entry 5
	26:24	Goto field for entry 5 when request vector is 11b
	23:21	Goto field for entry 5 when request vector is 10b
	20:18	Goto field for entry 5 when request vector is 01b
	17:15	Goto field for entry 5 when request vector is 00b
	14:13	Reserved
	12	Priority for entry 4
	11:9	Goto field for entry 4 when request vector is 11b
	8:6	Goto field for entry 4 when request vector is 10b
	5:3	Goto field for entry 4 when request vector is 01b
	2:0	Goto field for entry 4 when request vector is 00b



## **ARB\_R\_GAC\_GAM3 - GAC\_GAM R Arbitration Register 3**

Register Space:	MMI	D: 0/2/0
Source:	Rend	erCS
Default Value:	0x00	00000
Access:	R/W	
Size (in bits):	32	
Trusted Type:	1	
Address:	043E	Ch
DWord	Bit	Description
0	31:28	Reserved
	27	Priority for entry 7
	26:24	Goto field for entry 7 when request vector is 11b
	23:21	Goto field for entry 7 when request vector is 10b
	20:18	Goto field for entry 7 when request vector is 01b
	17:15	Goto field for entry 7 when request vector is 00b
	14:13	Reserved
	12	Priority for entry 6
	11:9	Goto field for entry 6 when request vector is 11b
	8:6	Goto field for entry 6 when request vector is 10b
	5:3	Goto field for entry 6 when request vector is 01b
	2:0	Goto field for entry 6 when request vector is 00b



## ARB\_RO\_GAC\_GAM0 - GAC\_GAM RO Arbitration Register 0

Register Space:	MMIC	D: 0/2/0
Source:	Rende	erCS
Default Value:	0x000	00000
Access:	R/W	
Size (in bits):	32	
Trusted Type:	1	
Address:	043D	0h
DWord	Bit	Description
0	31:28	Reserved
	27	Priority for entry 1
	26:24	Goto field for entry 1 when request vector is 11b
	23:21	Goto field for entry 1 when request vector is 10b
	20:18	Goto field for entry 1 when request vector is 01b
	17:15	Goto field for entry 1 when request vector is 00b
	14:13	Reserved
	12	Priority for entry 01
	11:9	Goto field for entry 01 when request vector is 11b
	8:6	Goto field for entry 01 when request vector is 10b
	5:3	Goto field for entry 01 when request vector is 01b
	2:0	Goto field for entry 01 when request vector is 00b



## ARB\_RO\_GAC\_GAM1 - GAC\_GAM RO Arbitration Register 1

Register Space:	MMIC	0: 0/2/0
Source:	Rende	erCS
Default Value:	0x000	00000
Access:	R/W	
Size (in bits):	32	
Trusted Type:	1	
Address:	043D4	łh
DWord	Bit	Description
0	31:28	Reserved
	27	Priority for entry 3
	26:24	Goto field for entry 3 when request vector is 11b
	23:21	Goto field for entry 3 when request vector is 10b
-	20:18	Goto field for entry 3 when request vector is 01b
	17:15	Goto field for entry 3 when request vector is 00b
	14:13	Reserved
	12	Priority for entry 2
	11:9	Goto field for entry 2 when request vector is 11b
	8:6	Goto field for entry 2 when request vector is 10b
	5:3	Goto field for entry 2 when request vector is 01b
	2:0	Goto field for entry 2 when request vector is 00b



## ARB\_RO\_GAC\_GAM2 - GAC\_GAM RO Arbitration Register 2

Register Space:	MMIC	): 0/2/0
Source:	Rende	erCS
Default Value:	0x000	00000
Access:	R/W	
Size (in bits):	32	
Trusted Type:	1	
Address:	043D8	3h
DWord	Bit	Description
0	31:28	Reserved
	27	Priority for entry 5
	26:24	Goto field for entry 5 when request vector is 11b
	23:21	Goto field for entry 5 when request vector is 10b
	20:18	Goto field for entry 5 when request vector is 01b
	17:15	Goto field for entry 5 when request vector is 00b
	14:13	Reserved
	12	Priority for entry 4
	11:9	Goto field for entry 4 when request vector is 11b
	8:6	Goto field for entry 4 when request vector is 10b
	5:3	Goto field for entry 4 when request vector is 01b
	2:0	Goto field for entry 4 when request vector is 00b



## ARB\_RO\_GAC\_GAM3 - GAC\_GAM RO Arbitration Register 3

Register Space:	MMIC	D: 0/2/0
Source:	Rende	erCS
Default Value:	0x000	00000
Access:	R/W	
Size (in bits):	32	
Trusted Type:	1	
Address:	043D0	Ch
DWord	Bit	Description
0	31:28	Reserved
	27	Priority for entry 7
	26:24	Goto field for entry 7 when request vector is 11b
	23:21	Goto field for entry 7 when request vector is 10b
	20:18	Goto field for entry 7 when request vector is 01b
	17:15	Goto field for entry 7 when request vector is 00b
	14:13	Reserved
	12	Priority for entry 6
	11:9	Goto field for entry 6 when request vector is 11b
	8:6	Goto field for entry 6 when request vector is 10b
	5:3	Goto field for entry 6 when request vector is 01b
	2:0	Goto field for entry 6 when request vector is 00b



ARB_WF	R_GAC_	GAM0 - GAC_GAM WR Arbitration Register
Register Space:	MMIO	: 0/2/0
Source:	Rende	rCS
Default Value:	0x0000	00000
Access:	R/W	
Size (in bits):	32	
Trusted Type:	1	
Address:	043F0I	h
DWord	Bit	Description
0	31:28	Reserved
	27	Priority for entry 1
	26:24	Goto field for entry 1 when request vector is 11b
	23:21	Goto field for entry 1 when request vector is 10b
	20:18	Goto field for entry 1 when request vector is 01b
	17:15	Goto field for entry 1 when request vector is 00b
	14:13	Reserved
	12	Priority for entry 0
	11:9	Goto field for entry 0 when request vector is 11b
	8:6	Goto field for entry 0 when request vector is 10b
	5:3	Goto field for entry 0 when request vector is 01b
	2:0	Goto field for entry 0 when request vector is 00b



ARB_WF	R_GAC_	GAM1 - GAC_GAM WR Arbitration Register
Register Space:	MMIO	: 0/2/0
Source:	Rende	rCS
Default Value:	0x0000	00000
Access:	R/W	
Size (in bits):	32	
Trusted Type:	1	
Address:	043F4	h
DWord	Bit	Description
0	31:28	Reserved
	27	Priority for entry 3
	26:24	Goto field for entry 3 when request vector is 11b
	23:21	Goto field for entry 3 when request vector is 10b
	20:18	Goto field for entry 3 when request vector is 01b
	17:15	Goto field for entry 3 when request vector is 00b
	14:13	Reserved
	12	Priority for entry 2
	11:9	Goto field for entry 2 when request vector is 11b
	8:6	Goto field for entry 2 when request vector is 10b
	5:3	Goto field for entry 2 when request vector is 01b
	2:0	Goto field for entry 2 when request vector is 00b



ARB_WF	R_GAC_	GAM2 - GAC_GAM WR Arbitration Register
Register Space:	MMIO	: 0/2/0
Source:	Rende	rCS
Default Value:	0x0000	00000
Access:	R/W	
Size (in bits):	32	
Trusted Type:	1	
Address:	043F8ł	1
DWord	Bit	Description
0	31:28	Reserved
	27	Priority for entry 5
	26:24	Goto field for entry 5 when request vector is 11b
	23:21	Goto field for entry 5 when request vector is 10b
	20:18	Goto field for entry 5 when request vector is 01b
	17:15	Goto field for entry 5 when request vector is 00b
	14:13	Reserved
	12	Priority for entry 4
	11:9	Goto field for entry 4 when request vector is 11b
	8:6	Goto field for entry 4 when request vector is 10b
	5:3	Goto field for entry 4 when request vector is 01b
	2:0	Goto field for entry 4 when request vector is 00b



ARB_WF	R_GAC_	GAM3 - GAC_GAM WR Arbitration Register
Register Space:	MMIO	: 0/2/0
Source:	Rende	rCS
Default Value:	0x0000	00000
Access:	R/W	
Size (in bits):	32	
Trusted Type:	1	
Address:	043FC	h
DWord	Bit	Description
0	31:28	Reserved
	27	Priority for entry 7
	26:24	Goto field for entry 7 when request vector is 11b
	23:21	Goto field for entry 7 when request vector is 10b
	20:18	Goto field for entry 7 when request vector is 01b
	17:15	Goto field for entry 7 when request vector is 00b
	14:13	Reserved
	12	Priority for entry 6
	11:9	Goto field for entry 6 when request vector is 11b
	8:6	Goto field for entry 6 when request vector is 10b
	5:3	Goto field for entry 6 when request vector is 01b
	2:0	Goto field for entry 6 when request vector is 00b



#### GAC\_HWSP\_REG - GAC Hardware Status Page Address Register

Register Space:	MMIO: 0/2/0
Default Value:	0x0000000
Size (in bits):	32
Address:	04180h-04183h

This register is used to program the 4 KB-aligned System Memory address of the Hardware Status Page used to report hardware status into (typically cacheable) System Memory. This address in this register is translated using the Global GTT in memory. The mapping type of the GTT entry determines the snoop nature of the transaction to memory.

DWord	Bit		Description	
0	31:12	GAC HWSP Register		
		Default Value:	00000h	
		Access:	R/W	
	11:0	Reserved		
		Default Value:	000h	
		Access:	RO	



## DONE\_REG - GAM Fub Done Lookup Register

Register Space:	M	MIO: 0/2/0					
Default Value:	0x(	0000000					
Size (in bits):	32						
Address:	Address: 040B0h-040B3h						
Gam Fub Don	e Lookup	o Registe	r				
DWord	Bit		Descriptio	on			
0	31:0	Gam Fub	Done Lookup Reg	<u> </u>			
		Default	/alue:	0000000h			
		Access:		RO			
		31	CVS Credit Fifo is Empty				
		30	CVS TLB Don't have any Cycles				
		29	Z Credit fifo is empty				
		28	ZTLB Don't have any cycles				
		27	RCC Credit Fifo is empty				
		26	RCC TLB Don't have any cycles				
		25	L3 Credit fifo is empty				
		24	L3 TLB is don't have any Cycles				
		23	VLF Credit fifo is empty				
		22	VLF TLB don't have any cycles				
		21	CASC Credit fifo empty				
		20	CASC TLB don't have any Cycles				
		19	Miss Fub Done				
		18	Read Stream Done				
		17	Read Steam Fifo is empty				
		16	Recycle Fifo in rstrm is empty				



## DONE\_REG - GAM Fub Done Lookup Register

	15	TLB Pend Done
	14	TLB Pend PQ Array Is done
	13	TLB pend PB Array is done
	12	Read route fub is done
	11	Gafm Data fifo is empty
	10	GAP data fifo is empty
	9	GAC data fifo is empty
	8	Wrdp is done with all the cycles
	7	Wrdp RID fifo is empty
	6	No hold from midarb to RTSTRM
	5	No hold from TLBPEND to MIDARB
	4	Reserved
	3	Tied to "1" - to be defined
	2	Fence FSM are IDLE
	1	Non PD Load Done
	0	Tied to "1" - to be defined



#### **GAM\_HWSP\_REG - GAM Hardware Status Page Address** Register

Register Space:	MMIO: 0/2/0
Default Value:	0x00000000
Size (in bits):	32
Address:	04080h-04083h

This register is used to program the 4 KB-aligned System Memory address of the Hardware Status Page used to report hardware status into (typically cacheable) System Memory. This address in this register is translated using the Global GTT in memory. The mapping type of the GTT entry determines the snoop nature of the transaction to memory.

DWord	Bit	Description	
0	31:12	GAM HWSP Register	
		Default Value: 00	000h
		Access: R/	W
	11.0	Decoursed	
	11:0	Reserved	
		Default Value:	000h
		Access:	RO



## **GFX\_PRIO\_CTRL - GFX Arbiter Client Priority Control**

Register Space		MIO: 0/2/0				
Default Value:		00011D10				
Size (in bits):	32					
Address:	04	02Ch-0402Fh				
GFX Arbiter	Client Pric	ority Control				
DWord	Bit		Description			
0	31:17	Extra 402C Register	1			
		Default Value:	000000000000000000000000000000000000000	00b		
		Access:	R/W			
		Extra 402C Register				
	16:12	Read Rstrm Max Reject				
		Default Value:		10001b		
		Access:		R/W		
		Read Rstrm Max Reject				
	11:9	gapc_gam_c_priority				
		Default Value:	110b			
		Access:	R/W			
		gapc_gam_c_priority - Lov	west Bit [9] is not used			
	8:6	gapc_gam_z_priority				
		Default Value:		100b		
		Access:	R/W			
		gapc_gam_z_priority - Lov	west Bit [6] is not used			
	5:3	gapc_gam_l3_priority				
		Default Value:		010b		
		Access:		R/W		
		gapc_gam_I3_priority - Lo	owest Bit [3] is not used			

Г



٦

GFX	FX_PRIO_CTRL - GFX Arbiter Client Priority Control							
	-							
	2:0	gafm_gam_priority						
		Default Value:	000b					
		Access:	R/W					
		Client Priority control bitss	·					
		gafm_gam_priority - Lowest Bit [0] is not used						



MID	ARB	_ <b>GO</b> 1	<b>OFI</b>	ELD_HIT0 - Goto Field Arbitration for Hit0				
Register S	oace:	MMIC	D: 0/2/0					
Source:		Rende	erCS					
Default Va	lue:	0x000	00000					
Access:		R/W						
Size (in bit	s):	32						
Trusted Ty	pe:	1						
Address:		043B(	Dh					
DWord	Bit			Description				
0	31:16	Reserve	ed					
		Format	:		MBZ			
	15:14			n request vector is 111				
				GOTO and priority register to be used no				
		Value	Name					
		00b		Use MIDARB_GOTOFIELD_HIT0 and MII				
		01b		Use MIDARB_GOTOFIELD_HIT1 and MI				
		10b		Use MIDARB_GOTOFIELD_HIT2 and MI				
	1	11b		Use MIDARB_GOTOFIELD_HIT3 and MII	DARB_PRIO_HIT_REGISTER[11:9]			
	13:12	Goto field when request vector is 110b.						
	11:10	Goto field when request vector is 101b.						
	9:8	Goto fi	Goto field when request vector is 100b.					
	7:6	Goto field when request vector is 011b.						
	5:4	Goto fie	eld whe	n request vector is 010b.				
	3:2	Goto fie	eld whe	n request vector is 001b.				
	1:0	Goto fi	eld whe	n request vector is 000b.				



MID	ARB	_ <b>GO</b> 1	OFI	ELD_HIT1 - Goto Field in Programmable Arbitration for Hit1				
Register Sp	oace:	MMIC	D: 0/2/0					
Source:		Rende	erCS					
Default Va	lue:	0x000	00000					
Access:		R/W						
Size (in bit	s):	32						
Trusted Ty	pe:	1						
Address:		043B4	4h					
DWord	Bit			Description				
0	31:16	Reserve	ed					
		Format	:	MBZ				
	15:14	Goto fi	eld whe	n request vector is 111				
		Determi		GOTO and priority register to be used next				
		Value	Name	Description				
		00b		Use MIDARB_GOTOFIELD_HIT0 and MIDARB_PRIO_HIT_REGISTER[2:0]				
		01b		Use MIDARB_GOTOFIELD_HIT1 and MIDARB_PRIO_HIT_REGISTER[5:3]				
		10b		Use MIDARB_GOTOFIELD_HIT2 and MIDARB_PRIO_HIT_REGISTER[8:6]				
		11b		Use MIDARB_GOTOFIELD_HIT3 and MIDARB_PRIO_HIT_REGISTER[11:9]				
	13:12	Goto fi	Goto field when request vector is 110b.					
	11:10	Goto fie	Goto field when request vector is 101b.					
	9:8	Goto fi	Goto field when request vector is 100b.					
	7:6	Goto fi	Goto field when request vector is 011b.					
	5:4	Goto fie	Goto field when request vector is 010b.					
	3:2	Goto fi	eld whe	n request vector is 001b.				
	1:0	Goto fi	eld whe	n request vector is 000b.				



MID	ARB	_ <b>GO</b> 1	<b>OFI</b>	ELD_HIT2 - Goto Field Arbitration for Hit2		
Register S	pace:	MMIC	D: 0/2/0			
Source:		Rend	erCS			
Default Va	lue:	0x000	00000			
Access:		R/W				
Size (in bit	s):	32				
Trusted Ty	pe:	1				
Address:		043B	3h			
DWord	Bit			Description		
0	31:16	Reserve	ed			
		Format	•		MBZ	
	15:14			<b>n request vector is 111.</b> GOTO and priority register to be used n	ext	
		Value	Name	Descri	ption	
		00b		Use MIDARB_GOTOFIELD_HIT0 and MI	DARB_PRIO_HIT_REGISTER[2:0]	
		01b		Use MIDARB_GOTOFIELD_HIT1 and MI	DARB_PRIO_HIT_REGISTER[5:3]	
		10b		Use MIDARB_GOTOFIELD_HIT2 and MI	DARB_PRIO_HIT_REGISTER[8:6]	
		11b		Use MIDARB_GOTOFIELD_HIT3 and MI	DARB_PRIO_HIT_REGISTER[11:9]	
	13:12	Goto field when request vector is 110b.				
	11:10	Goto fi	eld whe	n request vector is 101b.		
	9:8	Goto field when request vector is 100b.				
	7:6	Goto fi	eld whe	n request vector is 011b.		
	5:4	Goto field when request vector is 010b.				
	3:2	Goto fi	eld whe	n request vector is 001b.		
	1:0	Goto fi	eld whe	n request vector is 000b.		



MID	ARB	_ <b>GO</b>	ΓOFI	ELD_HIT3 - Goto Field in Programmable Arbitration for Hit3			
Register S	pace:	MMI	O: 0/2/0				
Source:		Rend	lerCS				
Default Va	lue:	0x00	000000				
Access:		R/W					
Size (in bit	:s):	32					
Trusted Ty	vpe:	1					
Address:		043B	Ch				
DWord	Bit			Description			
0	31:16	Reserve	d				
		Format	:	MBZ			
	15:14	Determi Field fo	<b>Goto field when request vector is 111.</b> Determines the GOTO and priority register to be used next. Field for arbitration on next clock cycle for request entries of 111 corresponding to arbitration action field entry of MIDARB_PRIO_HIT_REGISTER[11:9]				
		Value	Name	Description			
		00b		Use MIDARB_GOTOFIELD_HIT0 and MIDARB_PRIO_HIT_REGISTER[2:0]			
		01b		Use MIDARB_GOTOFIELD_HIT1 and MIDARB_PRIO_HIT_REGISTER[5:3]			
		10b		Use MIDARB_GOTOFIELD_HIT2 and MIDARB_PRIO_HIT_REGISTER[8:6]			
		11b		Use MIDARB_GOTOFIELD_HIT3 and MIDARB_PRIO_HIT_REGISTER[11:9]			
	13:12	Field for	arbitrat	<b>n request vector is 110.</b> ion on next clock cycle for request entries of 110 corresponding to arbitration y of MIDARB_PRIO_HIT_REGISTER[11:9]			
	11:10	<b>Goto field when request vector is 101.</b> Field for arbitration on next clock cycle for request entries of 101 corresponding to arbitration action field entry of MIDARB_PRIO_HIT_REGISTER[11:9]					
	9:8	<b>Goto field when request vector is 100.</b> Field for arbitration on next clock cycle for request entries of 100 corresponding to arbitratio action field entry of MIDARB_PRIO_HIT_REGISTER[11:9]					
	7:6	Goto field when request vector is 011. Field for arbitration on next clock cycle for request entries of 011 corresponding to arbitra action field entry of MIDARB_PRIO_HIT_REGISTER[11:9]					
	5:4	<b>Goto field when request vector is 010.</b> Field for arbitration on next clock cycle for request entries of 010 corresponding to arbitration action field entry of MIDARB_PRIO_HIT_REGISTER[11:9]					
	3:2	Field for	arbitrat	n request vector is 001. ion on next clock cycle for request entries of 001 corresponding to arbitration y of MIDARB_PRIO_HIT_REGISTER[11:9]			
	1:0	Goto fie	eld whei	n request vector is 000.			



# MIDARB\_GOTOFIELD\_HIT3 - Goto Field in Programmable Arbitration for Hit3 Field for arbitration on next clock cycle for request entries of 000 corresponding to arbitration action field entry of MIDARB\_PRIO\_HIT\_REGISTER[11:9]



MID	ARB	_ <b>GO</b> T		ELD_NP0 - Goto Field in Programmable Arbitration for Hit-NP0			
Register Sp	oace:	MMIC	D: 0/2/0				
Source:		Rende	erCS				
Default Va	lue:	0x000	00000				
Access:		R/W					
Size (in bit	s):	32					
Trusted Ty	pe:	1					
Address:		043C0	Dh				
DWord	Bit			Description			
0	31:30			n request vector is 1111.			
				GOTO and priority register to be used next.			
		Value	Name	Description			
		00b		Use MIDARB_GOTOFIELD_NP0 and MIDARB_PRIO_NP_REGISTER[4:0]			
		01b		Use MIDARB_GOTOFIELD_NP1 and MIDARB_PRIO_NP_REGISTER[9:5]			
		10b		Use MIDARB_GOTOFIELD_NP2 and MIDARB_PRIO_NP_REGISTER[14:10]			
	1	11b		Use MIDARB_GOTOFIELD_NP3 and MIDARB_PRIO_NP_REGISTER[19:15]			
	29:28	Goto field when request vector is 1110b.					
	27:26	Goto fi	eld whe	n request vector is 1101b.			
	25:24	Goto field when request vector is 1100b.					
	23:22	Goto field when request vector is 1011b.					
	21:20	Goto field when request vector is 1010b.					
	19:18	Goto fi	eld whe	n request vector is 1001b.			
	17:16	Goto field when request vector is 1000b.					
	15:14	Goto fi	eld whe	n request vector is 0111b.			
	13:12	Goto fi	eld whe	n request vector is 0110b.			
	11:10	Goto fi	Goto field when request vector is 0101b.				
	9:8	Goto fi	eld whe	n request vector is 0100b.			
	7:6	Goto fi	eld whe	n request vector is 0011b.			
	5:4	Goto fi	eld whe	n request vector is 0010b.			
	3:2	Goto fi	eld whe	n request vector is 0001b.			
	1:0	Goto fi	eld whe	n request vector is 0000b.			



MIDARB_GOTOFIELD_NP1 - Goto Field in Programmable
Arbitration for Hit-NP1

Register Sp	Space: MMIO: 0/2/0		D: 0/2/0				
Source:	RenderCS		erCS				
Default Value: 0x0000000		00000					
Access:		R/W					
Size (in bits	s):	32					
Trusted Typ	pe:	1					
Address:		043C4	4h				
DWord	Bit			Description			
0	31:30			n request vector is 1111.			
		Determi		GOTO and priority register to be used next.			
		Value	Name	Description			
		00b		Use MIDARB_GOTOFIELD_NP0 and MIDARB_PRIO_NP_REGISTER[4:0]			
		01b		Use MIDARB_GOTOFIELD_NP1 and MIDARB_PRIO_NP_REGISTER[9:5]			
		10b		Use MIDARB_GOTOFIELD_NP2 and MIDARB_PRIO_NP_REGISTER[14:10]			
		11b		Use MIDARB_GOTOFIELD_NP3 and MIDARB_PRIO_NP_REGISTER[19:15]			
	29:28	Goto fi	eld whei	n request vector is 1110b.			
	27:26	Goto fie	eld whei	n request vector is 1101b.			
	25:24	Goto fi	eld whei	n request vector is 1100b.			
	23:22	Goto fi	eld whei	n request vector is 1011b.			
	21:20	Goto fi	eld whei	n request vector is 1010b.			
	19:18	Goto fi	eld whei	n request vector is 1001b.			
	17:16	Goto fi	eld whei	n request vector is 1000b.			
	15:14	Goto fi	Goto field when request vector is 0111b.				
	13:12	Goto fi	eld whei	n request vector is 0110b.			
	11:10	Goto fi	eld whei	n request vector is 0101b.			
	9:8	Goto field when request vector is 0100b.					
	7:6	Goto fi	eld whei	n request vector is 0011b.			
	5:4	Goto fi	eld whei	n request vector is 0010b.			
	3:2	Goto fi	eld whe	n request vector is 0001b.			
	1:0	Goto fi	eld whe	n request vector is 0000b.			



MID	ARB	_ <b>GO</b> T		ELD_NP2 - Goto Field in Programmable Arbitration for Hit-NP2			
Register Sp	oace:	MMIC	D: 0/2/0				
Source:		Rende	erCS				
Default Va	lue:	0x000	00000				
Access:		R/W					
Size (in bit	s):	32					
Trusted Ty	pe:	1					
Address:		043C8	3h				
DWord	Bit			Description			
0	31:30			n request vector is 1111.			
				GOTO and priority register to be used next.			
		Value	Name	Description			
		00b		Use MIDARB_GOTOFIELD_NP0 and MIDARB_PRIO_NP_REGISTER[4:0]			
		01b		Use MIDARB_GOTOFIELD_NP1 and MIDARB_PRIO_NP_REGISTER[9:5]			
		10b		Use MIDARB_GOTOFIELD_NP2 and MIDARB_PRIO_NP_REGISTER[14:10]			
		11b		Use MIDARB_GOTOFIELD_NP3 and MIDARB_PRIO_NP_REGISTER[19:15]			
	29:28	Goto field when request vector is 1110b.					
	27:26	Goto fi	eld whe	n request vector is 1101b.			
	25:24	Goto field when request vector is 1100b.					
	23:22	Goto field when request vector is 1011b.					
	21:20	Goto fi	Goto field when request vector is 1010b.				
	19:18	Goto fi	eld whe	n request vector is 1001b.			
	17:16	Goto field when request vector is 1000b.					
	15:14	Goto fi	eld whe	n request vector is 0111b.			
	13:12	Goto fi	eld whe	n request vector is 0110b.			
	11:10	Goto fi	eld whe	n request vector is 0101b.			
	9:8	Goto fi	Goto field when request vector is 0100b.				
	7:6	Goto fi	Goto field when request vector is 0011b.				
	5:4	Goto fi	eld whe	n request vector is 0010b.			
	3:2	Goto fi	eld whe	n request vector is 0001b.			
	1:0	Goto fi	eld whe	n request vector is 0000b.			



MIDARB_GOTOFIELD_NP3 - Goto Field in Programmable
Arbitration for Hit-NP3

Register Sp	200.						
	ister Space: MMIO: 0/2/0		D: 0/2/0				
Source:		RenderCS					
Default Value: 0x0000000		00000					
Access:		R/W					
Size (in bits	5):	32					
Trusted Typ	be:	1					
Address:		043C0	Ch				
DWord	Bit			Description			
0	31:30	Goto fie	eld whei	n request vector is 1111.			
		Determi	ines the	GOTO and priority register to be used next.			
		Value	Name	Description			
		00b		Use MIDARB_GOTOFIELD_NP0 and MIDARB_PRIO_NP_REGISTER[4:0]			
		01b		Use MIDARB_GOTOFIELD_NP1 and MIDARB_PRIO_NP_REGISTER[9:5]			
		10b		Use MIDARB_GOTOFIELD_NP2 and MIDARB_PRIO_NP_REGISTER[14:10]			
		11b		Use MIDARB_GOTOFIELD_NP3 and MIDARB_PRIO_NP_REGISTER[19:15]			
	29:28	Goto fie	eld whei	n request vector is 1110b.			
	27:26	Goto fie	eld whei	n request vector is 1101b.			
	25:24	Goto fie	eld whei	n request vector is 1100b.			
	23:22	Goto fie	eld whei	n request vector is 1011b.			
	21:20	Goto fie	eld whei	n request vector is 1010b.			
	19:18	Goto fie	eld whei	n request vector is 1001b.			
	17:16	Goto fie	eld whei	n request vector is 1000b.			
	15:14	Goto fie	eld whei	n request vector is 0111b.			
	13:12	Goto fie	eld whei	n request vector is 0110b.			
	11:10	Goto fie	eld whei	n request vector is 0101b.			
	9:8	Goto field when request vector is 0100b.					
	7:6	Goto fie	eld whei	n request vector is 0011b.			
	5:4	Goto fie	eld whei	n request vector is 0010b.			
	3:2	Goto fie	eld whei	n request vector is 0001b.			
	1:0	Goto fie	eld whe	n request vector is 0000b.			



## **GPGPU\_DISPATCHDIMX - GPGPU Dispatch Dimension X**

Register Space:		MMIO: 0/2/0					
Source:		RenderCS					
Default Value:		0x0000000					
Access:		R/W					
Size (in bits):		32					
Address:		02500h					
DWord	Bit	Description					
0	31:0	Dispatch Dimension X					
		Format:	L	U32			
		The number of thread groups to be dispatched in the X dimension (max $x + 1$ ).					
		Value	Name				
		1,FFFFFFFh					



## **GPGPU\_DISPATCHDIMY - GPGPU Dispatch Dimension Y**

Register Space:		MMIO: 0/2/0					
Source:		RenderCS					
Default Value:		0x0000000					
Access:		R/W					
Size (in bits):		32					
Address:		02504h					
DWord	Bit	Description					
0	31:0	Dispatch Dimension Y					
		Format:	at:				
		The number of thread groups to be dispatched in the Y dimension (max y + 1					
		Value	Name				
		1,FFFFFFFh					



## **GPGPU\_DISPATCHDIMZ - GPGPU Dispatch Dimension Z**

Register Space:		MMIO: 0/2/0					
Source:		RenderCS					
Default Value:		0x0000000					
Access:		R/W					
Size (in bits):		32					
Address:		02508h					
DWord	Bit	Description					
0	31:0	Dispatch Dimension Z					
		Format:	U3	2			
		The number of thread groups to be dispatched in the Zdimension (max Z + 1)					
		Value	Name				
		1,FFFFFFFh					



## **GDRST - Graphics Device Reset Control**

Register S	pace:	MMIO: 0/2/0					
Default Value:		0x0000000					
Size (in bit	:s):	32					
Address:		0941Ch					
Graphics [	Device	Reset Control Registers					
DWord	Bit	Description					
0	31:7	Reserved					
		Access: RO					
		Reserved					
=	6:4	Reserved					
		Access: RO					
		Reserved.					
-	3	Initiate Graphics Blitter Soft Reset					
		Access: R/W Set					
		Graphics Blitter Soft-Reset Control:					
		'1': Initiate a graphics blitter domain reset.					
		- Cleared by CP once the reset is complete '0': N/A					
		- Once set, clearing of this bit has no effect on CP. Only CP can reset this bit.					
		<b>Note:</b> This is a non-posted register.					
-		Initiate Creaties Media Coft Beent					
	2	Initiate Graphics Media Soft Reset       Access:     R/W Set					
		Graphics Media Soft-Reset Control:					
		'1': Initiate a graphics media 0 domain reset.					
		- Cleared by CP once the reset is complete					
		'0': N/A					
		- Once set, clearing of this bit has no effect on CP. Only CP can reset this bit.					
		<b>Note:</b> This is a non-posted register.					
-	1	Initiate Graphics Render Soft Reset					
		Access: R/W Set					
		Graphics Render Soft-Reset Control:					
		'1': Initiate a graphics render domain reset.					
		- Cleared by CP once the reset is complete					
		'0': N/A - Once set, clearing of this bit has no effect on CP. Only CP can reset this bit.					
		<b>Note:</b> This is a non-posted register.					



		phics Device Reset Control		
0	Initiate Graphics Full Soft	t Reset		
	Access:	R/W Set		
	Graphics Full Soft-Reset Control:			
	'1': Initiate a full graphics reset (i.e., graphics render, media, and blitter reset).			
	- Cleared by CP once the reset is complete			
	'0': N/A	•		
	- Once set, clearing of this bit has no effect on CP. Only CP can reset this bit.			
	<b>Note:</b> This is a non-posted register.			



## **GFX\_ENG\_FR - Graphics Engine Fault Register**

Register Space:	MMIO: 0/2/0
Default Value:	0x0000000
Size (in bits):	32
Address:	04094h-04097h

Graphics Engine Fault Register

DWord	Bit Description							
0	31:12	Virtual Address of Fault						
		Default Value:	00000h					
		Access:	R/W					
		This is the original Address of the Page that generated the	e First fault for this engine.					
		This value is locked and not updated on subsequent faults, until the valid bit of this register is cleared by SW						
	11	GTTSEL						
		Default Value:	0b					
		Access:	R/W					
		This bit indicates if the valid bit happened while using PPGTT or GGTT: 0 - PPGTT, 1 - GGTT						
		GGT						
		GGTT This value is locked and not updated on subsequent faul register is cleared by SW	ts, until the valid bit of this					
	10:3	This value is locked and not updated on subsequent faul register is cleared by SW	ts, until the valid bit of this					
	10:3	This value is locked and not updated on subsequent faul	ts, until the valid bit of this					
	10:3	This value is locked and not updated on subsequent fault register is cleared by SW          SRCID of Fault						
	10:3	This value is locked and not updated on subsequent fault register is cleared by SW SRCID of Fault Default Value:	00h R/W					
	10:3	This value is locked and not updated on subsequent fault register is cleared by SW SRCID of Fault Default Value: Access: This is the Source ID of the unit that requested the cycle	00h R/W that generated the First Page					
	2:1	This value is locked and not updated on subsequent fault register is cleared by SW <b>SRCID of Fault</b> Default Value: Access: This is the Source ID of the unit that requested the cycle fault for this engine. This value is locked and not updated on subsequent fault	00h R/W that generated the First Page					



## **GFX\_ENG\_FR - Graphics Engine Fault Register**

	Access:	R/W			
	Type of Fault recorded:				
	00 - Page Fault.				
	01 - Invalid PD Fault				
	10 - Unloaded PD Fault				
0	11 - Invalid and Unloaded PD fault				
	This value is locked and not updated on subsequent register is cleared by SW	faults, until the valid bit of this			
	Valid Bit				
	Default Value:	0b			
	Access:				
	Access.	R/W			
	This bit indicates that the first fault for this engine has cleared by SW, which will also clear the other fields.				



## **FENCE - Graphics Memory Fence Table Register**

Register Space:	MMIO: 0/2/0
Source:	BSpec
Default Value:	0x0000000, 0x0000000
Access:	R/W
Size (in bits):	64
Trusted Type:	1
Address:	100000h-100007h
Name:	FENCE_0
Address:	100008h-10000Fh
Name:	FENCE_1
Address:	100010h-100017h
Name:	FENCE_2
Address:	100018h-10001Fh
Name:	FENCE_3
Address:	100020h-100027h
Name:	FENCE_4
Address:	100028h-10002Fh
Name:	FENCE_5
Address:	100030h-100037h
Name:	FENCE_6
Address:	100038h-10003Fh
Name:	FENCE_7
Address:	100040h-100047h
Name:	FENCE_8
Address:	100048h-10004Fh
Name:	FENCE_9
Address:	100050h-100057h
Name:	FENCE_10
Address:	100058h-10005Fh
Name:	FENCE_11
Address:	100060h-100067h
Name:	FENCE_12
Address:	100068h-10006Fh
Name:	FENCE_13



### **FENCE - Graphics Memory Fence Table Register**

Address:	100070h-100077h
Name:	FENCE_14
Address:	100078h-10007Fh
Name:	FENCE_15

The graphics device performs address translation from linear space to tiled space for a CPU access to graphics memory (See Memory Interface Functions chapter for information on these memory layouts) using the fence registers. Note that the fence registers are used only for CPU accesses to gfx memory. Graphics rendering/display pipelines use Per Surface Tiling (PST) parameters (found in SURFACE\_STATE - see the Sampling Engine chapter) to access tiled gfx memory.

The intent of tiling is to locate graphics data that are close (in X and Y surface axes) in one physical memory page while still locating some amount of line oriented data sequentially in memory for display efficiency. All 3D rendering is done such that the QWords of any one span are all located in the same memory page, improving rendering performance. Applications view surfaces as linear, hence when the cpu access a surface that is tiled, the gfx hardware must perform linear to tiled address conversion and access the correct physical memory location(s) to get the data.

Tiled memory is supported for rendering and display surfaces located in graphics memory. A tiled memory surface is a surface that has a width and height that are subsets of the tiled region's pitch and height. The device maintains the constants required by the memory interface to perform the address translations. Each tiled region can have a different pitch and size. The CPU-memory interface needs the surface pitch and tile height to perform the address translation. It uses the GMAddr (PCI-BAR) offset address to compare with the fence start and end address, to determine if the rendering surface is tiled. The tiled address is generated based on the tile orientation determined from the matching fence register. Fence ranges are at least 4 KB aligned. Note that the fence registers are used only for CPU accesses to graphics memory.

A Tile represents 4 KB of memory. Tile height is 8 rows for X major tiles and 32 rows for Y major tiles. Tile Pitch is 512Bs for X major tiles and 128Bs for Y major tiles. The surface pitch is programmed in 128B units such that the pitch is an integer multiple of "tile pitch".

Engine restrictions on tile surface usage are detailed in Surface Placement Restrictions (Memory Interface Functions). Note that X major tiles can be used for Sampler, Color, Depth, motion compensation references and motion compensation destination, Display, Overlay, GDI Blt source and destination surfaces. Y major tiles can be used for Sampler, depth, color and motion compensation assuming they do not need to be displayed. GDI Blit operations, overlay and display cannot used Tiled Y orientations.

A "PST" graphics surface that will also be accessed via fence needs its base address to be tile row aligned.

Hardware handles the flushing of any pending cycles when software changes the fence upper/lower bounds.

Fence Table Registers occupy the address range specified above. Each Fence Table Register has the following format.

FENCE registers are not reset by a graphics reset. They will maintain their values unless a full chipset reset is performed.



DWord	Bit				Description
0	63:44	Fence l	Jpper Boun	d	
		Format	t:		GraphicsAddress[31:12]
		Bits 31:	12 of the en	ding G	raphics Address of the fence region. Fence regions must be aligned to
		а 4КВ р	age. This ad	ldress r	represents the last 4KB page of the fence region (Upper Bound is
			d in the fend		
		Graphi	cs Address i	s the o	ffset within GMADR space.
	42.42	_			
	43:42	Reserve Format			107
					MBZ
	41:32	Fence F		1.	
		Format			J10-1 Width in 128 bytes
					Ith (pitch) of the fence region in multiple of "tile width". For Tile X this
					d to a multiple of 512B ("003" is the minimum value) and for Tile Y this d to a multiple of 128B ("000" is the minimum value).
		neid mit	ist be progr	annie	
		000h =	128B		
		001h =	256B		
		3FFh =	128KB		
	31:12	Fence L	ower Boun	d	
		Format			GraphicsAddress[31:12]
					Graphics Address of the fence region. Fence regions must be aligned to
			is address re ce region).	epreser	nts the first 4KB page of the fence region (Lowe Bound is included in
			0	s the o	ffset within GMADR space.
		Crupin		5 110 0	
	11:2	Reserve			
		Format	t:		MBZ
	1	Tile Wa	alk		
		This fiel	d specifies t	the spa	tial ordering of QWords within tiles.
		Value	Name	e	Description
		0h	MI_TILE_XN	/IAJOR	Consecutive SWords (32 Bytes) sequenced in the X direction
		1h	MI TILE YN	1AJOR	Consecutive OWords (16 Bytes) sequenced in the Y
					direction
	0	Fence \	/alid		
		Format			MI_FenceValid
		This fiel	d specifies v	whethe	er or not this fence register defines a fence region.
		١	/alue		Name
		0h		MI_FE	NCE_INVALID
		1h		MI_FE	NCE_VALID



		G	FX_MODE -	Graphics Mode	e Register				
Register	Space:	. N	1MIO: 0/2/0						
Source:		R	enderCS						
Default \	/alue:	0	x00000800						
Size (in b	oits):	3	2						
Trusted	Туре:	1	1						
Address:		0	229Ch						
			I	Description					
This reg	ister co	ontains a	a control bit for the ne	ew 2-level PPGTT functions.					
Default	Value =	= 000028	300h						
DWord	Bit			Description					
0	31:16	Mask B	lits						
		Format	t:	Mask[15:0]					
		Must be	e set to modify corres	ponding bit in Bits 15:0. (All ir	mplemented bits)				
	15	Reserved							
	14	Reserved							
		Format	t:		MBZ				
	13	Flush T	LB invalidation Mod	e					
		Format	t:		U1				
		limits th which h	ne invalidation of the	ation if the TLB cache inside t TLB only to batch buffer boun on bit set and sync flushes. If eline.	ndaries, to pipe_control comm	nands			
	12	Reserve	ed						
		Format	t:		MBZ				
	11	Replay	Mode		·				
		Format	t: U1 Cor	ntext Switch Granularity					
		This field controls the granularity of the replay mechanism when coming back into a previous preempted context.							
		Value	Name	Desci	ription				
		0h	mid-triangle	Super span Level. Pipeline is	•				
			preemption	commands parsed are execu not complete before a conte					
		1h	mid-cmdbuffer	Drawcall Level. Pipeline is flu	ushed before switching to the	9			
			preemption [ <b>Default]</b>	next context. Commands par completing before a context					
				December 1		_			
		Programming Notes							



## **GFX\_MODE - Graphics Mode Register**

			the bit must be set.			
10	Reserv					
	Forma	t:	MBZ			
9	Per-Pro	ocess GTT Er	nable			
	Forma	t:	Enabled			
	Per-Pro	cess GTT Ena	able			
	Value	Name	Description			
	0h	PPGTT Disable [ <b>Default]</b>	When clear, the Global GTT will be used to translate memory access from designated commands and for commands that select the PPGTT as their translation space.			
	1h	PPGTT Enable	When set, the PPGTT will be used to translate memory access from designated commands and for commands that select the PPGTT as the translation space. The PD Offset and PD Cacheline Valid registers must set in all pipes (blitter, MFX, render) before any workload is submitted hardware. This mode enables support for big pages (32k).			
8	Reserv	ed				
	Forma	t:	MBZ			
7	Reserv	ed				
	Forma	t:	MBZ			
6:1	Reserv	ed				
	Format: MBZ					
0	Reserve	ed				
÷	Forma	t.	MBZ			



### **GS\_INVOCATION\_COUNT** - **GS** Invocation Counter

Register Space:		e: MMIO: 0/2/0				
Source:		RenderCS				
Default Value:		0x0000000, 0x0000000				
Access:		R/W				
Size (in b	oits):	64				
Trusted Type:		1				
Address:		02328h				
J J	This register stores the number of objects that are part of geometry shader threads. This register is part of the context save and restore.					
DWord	Bit	Description				
0	63:0	GS Invocation Count				
		Number of objects that are dispatched as a geometry shader threads invoked by the GS stage. Updated only when <b>Statistics Enable</b> is set in GS_STATE (see the Geometry Shader Chapter in the 3D Volume.)				



### **GS\_PRIMITIVES\_COUNT - GS Primitives Counter**

Register Space:		e: MMIO: 0/2/0				
Source:		RenderCS				
Default Value:		0x0000000, 0x0000000				
Access:		R/W				
Size (in b	oits):	64				
Trusted Type:		1				
Address:		02330h				
5		reflects the total number of primitives that have been output by the Geometry Shader stage. This t of the context save and restore.				
DWord	Bit	Description				
0	63:0	GS Primitives Count				
		Total number of primitives output by the geometry stage. Updated only when Statistics Enable is set in GS_STATE (see the Geometry Shader Chapter in the 3D Volume.)				



## FLRCTLMSG - GT Function Level Reset Control Message

Register	Space:	MMIO: 0/2/0				
Default \	/alue:	0×0000000				
Size (in b	oits):	32				
Address:		08100h				
GT FLR C	Control	Register				
DWord	Bit		De	scription		
0	31:16	Message Mask				
		Access:			RO	
		Message Mask				
		In order to write to bits 15:0, the corresponding message mask bits must be written.				
		For example, for bit 14 to be set,	bit 30 ne	eds to be 1: 400	04000	
	15:1	Reserved				
		Access:			RO	
		Reserved				
	0	Initiate GT Function Level Reset M	essage			
		Access:		R/W Set		
		GT Function Level Reset (FLR)				
		1: Initiate GT FLR				
		- This is a Non-Posted message t	o reset R	ender, Media, Bl	litter, and GTI-Device domains.	
		- This bit is cleared by the CPunit	upon co	mpletion of the	reset.	



Γ

		G	T_MODE - G	T Mode Re	egister		
Register	Space:	MMIO: 0	/2/0				
Source:		RenderC	S				
Default	Value:	0x00000	000				
Access:		R/W					
Size (in	bits):	32					
Trusted	Туре:	1					
Address	:	07008h					
			Descript	ion			
	0		ol the 6EU and 12EU con gister enables the 6EU	0			
Registe	rType =	= MMIO_SVL					
DWord	Bit			Description			
0	31:16	Mask Bits					
		Format: Mask[15:0]					
		Must be set to modify corresponding bit in Bits 15:0. (All implemented bits)					
	15	Reserved					
		Format:			MBZ		
	14:13	Reserved			·		
		Format:			MBZ		
	12:11	Reserved			-		
		Format:			MBZ		
	10	Reserved			1		
		Format:			MBZ		
	9	WIZ Hashing	Mode High Bit		·		
		Format:			U1		
		Value column i (low bit).	n the table below refer	s to this field (high b	with the WIZ Hashing Mode fie vit) and the WIZ Hashing Mode		
			n't care if the Hashing	Disable bit is set.			
		Value	Name		Description		
		0h	[Default]	8x8 Checkerboard	0		
		1h		8x4 Checkerboard	5		
		2h		16x4 Checkerboard	1 hashing		
		3h		Reserved			
				Programming Not	25		
				5			



		GT_MODE - GT Mode Re	gist	er	
		8x4 hashing preferred for when msaa enabled			
5	8	Reserved			
		Format:	MBZ		
	7	WIZ Hashing Mode			
		Format:		U1	
		Description			
		This field configures the Hashing mode in Windower. This Hashing Disable bit is set.	field is	don't care if the	
		The WIZ Hashing Mode High Bit field is combined with th modes.	is field t	to enable additional	
6	5:3	Reserved			
		Format:	MBZ		
	2	Reserved	I		
		Format:	MBZ		
	1	Reserved			
		Format:	MBZ		
(	0	Reserved			



### **HWSTAM - Hardware Status Mask Register**

Register Space:	MMIO: 0/2/0
Source:	RenderCS
Default Value:	0xFFFFFFF
Access:	R/W,RO
Size (in bits):	32
Trusted Type:	1
Address:	02098h

The HWSTAM register has the same format as the Interrupt Control Registers. The bits in this register are mask bits that prevent the corresponding bits in the Interrupt Status Register from generating a Hardware Status Write (PCI write cycle). Any unmasked interrupt bit (HWSTAM bit set to 0) will allow the Interrupt Status Register to be written to the ISR location (within the memory page specified by the Hardware Status Page Address Register) when that Interrupt Status Register bit changes state.

#### **Programming Notes**

- To write the interrupt to the HWSP, the corresponding IMR bit must also be clear (enabled).
- At most 1 bit can be unmasked at any given time.

DWord	Bit		Desc	ription
0	31:0	Hardware Status Mask Register		
		Default Value:		FFFFFFFh
		Format:		Array of Masks
		Refer to the Interrupt Control Regist	ter secti	on for bit definitions. Reserved bits are RO.



## HWS\_PGA - Hardware Status Page Address Register

Register S	pace:	MMIO: 0/2/0		
Source:		BSpec		
Default Va	alue:	0x0000000		
Access:	ess: R/W			
Size (in bits): 32				
Trusted Ty	/pe:	1		
Address:		22080h		
Name:		BCS Hardware Status Page Address Register		
ShortNam	ne:	BCS_HWS_PGA		
Address:		04080h		
Name:		RCS Hardware Status Page Address Register		
ShortNam	ne:	RCS_HWS_PGA		
Address:		12080h		
Name:		VCS Hardware Status Page Address Register		
ShortNam	ne:	VCS_HWS_PGA		
0		used to program the 4 KB-aligned System Memory address of the Hardware Status Page used to e status into (typically cacheable) System Memory.		
		Programming Notes		
If this reg streamer.		s written, a workload must subsequently be dispatched to the Render command		
DWord	Bit	Description		
0 3	31:12	Address		
		Format: GraphicsAddress[31:12]		
		This field is used by SW to specify Bits 31:12 of the 4 KB-aligned System Memory address		
		4 KB page known as the Hardware Status Page. The Global GTT is used to map this page from		
		the graphics virtual address to physical address.		
-	11:0	Reserved		
		Format: MBZ		



## **HS\_INVOCATION\_COUNT - HS Invocation Counter**

Register S	ister Space: MMIO: 0/2/0			
Source: RenderCS		RenderCS		
Default Value: 0x0000000, 0x0000000		0x0000000, 0x0000000		
Access: R/W				
Size (in bit	e (in bits): 64			
Trusted Ty	usted Type: 1			
Address:	lress: 02300h			
vertices wo	ould ca	es the number of patch objects processed by the HS unit. E.g., A PATCHLIST_2 topology with 6 use this counter to increment by 3 (there are 3 2-vertex patch objects in that topology). This the context save and restore.		
DWord	Bit	Description		
0	63:0	HS Invocation Count		
		Number of patch objects processed by the HS stage. Updated only when HS Enable and HS		
		Statistics Enable are set in 3DSTATE_HS		



IA_VERTICES_CO	DUNT - IA	Vertices	Count
----------------	-----------	----------	-------

Register	Space	ace: MMIO: 0/2/0		
Source:		RenderCS		
Default Value: 0x0000000, 0x0000000		0x0000000, 0x0000000		
Access:		R/W		
Size (in b	oits):	64		
Trusted <sup>-</sup>	sted Type: 1			
Address:	Address: 02310h			
This reg	ister s	stores the count of vertices processed by VF. This register is part of the context save and restore.		
DWord	Bit	Description		
0	63:0	IA Vertices Count Report		
		Total number of vertices fetched by the VF stage. This count is updated for every input vertex as		
		long as Statistics Enable is set in VF_STATE (see the Vertex Fetch Chapter in the 3D Volume.)		



### **INSTPM - Instruction Parser Mode Register**

Register Space:	MMIO: 0/2/0
Source:	RenderCS
Default Value:	0x0000000
Access:	R/W,RO
Size (in bits):	32
Trusted Type:	1
Address:	020C0h

The INSTPM register is used to control the operation of the Instruction Parser. Certain classes of instructions can be disabled (ignored) - often useful for detecting performance bottlenecks. Also, Synchronizing Flush operations can be initiated - useful for ensuring the completion (vs. only parsing) of rendering instructions.

#### **Programming Notes**

- If an instruction type is disabled, the parser will read those instructions but not process them.
- Error checking will be performed even if the instruction is ignored.
- All Reserved bits are implemented.

• This Register is saved and restored as part of Context.

DWord	Bit	Description		
0	31:16	Mask Bits		
		Format: Mask[15:0]		
		Masks: These bits serve as write enables for bits 15:0. If thi	5	
		bits clear the corresponding bit in the field 15:0 will not be	e modifie	ed. Reading these bits always
		returns 0s.		
	15	Reserved		
	14:13	Reserved		
		Format:	MBZ	
	12	Reserved		
	11	CLFLUSH Toggle		
		Access:		RO
		Format:	I	U1
		This bit changes polarity each time the MI_CLFLUSH comm	nand cor	npletes. This bit is Read Only.
	10	Reserved		
		Format:	MBZ	
	9	TLB Invalidate		
		Format:		U1
		If set, this bit allows the command stream engine to invalid valid only with the Sync flush enable. Note: GFX soft resets		



	driver to explicitly invalidate TLBs pos	t reset.		
8	Memory Sync Enable			
•	Format:	U1		
	If set, this bit allows the command strumemory. This bit is valid only with the	eam engine to write out the data from the local c e Sync flush enable		
7	Force Sync Command Ordering			
	Format:	Enable		
		Description		
	By default, driver/OS synchronization commands (MI_STORE_DATA_IMM, for instance) can execute out of order with respect to 3D state and 3D primitive commands. When set, this bit forces ordering of these commands. See section 3.2.2 for a list of these commands.			
	This bit should be programmed to 1.			
	This bit should be programmed to 1.			
	This bit should be programmed to 1.			
	Value	Name		
	Ob	[Default]		
	1b			
6	CONSTANT_BUFFER Address Offset	t Disable		
	Format:	Disable		
	· · · · · · · · · · · · · · · · · · ·	ONSTANT_* Buffers' Starting Address is used as a		
	DynamicStateOffset. That is, it serves as an offset from the Dynamic State Base Address. Acces			
	will be subject to Dynamic State bounds checking. When this bit is set, the 3DSTATE_CONSTANT_* Buffers' Starting Address is used as a true			
	GraphicsAddress (not an offset). No bounds checking will be performed during access.			
		5		
5		5		
5	GraphicsAddress (not an offset). No b	5		
5	GraphicsAddress (not an offset). No b Sync Flush Enable Format: This field is used to request a Sync Flu	U1 Jush operation. The device will automatically clear		
5	GraphicsAddress (not an offset). No b Sync Flush Enable Format: This field is used to request a Sync Flu before completing the operation. See	U1 U1 Sync Flush (Programming Environment).		
5	GraphicsAddress (not an offset). No b Sync Flush Enable Format: This field is used to request a Sync Flu before completing the operation. See	ounds checking will be performed during access U1 ush operation. The device will automatically clear		
	GraphicsAddress (not an offset). No b Sync Flush Enable Format: This field is used to request a Sync Flu before completing the operation. See • The command parser must be Stop Rings bit in register MI_N a Sync Flush be issued by setti	U1 U1 U1 U1 USH operation. The device will automatically clea Sync Flush (Programming Environment). Programming Notes e stopped prior to issuing this command by setti MODE. Only after observing Rings Idle set in MI_ ing this bit. Once this bit becomes clear again, in		
	GraphicsAddress (not an offset). No b Sync Flush Enable Format: This field is used to request a Sync Flu before completing the operation. See • The command parser must be Stop Rings bit in register MI_N a Sync Flush be issued by setti	U1 U1 Sync Flush (Programming Environment).		



## **INSTPM - Instruction Parser Mode Register**

3	Media Instruction Disable		
	Format:	U1	
	This bit instructs the Renderer instruction parser to not execute them.	parse and error-check Media instructions,	but
	Format = Disable		
2	3D Rendering Instruction Disable		
	Format:	U1	
	instructions, but not execute them. This bit mus Instruction Disable is set. Setting this bit without allowed.		
	Format = Disable		
1	Reserved		
0	Texture Palette Load Instruction Disable		
	Format:	U1	
	This bit instructs the Renderer instruction parser to instructions, but not execute them.	parse and error-check Texture Palette Loa	d
	Format = Disable		



		IMR -	Interrupt N	lask Register
Register S	oace:	MMIO: 0/2/0		
Source:		RenderCS		
Default Va	lue:	0xFFFFFFFF		
Access:		R/W,RO		
Size (in bit	s):	32		
Address:		020A8h		
Unmasked cleared by	bits w softwa	ill be reported in the	IIR, possibly triggering not be reported in the I	ot Status Register bits are masked or unmasked. a CPU interrupt, and will persist in the IIR until IR and therefore cannot generate CPU interrupts.
DWord	Bit		D	escription
0	31:0	Interrupt Mask Bits	i	
		Format: InterruptN	lask[32] Refer to the In	terrupt Control Register section for bit definitions.
			bit mask which selects s in the Interrupt Contr	which interrupt bits (from the ISR) are reported in ol Register are RO.
		Value	Name	Description
		FFFF FFFFh	[Default]	
		0h	Not Masked	Will be reported in the IIR
		1h	Masked	Will not be reported in the IIR



### L3\_LRA\_0 - L3 LRA 0

Register Space:	MMIC	D: 0/2/0		
Default Value:	0x3F2	01F00		
Size (in bits):	32			
Address:	04030	Ch-0403Fh		
L3 LRA 0				
DWord	Bit		Description	
0	31:24	L3 LRA1 Max		
		Default Value:	00111111b	
		Access:	R/W	
		L3 LRA1 Max		
		<b>F</b>		
		Format: U6		
		Maximum value of programr	nable LRA1	
	23:16	L3 LRA1 Min		
		Default Value:	0010000b	
		Access:	R/W	
		L3 LRA1 Min		
		Format: U6		
		Minimum value of programn	nable LRA1	
	15:8	L3 LRA0 Max		1
		Default Value:	00011111b	
		Access:	R/W	
		L3 LRA0 Max		
		Format: U6		
		Maximum value of programr	nable LRA0	
	7:0	L3 LRA0 Min		
		Default Value:	0000000b	
		Access:	R/W	
		L3 LRA0 Min		
		Format: U6		
		Minimum value of programn	nable LRA1	





### L3\_LRA\_1 - L3 LRA 1

Register Space:	MMIC	D: 0/2/0	
Default Value:	0x090	0FF40	
Size (in bits):	32		
Address:	04040	)h-04043h	
L3 LRA 1			
DWord	Bit		Description
0	31:30	Reserved	
		Default Value:	00b
		Access:	RO
		Reserved Bits	
-	29:28	DC	
		Default Value:	00b
		Access:	R/W
_	27:26	TEXTURE         Default Value:         Access:         Which LRA should TEXTURE	10b R/W use
-	25:24	L3 Default Value:	01b
		Access: Which LRA should L3 use	R/W
-	23:16	Reserved	
		Default Value:	0000000b
		Access:	RO
		Reserved Bits	



### L3\_LRA\_1 - L3 LRA 1

15:8	L3 LRA2 Max	
	Default Value:	11111111b
	Access:	R/W
	L3 LRA2 Max	
	Format: U6	
	Maximum value of programmable LRA2	
7:0	L3 LRA2 Min	
	Default Value:	0100000b
	Access:	R/W
	L3 LRA2 Min	
	Format: U6	
	Minimum value of programmable LRA2	



### **3DPRIM\_BASE\_VERTEX - Load Indirect Base Vertex**

Register	Space	e: MMIO: 0/2/0	
Source:		RenderCS	
Default Value:		0x0000000	
Access:		R/W	
Size (in b	oits):	32	
Address	:	02440h-02443h	
DWord	Bit		Description
0	31:0	Base Vertex	
		Format:	S31
		This register is used to store the Enable is set.	Base Vertex of the 3D_PRIMITIVE command when Load Indirect



3DP	RIN	//_INSTANCE_COUNT - Load Indirect Instance Count
Register	Space	e: MMIO: 0/2/0
Source:		RenderCS
Default \	/alue:	0x0000000
Access:		R/W
Size (in b	oits):	32
Address:		02438h-0243Bh
DWord	Bit	Description
0	31:0	Instance Count
		This register is used to store the Instance Count of the 3D_PRIMITIVE command when Load Indirect Enable is set.



### **3DPRIM\_START\_INSTANCE** - Load Indirect Start Instance

Space	e: MMIO: 0/2/0		
	RenderCS		
Value:	0x0000000		
	R/W		
oits):	32		
	0243Ch-0243Fh		
Bit		Description	
31:0	Start Vertex		
	Format:	U32	
	This register is used to store the Star Enable is set.	Instance of the 3D_PRIMITIVE command when Load Inc	lirect
	/alue: bits): Bit	RenderCS Value: 0x0000000 R/W oits): 32 0243Ch-0243Fh Bit 31:0 Start Vertex Format: This register is used to store the Start	RenderCS         /alue:       0x0000000         R/W         bits):       32         0243Ch-0243Fh         Bit       Description         31:0       Start Vertex         Format:       U32         This register is used to store the Start Instance of the 3D_PRIMITIVE command when Load Income



# 3DPRIM\_START\_VERTEX - Load Indirect Start Vertex

Register	Space	e: MMIO: 0/2/0			
Source:		RenderCS			
Default Value:		0x0000000	0x0000000		
Access:		R/W			
Size (in b	oits):	32			
Address:		02430h-02433h			
DWord	Bit		Description		
0	31:0	Start Vertex			
		Format:	U32		
		This register is used to store the St	art Vertex of the 3D_PRIMITIVE command when Load Indirect		
		Enable is set.			



### **3DPRIM\_VERTEX\_COUNT - Load Indirect Vertex Count**

Register	Space	:: MMIO: 0/2/0	
Source:		RenderCS	
Default Value:		0x0000000	
Access:		R/W	
Size (in b	oits):	32	
Address:		02434h-02437h	
DWord	Bit		Description
0	31:0	Vertex Count	
		Format:	U32
		This register is used to store the Vertex	Count of the 3D_PRIMITIVE command when Load Indirect
		Enable is set.	



### **ERROR - Main Graphic Arbiter Error Report**

Register	Space:	: MMIO: 0/2/0	
Default	Value:	0x0000000	
Size (in	bits):	32	
Address	:	040A0h-040A3h	
This reg	gister is	s used to report different error condition	s. Error bits are writable.
DWord	Bit	D	Description
0	31:16	Reserved	
		Default Value:	0000h
		Access:	RO
		Reserved Bits	
	15	Reserved Error Bits 15	
		Default Value:	0b
		Access:	R/W
	14	Reserved Error bits (Future expansion) Reserved Error Bits 14	
		Default Value:	0b
		Access:	R/W
		Reserved Error bits (Future expansion)	
	13	Reserved Error Bits 13	
		Default Value:	0b
		Access:	R/W
		Reserved Error bits (Future expansion)	
	12	Reserved Error Bits 12	
	12	Reserved Error Bits 12 Default Value:	0b
	12		0b R/W



## **ERROR - Main Graphic Arbiter Error Report**

	Reserved Error Bits 11				
	Default Value:	0b			
	Access:	R/W			
	Reserved Error bits (Future expansion)				
10	Reserved Error Bits 10				
	Default Value:	0b			
	Access:	R/W			
	Reserved Error bits (Future expansion)				
9	Reserved Error Bits 9				
	Default Value:	0b			
	Access:	R/W			
	Reserved Error bits (Future expansion)				
8	Unloaded PD Error				
	Default Value:	0b			
	Access:	R/W			
	Access: Unloaded PD error	R/W			
	Unloaded PD error				
7	Unloaded PD error The Cache Line containing a PD entry be PD load cycle. Reserved Error Bits 7	ing accessed, was marked as invalid in the			
7	Unloaded PD error The Cache Line containing a PD entry be PD load cycle.	ing accessed, was marked as invalid in the			
7	Unloaded PD error The Cache Line containing a PD entry be PD load cycle. Reserved Error Bits 7 Default Value: Access:	ing accessed, was marked as invalid in the			
7	Unloaded PD error The Cache Line containing a PD entry be PD load cycle. Reserved Error Bits 7 Default Value:	ing accessed, was marked as invalid in the			
7	Unloaded PD error The Cache Line containing a PD entry be PD load cycle. Reserved Error Bits 7 Default Value: Access:	ing accessed, was marked as invalid in the			
	Unloaded PD error The Cache Line containing a PD entry be PD load cycle. Reserved Error Bits 7 Default Value: Access: Reserved Error bits (Future expansion)	ing accessed, was marked as invalid in the			



ERROR - Main Graphic Arbiter Error Report	
---	--

	3	Reserved	
-	2	Invalid Page Directory Entry Error	
		Default Value:	0b
		Access:	R/W
		Invalid Page Directory entry error	
		PD entry's valid bit is 0	
-	1	Reserved	
	0	TLB Page Fault Error	
		Default Value:	0b
		Access:	R/W
		TLB Page Fault error	
		A TLB Page's GTT translation generated a page fault (GTT entry	not valid)



GFX_	ARE	B_ERROR_RPT - Main Graphic Arbiter Error Report Register
Register S	pace:	MMIO: 0/2/0
Source:		RenderCS
Default Va	lue:	0x0000000
Access:		R/W
Size (in bi	ts):	32
Trusted Ty	/pe:	1
Address:		040A0h
This regist	er is us	ed to report error conditions. Error bits are writable.
DWord	Bit	Description
0	31:16	Reserved
		Format: MBZ
	15:9	Reserved
	8	<b>Unloaded PD Error</b> The Cache Line containing a PD entry being accessed was marked as invalid in the last PD load cycle.
	7	Reserved
	6	Reserved
	5	Reserved
	4	Reserved
	3	Hardware Status Page Fault Error HWSP's GTT translation generated a page fault (GTT entry not valid).
	2	Invalid Page Directory entry error PD entry's valid bit is 0.
	1	<b>Context Page Fault Error</b> A Context Page's GTT translation generated a page fault (GTT entry not valid).
	0	<b>TLB Page Fault Error</b> A TLB Page's GTT translation generated a page fault (GTT entry not valid).



Register	Space	MMIO: 0/2/0			
Default \		0x0000000			
Size (in l	oits):	32			
Address	:	04034h-04037h			
GFX_PE	ND_TL	.B_0 - Max Outstanding Pending TLB Re	quests 0		
DWord	Bit		Description		
0	31	TEX Limit Enable Bit			
		Default Value:		0b	
		Access:		R/W	
		TEX Limit Enable bit			
		This bit is used to enable the pending <sup>-</sup> Cache			
		Cache When set, the number of internal pend	ing read requests which r		
	30	Cache When set, the number of internal pend not exceed the programmed counter v	ing read requests which r		
	30	Cache When set, the number of internal pend	ing read requests which r		
	30	Cache When set, the number of internal pend not exceed the programmed counter v <b>Reserved</b>	ing read requests which r	equire a TLB rea	
	30	Cache When set, the number of internal pend not exceed the programmed counter v <b>Reserved</b> Default Value:	ing read requests which r	equire a TLB rea	
	30	Cache When set, the number of internal pend not exceed the programmed counter v <b>Reserved</b> Default Value: Access:	ing read requests which r	equire a TLB rea	
		Cache When set, the number of internal pend not exceed the programmed counter v Reserved Default Value: Access: Reserved Format: MBZ TEX TLB Limit Count	ling read requests which r alue.	equire a TLB rea	
		Cache When set, the number of internal pend not exceed the programmed counter v Reserved Default Value: Access: Reserved Format: MBZ TEX TLB Limit Count Default Value:	ling read requests which r alue.	equire a TLB rea	
		Cache When set, the number of internal pend not exceed the programmed counter v Reserved Default Value: Access: Reserved Format: MBZ TEX TLB Limit Count Default Value: Access:	ling read requests which r alue.	equire a TLB rea	
		Cache When set, the number of internal pend not exceed the programmed counter v Reserved Default Value: Access: Reserved Format: MBZ TEX TLB Limit Count Default Value:	ling read requests which r alue.	equire a TLB rea	
		Cache When set, the number of internal pend not exceed the programmed counter v Reserved Default Value: Access: Reserved Format: MBZ TEX TLB Limit Count Default Value: Access:	ling read requests which r alue.	equire a TLB rea	
		Cache When set, the number of internal pend not exceed the programmed counter v Reserved Default Value: Access: Reserved Format: MBZ TEX TLB Limit Count Default Value: Access: TEX TLB Limit Count	ling read requests which r alue. 000000b R/W	equire a TLB rea	ad will
		Cache When set, the number of internal pend not exceed the programmed counter v Reserved Default Value: Access: Reserved Format: MBZ TEX TLB Limit Count Default Value: Access: TEX TLB Limit Count Format: U6 This is the MAX number of Allowed inter	ling read requests which r alue. 000000b R/W	equire a TLB rea	ad will
	29:24	Cache When set, the number of internal pend not exceed the programmed counter v Reserved Default Value: Access: Reserved Format: MBZ TEX TLB Limit Count Default Value: Access: TEX TLB Limit Count Format: U6 This is the MAX number of Allowed inter read	ling read requests which r alue. 000000b R/W	equire a TLB rea	ad will



	0		
	DC Limit Enable bit		
	Format: U1		
	This bit is used to enable the pending TLE Instruction Cache.	3 requests limitation fu	nction for the
	When set, the number of internal pending not exceed the programmed counter values		equire a TLB read v
22	Reserved		
	Default Value:		0b
	Access:		RO
	Reserved		
	Format: MBZ		
21:16	DC TLB Limit Count		
	Default Value:	000000b	
	Access:	R/W	
	DC TLB Limit Count Format: U6		
	This is the MAX number of Allowed interr read.	nal pending read reque	sts which require a
15	VF Limit Enable bit		01-
	Default Value:		0b
	Access: VF Limit Enable bit		R/W
	Format: U1		
	This bit is used to enable the pending TLE Fetch	3 requests limitation fu	nction for the Verte
		g read requests which r	equire a TLB read
	When set, the number of internal pending not exceed the programmed counter value	Je.	
14		Je.	
14	not exceed the programmed counter value	Je.	0b



	0		
	Reserved		
	Format: MBZ		
13:8	VF TLB Limit Count		
	Default Value:	00000b	
	Access:	R/W	
	VF TLB Limit Count		
	Format: U6		
	This is the MAX number of Allowed internal pe	ending read reque	sts which requi
	read.		
7	VMC Limit Enable bit		1
	Default Value:		0b
	Access:		R/W
	Format: U1 This bit is used to enable the pending TLB req Motion Compensation . When set, the numbe	r of internal pendi	ng read reques
			Ie
	require a TLB read will not exceed the program	nmed counter valu	
6	require a TLB read will not exceed the program	nmed counter valu	
6	require a TLB read will not exceed the program <b>Reserved</b> Default Value:	nmed counter vail	0b
6	require a TLB read will not exceed the program <b>Reserved</b> Default Value: Access:	nmed counter vail	
6	require a TLB read will not exceed the program <b>Reserved</b> Default Value:	nmed counter valu	0b
6	require a TLB read will not exceed the program <b>Reserved</b> Default Value: Access:		0b
-	require a TLB read will not exceed the program <b>Reserved</b> Default Value: Access: Reserved		0b
-	require a TLB read will not exceed the program <b>Reserved</b> Default Value: Access: Reserved Format: MBZ	nmed counter valu	0b
6	require a TLB read will not exceed the program Reserved Default Value: Access: Reserved Format: MBZ VMC TLB Limit Count		0b
-	require a TLB read will not exceed the program Reserved Default Value: Access: Reserved Format: MBZ VMC TLB Limit Count Default Value:	00000b	0b
-	require a TLB read will not exceed the program Reserved Default Value: Access: Reserved Format: MBZ VMC TLB Limit Count Default Value: Access:	00000b	0b



#### GFX\_PEND\_TLB\_0 - Max Outstanding Pending TLB Requests 0 **Register Space:** MMIO: 0/2/0 Source: RenderCS Default Value: 0x00000000 Access: R/W Size (in bits): 32 Trusted Type: 1 Address: 04034h-04037h **DWord** Bit Description 0 31 **TEX Limit Enable bit** U1 Format: This bit is used to enable the pending TLB requests limitation function for the Texture Cache. When set, the number of internal pending read requests which require a TLB read will not exceed the programmed counter value. Reserved 30 Format: MBZ 29:24 TEX TLB Limit Count U6 Format: This is the MAX number of Allowed internal pending read requests which require a TLB read. **ISC Limit Enable bit** 23 U1 Format: This bit is used to enable the pending TLB requests limitation function for the Instruction Cache. When set, the number of internal pending read requests which require a TLB read will not exceed the programmed counter value. 22 Reserved Format: MBZ 21:16 **ISC TLB Limit Count** Format: U6 This is the MAX number of Allowed internal pending read requests which require a TLB read. VF Limit Enable bit 15 U1 Format: This bit is used to enable the pending TLB requests limitation function for the Vertex Fetch. When set, the number of internal pending read requests which require a TLB read will not exceed the programmed counter value.



14	Reserved			
	Format:	Ν	1BZ	
13:8	VF TLB Limit Count			
	Format:		U6	
	This is the MAX number of Allowed internal pending read requests which require a TLB read			
7	CS Limit Enable bit			
	Format:		U1	
	This bit is used to enable the pending T Streamer. When set, the number of inter not exceed the programmed counter val	nal pending read re		
6	Reserved			
	Format:	Ν	1BZ	
5:0	CS TLB Limit Count			
	Format:		U6	
	This is the MAX number of Allowed inter	rnal pending read re	quests which requir	e a TLB



Γ

GFX_	PEN	ND_TLB_1 - Max Out	standing Pending	TLB Request
Register Default \ Size (in b	/alue:	MMIO: 0/2/0 0x0000000 32		
Address:		04038h-0403Bh		
GFX_PEI	ND_TL	B_1 - Max Outstanding pending TL	B requests 1	
DWord	Bit		Description	
0	31	SOL Limit Enable bit		
		Default Value:		0b
		Access:		R/W
		SOL Limit Enable bit		
		Format: U1		
		This bit is used to enable the pend When set, the number of internal not exceed the programmed cour	pending read requests which re	
-	30	Reserved		
		Default Value:		0b
		Access:		RO
		Reserved		
-		Format: MBZ		
	29:24	SOL TLB Limit Count		
		Default Value:	000000b	
		Access:	R/W	
		SOL TLB Limit Count		
		Format: U6		
		This is the MAX number of Allowe read.	d internal pending read reques	sts which require a TLB
	23	L3 Limit Enable bit		
		Default Value:		0b
		Access:		R/W
		L3 Limit Enable bit		



X_PE	ND_TLB_1 - Max Outs	standing Pending	g TLB Reques
	Format: U1 This bit is used to enable the pendi set, the number of internal pending exceed the programmed counter v	g read requests which require	
22	Reserved		
	Default Value:		0b
	Access:		RO
	Reserved		
	Format: MBZ		
21:16	L3 TLB Limit Count		
	Default Value:	00000b	
	Access:	R/W	
	Format: U6 This is the MAX number of Allowed read.	l internal pending read reque	ests which require a TL
15	RCZ Limit Enable bit		
	Default Value:		0b
	Access:		R/W
	<ul> <li>RCZ Limit Enable bit</li> <li>Format: U1</li> <li>This bit is used to enable the pending TLB requests limitation function for the Render Depth Cache</li> <li>When set, the number of internal pending read requests which require a TLB read will not exceed the programmed counter value.</li> </ul>		
14	Reserved		
	Default Value:		0b
	Access:		RO
	Reserved		



13:8	RCZ TLB Limit Count		
	Default Value:	000000b	
	Access:	R/W	
	RCZ TLB Limit Count		
	Format: U6		
	This is the MAX number of Allowe read.	ed internal pending read requ	uests which requi
7	RCC Limit Enable bit		
	Default Value:		0b
	Access:		R/W
	RCC Limit Enable bit		
	Format: U1		
6	Format: U1 This bit is used to enable the pend Color Cache. When set, the numb TLB read will not exceed the prog	per of internal pending read i	
6	Format: U1 This bit is used to enable the pene Color Cache. When set, the numb TLB read will not exceed the prog <b>Reserved</b>	per of internal pending read i	requests which re
6	Format: U1 This bit is used to enable the pend Color Cache. When set, the numb TLB read will not exceed the prog	per of internal pending read i	
6	Format: U1 This bit is used to enable the pene Color Cache. When set, the numb TLB read will not exceed the prog Reserved Default Value:	per of internal pending read i	requests which re
6	Format: U1 This bit is used to enable the pene Color Cache. When set, the numb TLB read will not exceed the prog <b>Reserved</b> Default Value: Access:	per of internal pending read i	requests which re
6	Format: U1 This bit is used to enable the pene Color Cache. When set, the numb TLB read will not exceed the prog <b>Reserved</b> Default Value: Access: Reserved	per of internal pending read i rammed counter value.	0b RO
	Format: U1 This bit is used to enable the pene Color Cache. When set, the numb TLB read will not exceed the prog <b>Reserved</b> Default Value: Access: Reserved Format: MBZ	per of internal pending read i	0b RO
	Format: U1 This bit is used to enable the pene Color Cache. When set, the numb TLB read will not exceed the prog <b>Reserved</b> Default Value: Access: Reserved Format: MBZ <b>RCC TLB Limit Count</b> Default Value: Access:	per of internal pending read i rammed counter value.	0b RO
	Format: U1 This bit is used to enable the pene Color Cache. When set, the numb TLB read will not exceed the prog <b>Reserved</b> Default Value: Access: Reserved Format: MBZ <b>RCC TLB Limit Count</b> Default Value:	per of internal pending read rammed counter value.	0b RO



GFX	PEN	ND TLB 1 - Max Out	standing Pending TLB Requests
<u> </u>			1
Register	Space:	MMIO: 0/2/0	
Source:		RenderCS	
Default \	/alue:	0x0000000	
Access:		R/W	
Size (in b	oits):	32	
Trusted <sup>-</sup>	Гуре:	1	
Address:		04038h-0403Bh	
DWord	Bit		Description
0	31:16	Reserved	
		Format:	MBZ
	15	RCZ Limit Enable bit	
		Format:	U1
		•	g TLB requests limitation function for the Render Depth rnal pending read requests which require a TLB read will not ue.
	14	Reserved	
		Format:	MBZ
	13:8	RCZ TLB Limit Count	
		Format:	U6
			g TLB requests limitation function for the Render Color mal pending read requests which require a TLB read will not ue.
	7	RCC Limit Enable bit	
		Format:	U1
		•	g TLB requests limitation function for the Render Color nal pending read requests which require a TLB read will not ue.
	6	Reserved	
		Format:	MBZ
	5:0	RCC TLB Limit Count	
		Format:	U6
		This is the MAX number of Allowed ir	nternal pending read requests which require a TLB read.



# MEDIA\_MAX\_REQ\_COUNT - MAX Requests Allowed - CASC

Register Space		MMIO: 0/2/0				
Default \	Value:	0x10202020				
Size (in k	bits):	32				
Address: 04070h-04073h						
Prograr	nmabl	e Request Count - CASC				
DWord Bit			Description			
0	31:24	GFX Max Request Limit Count	000	10000		
		Default Value:		10000b		
		Access:	R/W			
		This is the MAX number of Allowed Requests Count - These counters keep track of the accepted requests from each engine . Requests are counted, regardless of kind of cycle (Miss/Hit/Present)				
		Minimum count value must be =				
	23:16	MFX/BLT Max Request Limit Coun				
		Default Value:	001	00000b		
		Access: This is the MAX number of Allow accepted requests from each eng	•	- These count		
		This is the MAX number of Allow accepted requests from each eng (Miss/Hit/Present)	ed Requests Count - jine . Requests are c	- These count	•	
	15.14	This is the MAX number of Allow accepted requests from each eng (Miss/Hit/Present) Minimum count value must be =	ed Requests Count - jine . Requests are c	- These count	•	
	15:14	This is the MAX number of Allow accepted requests from each eng (Miss/Hit/Present) Minimum count value must be = <b>Reserved</b>	ed Requests Count - jine . Requests are c	- These count	rdless of kind of cycle	
	15:14	This is the MAX number of Allow accepted requests from each eng (Miss/Hit/Present) Minimum count value must be =	ed Requests Count - jine . Requests are c	- These count	rdless of kind of cycle	
	15:14	This is the MAX number of Allow accepted requests from each eng (Miss/Hit/Present) Minimum count value must be = Reserved Default Value:	ed Requests Count - jine . Requests are c	- These count	rdless of kind of cycle	
		This is the MAX number of Allow accepted requests from each eng (Miss/Hit/Present) Minimum count value must be = <b>Reserved</b> Default Value: Access: Reserved Bits	ed Requests Count - jine . Requests are c	- These count	rdless of kind of cycle	
	15:14	This is the MAX number of Allow accepted requests from each eng (Miss/Hit/Present) Minimum count value must be = <b>Reserved</b> Default Value: Access:	ed Requests Count - jine . Requests are c	- These count	rdless of kind of cycle	
		This is the MAX number of Allow accepted requests from each eng (Miss/Hit/Present) Minimum count value must be = <b>Reserved</b> Default Value: Access: Reserved Bits <b>VLF Max Request Limit Count</b> Default Value:	ed Requests Count - jine . Requests are c	- These count ounted, rega	rdless of kind of cycle	
		This is the MAX number of Allow accepted requests from each eng (Miss/Hit/Present) Minimum count value must be = <b>Reserved</b> Default Value: Access: Reserved Bits <b>VLF Max Request Limit Count</b>	ed Requests Count - jine . Requests are c 1 ed Requests Count -	- These count ounted, regar 100000b R/W - These count	rdless of kind of cycle 00b RO ters keep track of the	
		This is the MAX number of Allow accepted requests from each eng (Miss/Hit/Present) Minimum count value must be = <b>Reserved</b> Default Value: Access: Reserved Bits <b>VLF Max Request Limit Count</b> Default Value: Access: This is the MAX number of Allow accepted requests from each clie	ed Requests Count - jine . Requests are c 1 ed Requests Count - nt. Requests are cou	- These count ounted, regar 100000b R/W - These count	rdless of kind of cycle 00b RO ters keep track of the	
		This is the MAX number of Allow accepted requests from each eng (Miss/Hit/Present) Minimum count value must be = <b>Reserved</b> Default Value: Access: Reserved Bits <b>VLF Max Request Limit Count</b> Default Value: Access: This is the MAX number of Allow accepted requests from each clie (Miss/Hit/Present )	ed Requests Count - jine . Requests are c 1 ed Requests Count - nt. Requests are cou	- These count ounted, regar 100000b R/W - These count	rdless of kind of cycle 00b RO ters keep track of the	
	13:8	This is the MAX number of Allow accepted requests from each eng (Miss/Hit/Present) Minimum count value must be = <b>Reserved</b> Default Value: Access: Reserved Bits <b>VLF Max Request Limit Count</b> Default Value: Access: This is the MAX number of Allow accepted requests from each clie (Miss/Hit/Present ) Minimum count value must be =	ed Requests Count - jine . Requests are c 1 ed Requests Count - nt. Requests are cou	- These count ounted, regar 100000b R/W - These count	rdless of kind of cycle 00b RO ters keep track of the	
	13:8	This is the MAX number of Allow accepted requests from each eng (Miss/Hit/Present) Minimum count value must be = <b>Reserved</b> Default Value: Access: Reserved Bits <b>VLF Max Request Limit Count</b> Default Value: Access: This is the MAX number of Allow accepted requests from each clie (Miss/Hit/Present ) Minimum count value must be = <b>Reserved</b>	ed Requests Count - jine . Requests are c 1 ed Requests Count - nt. Requests are cou	- These count ounted, regar 100000b R/W - These count	rdless of kind of cycle 00b RO ters keep track of the less of kind of cycle	



GFX_PE	ND_TLB_1 - Max Outstanding P 1	ending TLB Requests
5:0	Format: MBZ CASC Max Request Limit Count	
5.0	Default Value:	100000b
	Access:	R/W
	This is the MAX number of Allowed Requests Count accepted requests from each client. Requests are cou (Miss/Hit/Present )	•
	Minimum count value must be = 1	



### **GFX\_MAX\_REQ\_COUNT** - **MAX** Requests Allowed - **GAM**

Register Sp	-	MMIO: 0/2/0					
Default Va	alue:	0x43F20101					
Size (in bit	ts):	32					
Address:		04074h-04077h					
Programr	mabl	e Request Count - GAM					
DWord	Bit	Description					
		GAP Writes Max Request Limit Count					
		Default Value:	010000b				
		Access:	R/W				
		This is the MAX number of Allowed Write Requests Count - These counters keep track of the accepted write requests from all GAP clients (RCZ, HiZ,Stc, RCC, L3).					
		Minimum count value must be = 1					
2	25:20	CVS Max Request Limit Count					
		Default Value:	111111b				
		Access:	R/W Rrs Count - These counters keep track of the				
		accepted requests from each client. Reques (Miss/Hit/Present ) Minimum count value must be = 1					
	19	Reserved					
		Default Value:	Ob				
		Access:	RO				
		Reserved Format: MBZ					
1	.8:13	L3 Max Request Limit Count					
		Default Value:	010000b				
		Access:	R/W				
		This is the MAX number of Allowed Requests Count - These counters keep track of t accepted requests from each client. Requests are counted, regardless of kind of cycl (Miss/Hit/Present)					
		· · · ·	is are counted, reguratess of kind of cycle				



### **GFX\_MAX\_REQ\_COUNT** - **MAX** Requests Allowed - **GAM**

	Default Value:		0b
	Access:		RO
	Reserved		
	Format: MBZ		
11:6	Z Request Limit Count		
	Default Value:	000100b	
	Access:	R/W	
	This is the MAX number of Allowed Requests Count - accepted requests from each client. Requests are cou (Miss/Hit/Present) Minimum count value must be = 1		•
5:0	RCC Request Limit Count		
	Default Value:	000001b	
	Access:	R/W	
	This is the MAX number of Allowed Requests Count - These counters keep track of the accepted requests from each client. Requests are counted, regardless of kind of cycle (Miss/Hit/Present)		
	Minimum count value must be = 1		



# **GAC\_ERROR - Media Arbiter Error Report Register**

Register	Space:	MMIO: 0/2/0
Source:		VideoCS
Default \	ault Value: 0x0000000	
Access:	Access: R/W	
Size (in b	oits):	32
Trusted <sup>-</sup>	Гуре:	1
Address:		140A0h
These re	gisters	are directly mapped for the Error Reporting bits.
DWord	Bit	Description
0	0 31:11 <b>Reserved/ECO</b>	
	10	Reserved
	9	Reserved
	8	<b>Unloaded PD error</b> The Cache Line containing a PD entry being accessed was marked as invalid in the last PD load cycle.
	7	Reserved
	6	Reserved
	5	Reserved
	4	Reserved
	3	Reserved
	2	Reserved
	1	<b>Context Page Fault Error</b> A Context Page's GTT translation generated a page fault (GTT entry not valid)
	0	<b>TLB Page Fault Error</b> A TLB Page's GTT translation generated a page fault (GTT entry not valid)



#### **MEDIA\_ENG\_FR - Media Engine Fault Register**

Register	Space	MMIO: 0/2/0					
Default Value:		0x0000000	0x0000000				
Size (in bits):		32					
Address:		04194h-04197h					
Media E	ingine	Fault Register					
DWord	Bit		Description				
0 3	31:12	Virtual Address of Fault					
		Default Value:	C	0000h			
		Access:	F	R/W			
	11	This value is locked and not updated register is cleared by SW	on subsequent fault	5, until the valid bit	of this		
	ΤT	Default Value:		0b			
		Access:		R/W			
		This bit indicates if the valid bit happ GGTT					
		This value is locked and not updated register is cleared by SW	on subsequent fault	s, until the valid bit	of this		
	10:3	SRCID of Fault					
		Default Value:		00h			
		Access:		R/W			
		This is the Source ID of the unit that r fault for this engine.	requested the cycle t	hat generated the	First Page		

This value is locked and not updated on subsequent faults, until the valid bit of this register is cleared by SW

2:1	Fault Type	
	Default Value:	00b
	Access:	R/W
	Type of Fault recorded:	
	00 - Page Fault.	
	01 - Invalid PD Fault	



# MEDIA\_ENG\_FR - Media Engine Fault Register

10	- Unloaded PD Fault			
11	11 - Invalid and Unloaded PD fault			
	s value is locked and not updated on subsequent faults, until ister is cleared by SW	the valid bit of this		
0 Vali	id Bit			
De	fault Value:	0b		
Ace	cess:	R/W		
This	s bit indicates that the first fault for this engine has been reco	orded. It can only be		
clea	ared by SW, which will also clear the other fields.			



AVC_C	ABAC_INSERTION_COUNT -	
MFC_AVG	CABAC_INSERTION_COUNT	

	MMIO: 0/2/0 VideoCS 0x0000000 RO
	0x0000000
-).	RO
-).	
5).	32
pe:	1
	124ACh
	e count in bytes of CABAC ZERO_WORD insertion. It is primarily provided for statistical
ening.	
ord Bit Description	
:0 MFC AVC Cabac Insertion Count	
Total number of bytes in the bitstream output before for the CABAC zero word insertion. This count is updated each time when the insertion count is incremented.	
	r stores th ring. :0 MFC A Total n



#### MFC\_VIN\_AVD\_ERROR\_CNTR - MFC\_AVC Bitstream Decoding Front-End Parsing Logic Error Counter

Register Space:	MMIO: 0/2/0			
Source:	VideoCS			
Default Value:	0x0000000			
Access:	R/W			
Size (in bits):	32			
Trusted Type:	1			
Address:	12804h			
	DWord	Bit	Des	scription
	0	31:0	Reserved	
			Format:	MBZ
	avd_error_flagsR[31:0]			•



### MFC\_IMAGE\_STATUS\_CONTROL - MFC Image Status Control

Register Sp	ace:	MMIO: 0/2/0					
Source:		VideoCS					
Default Value:		0x0000000					
Access:		RO					
Size (in bits):		32					
Trusted Type:		1					
Address:		124B8h					
This registe	r stores t	he suggested data for next frame in multi-pass	5.				
DWord	Bit	Desci	ription				
0	31:24	Cumulative slice delta QP					
	23:16	<b>QP Value</b> suggested slice QP delta value for frame level Rate control. This value can be +					
	15	<b>QP-Polarity Change</b> Cumulative slice delta QP polarity change.					
	14:13	<b>Num-Pass Polarity Change</b> Number of passes after cumulative slice delta QP polarity changes.					
	12	Reserved					
		Format:	MBZ				
	11:8	Total Num-Pass					
	7:4	Reserved					
		Format:	MBZ				
	3	Reserved					
		Format:	MBZ				
	2	<b>Panic</b> Panic triggered to avoid too big packed file.	I				
	1	Frame Bit Count Frame Bit count over-run/under-run flag					
	0	Max Conformance Flag Max Macroblock conformance flag or Frame I	Bit count over-run/under-run				



# MFC\_IMAGE\_STATUS\_MASK - MFC Image Status Mask

Register Space:	MMIO: 0/2/0					
Source:	VideoCS					
Default Value:	0x00000000					
Access:	RO					
Size (in bits):	32					
Trusted Type:	1					
Address:	124B4h					
This register stores the	This register stores the image status(flags).					
DWord	Bit	Description				
0	31:0	Control Mask				
		Control Mask for dynamic frame repeat.				



		MFC_QUP_CT - MFC	QP Status	s C	ount		
Register Space: MMIO: 0/2/0							
Source:		VideoCS					
Default <b>'</b>	Value:	0x0000000					
Access:		RO					
Size (in l	oits):	32					
Trusted	Type:	1					
Address	:	124BCh					
This regi	ister st	pres the suggested QP COUNTS in multi-pas	S.				
DWord	Bit		Description				
0	31:24	Cumulative QP Adjust					
		Format:			U8		
	Cumulative QP adjustment after multiple passes. If there is no need to multi-pass, this value would be zero. (This is in sign magnitude form).						
	23:0	23:0 Cumulative QP					
		Format:	L	J24			
		Cumulative QP for all MB of a Frame ( Can b	be used for comput	ing a	verage QP).		



#### **MFD\_ERROR\_STATUS - MFD Error Status**

Register Space: MMIO: 0/2/0			: 0/2/0				
Source:		Video	VideoCS				
Default '	Value:	0x000	00000				
Access:		RO					
Size (in l	oits):	32					
Trusted	Type:	1					
Address	:	12400	h				
_	jister is	not part of h	-	t reports by the bit-stream decoder. and restore. Driver should read the content prior to starting a Description			
0	31:16	Number of Error Events					
		Exists If:		//JPEG == True			
		Format:		U16			
		This 16-bit field indicates the number of error events detected during decoding the current frame. This field is clear at the start of decoding a new frame.					
	31:16	1:16 Number of MB Concealment					
		Exists If:	//AVC CAVLC, AVC	CABAC, VC1 and MPEG2 == True			
		Format:	U16				
			eld indicates the num ding a new frame.	ber of MB is concealmed by hardware. This field is clear at the			



MFD	_PICTU	RE_PARA	M - MFD P	icture Par	ameter
Register Space:	MMIO: 0/2	/0			
Source:	VideoCS				
Default Value:	0x0000000	0			
Access:	R/W				
Size (in bits):	32				
Trusted Type:	1				
Address:	12420h				
DWord		Bit		Description	
0		31:0	Reserved		
			Format:		MBZ



# MFX\_LAT\_CT1 - MFX\_Memory\_Latency\_Count1

Register	Space	: MMIO: 0/2/0
Source:		VideoCS
Default \	Value:	0x0000000
Access:		RO
Size (in b	oits):	32
Trusted	Type:	1
Address:		12470h
0		ores the max and min memory latency counts reported on reference read requests. not part of hardware context save and restore.
DWord	Bit	Description
0	31:24	Max Request Count This field indicates the maximum number of requests allowed by the memory sub-system channel.
This This		<b>Current Request Count</b> This field indicates the number of requests currently outstanding in the memory sub-system. This field should report with a value of zero at the end of frame; otherwise the motion compensation engine is most likely hung waiting for read data to be returned from sub-system.
15:8 <b>MFX Reference picture read request - Max Latency Count in 8xMedia clock o</b> This field reports the maximum memory latency count on all reference reads requirement motion compensation engine.		
	7:0	<b>MFX Reference picture read request - Min Latency Count in 8xMedia clock cycles</b> This field reports the minimum memory latency count on all reference reads requested by the motion compensation engine.



### MFX\_SE-BIN\_CT - MFX Frame BitStream SE/BIN Count

Register Space:		e: MMIO: 0/2/0		
Source:		VideoCS		
Default V	'alue:	0x0000000		
Access:		RO		
Size (in b	its):	32		
Trusted T	ype:	1		
Address:		1246Ch		
This regis	ster s	tores the number of BINs (AVC CABAC) and SEs (CAVLD, VLD) decoded in a frame.		
This regi	ster i	s not part of hardware context save and restore.		
DWord	Bit	Description		
0	31:0	X Frame Bit-stream SE/BIN Count		
		Total number of BINs/SEs decoded in current frame. This number is used with frame performance		
		count to derive Bin/clk or SE/clk.		



#### MFX\_MB\_COUNT - MFX Frame Macroblock Count

Register Space:		e: MMIO: 0/2/0		
Source:		VideoCS		
Default Va	alue:	0x0000000		
Access:		RO		
Size (in bi	ts):	32		
Trusted T	ype:	1		
Address:		12468h		
This regis <sup>-</sup>	ter st	tores the number of Macro-blocks decoded/encoded in current frame.		
This regis	ster i	s not part of hardware context save and restore.		
DWord	Bit	Description		
0 3	31:0	MFX Frame Macro-block Count		
		Total number of Macro-block decoded/encoded in current frame. This number is used with frame		
		performance count to derive clk/mb.		



# MFX\_MISS\_CT - MFX Frame Motion Comp Miss Count

Register Space:		MMIO: 0/2/0			
Source:		VideoCS			
Default '	Value:	0x0000000			
Access:		RO			
Size (in l	oits):	32			
Trusted	Туре:	1			
Address		12488h			
This register stores the total number of cacheline hits occurred in the motion compensation cache per frame					
		not part of hardware context save and restore.			
DWord	Bit	Description			
0	31:16	Reserved			
Format: MBZ		Format: MBZ			
	15:0	MFX Frame Motion Comp cache miss Count Total number of CL misses occurred in the 12KB cache of the motion compensation engine per frame. This number is used along with MFX Frame Motion Comp Read Count to derive motion comp cache miss/hit ratio.			



# MFX\_READ\_CT - MFX Frame Motion Comp Read Count

Register Space:		MMIO: 0/2/0			
Source:		VideoCS			
Default V	alue:	0x0000000			
Access:		RO			
Size (in b	its):	32			
Trusted T	уре:	1			
Address:		12484h			
This register stores the total number of reference picture read requests made by the Motion Compensation engine per frame. This register is not part of hardware context save and restore.			de by the Motion Compensation		
DWord	Bit	Description			
0	31:20	Reserved			
		Format: MBZ			
	19:0	<b>MFX Frame Motion Comp CL read request Count</b> Total number of reference picture read requests by the motion compensation engine per frame.			



MFX_F	RAN	IE_PERFORMANCE_CT - MFX Frame Performance Count			
Register Spa	ace:	MMIO: 0/2/0			
Source:		VideoCS			
Default Valu	le:	0x0000000			
Access:		RO			
Size (in bits)	):	32			
Trusted Typ	e:	1			
Address:		12460h			
0		the number of clock cycles spent decoding/encoding the current frame. part of hardware context save and restore.			
DWord	DWord Bit Description				
0	31:0	<b>MFX Frame Performance Count</b> Total number of clocks between frame start and frame end. This count is incremented on crm_clk			



#### MFX\_ROW-PER-BS\_COUNT - MFX Frame Row-Stored/BitStream Read Count

Register Space:		MMIO: 0/2/0				
Source:		VideoCS				
Default Va	alue:	0x0000000				
Access:		RO				
Size (in bi	ts):	32				
Trusted T	/pe:	1				
Address:		12480h				
frame.		s the total number of row-stored/bit-stream read requests made by the pre-fetch engine per of part of hardware context save and restore.				
DWord	Bit	Description				
0	31:16	Reserved				
		Format: MBZ				
	15:0	<b>MFX row-stored/bit-stream read request Count</b> Total number of row-stored/bit-stream read requests sent by the memory pre-fetch engine per frame.				



# MFX\_LAT\_CT2 - MFX Memory Latency Count2

Register Space:		MMIO: 0/2/0	MMIO: 0/2/0				
Source:		VideoCS					
Default <b>`</b>	Value:	0x0000000					
Access:		RO					
Size (in l	oits):	32					
Trusted	Type:	1					
Address		12474h					
This regi	ister sto	pres the accumulative memory later	cy count on reference picture read requests.				
This reg	ister is	not part of hardware context save a	nd restore.				
DWord	Bit	Description					
0	31:26	Reserved					
		Format: MBZ					
25:0 MFX Reference picture read request - Accumulative Memory Latency Count for			est - Accumulative Memory Latency Count for the entire				
frame in 8xMedia clock cycles							
	count of all reference reads requested by motion						
		compensative engine per frame.					
		This number is used with MFX Fra	me Motion Comp Read Count to derive average memory				
		latency.					



# MFX\_LAT\_CT3 - MFX Memory Latency Count3

Register	Space	MMIO: 0/2/0		
Source:		VideoCS		
Default \	Value:	0x0000000		
Access:		RO		
Size (in l	bits):	32		
Trusted	Type:	1		
Address	:	12478h		
Max and	d currer	pres the max and min memory latency counts reported on row-stored/bit-stream read requests. In requests into memory sub-system engine. Not part of hardware context save and restore.		
DWord	Bit	Description		
0	31:24	<b>Max Request Count</b> This field indicates the maximum number of requests allowed by the memory sub-system channel.		
23:1		<b>Current Request Count</b> This field indicates the number of requests currently outstanding in the memory sub-system. This field should report with a value of zero at the end of frame; otherwise the pre-fetch engine most likely hung waiting for read data to be returned from sub-system.		
	15:8	<b>MFX row-stored/bit-stream read request - Max Latency Count in 8xMedia clock cycles</b> This field reports the maximum memory latency count on all row-stored/bit-stream reads requested by the memory pre-fetch engine.		
	<b>MFX row-stored/bit-stream read request - Min Latency Count in 8xMedia clock cycles</b> This field reports the minimum memory latency count on all row-stored/bit-stream reads requested by the memory pre-fetch engine.			



# MFX\_LAT\_CT4 - MFX Memory Latency Count4

Register Space:		MMIO: 0/2/0	MMIO: 0/2/0		
Source:		VideoCS			
Default '	Value:	0x0000000			
Access:		RO			
Size (in l	bits):	32			
Trusted	Type:	1			
Address	:	1247Ch			
		ores the accumulative memory latency not part of hardware context save an	count on row-stored/bit-stream read requests. I restore.		
DWord	Bit		Description		
0	31:26	Reserved			
		Format: MBZ			
	25:0	MFX row-stored/bit-stream read request - Accumulative Memory Latency Count for the entire frame in 8xMedia clock cycles The accumulative memory latency count of all row-stored/bit-stream reads requested by pre- fetch engine per frame. This number is used with Frame row-stored/bit-stream memory read count to derive average memory latency.			



# MFX\_STATUS\_FLAGS - MFX Pipeline Status Flags

			•	- <b>-</b>			
Register Spac	e: M	MIO: 0/2/0					
		ideoCS					
		x00000000					
Access:	R	0					
Size (in bits):	32	2					
Trusted Type:	: 1						
Address:	12	2438h					
This register s	stores the	various pipelir	ne status flags.				
			context save and restore.				
DWord	Bit		C.	Description			
0	31:17	Reserved					
		Format:			MBZ		
	16						
	15:10	Reserved					
		Format:			MBZ		
	9	Streamout Enable					
	8	Reserved					
	7	Post Deblocking Mode Enable					
	6	Pre Deblocking Mode Enable					
	5	Decoder M	—				
		Value		Nam	ne		
		0	Configure the MFD Engin	e for VLD Mo	ode		
		1     Configure the MFD Engine for IT Mode					
	4	Codec Selec					
			Value		Name		
		0		Decode			
		1		Encode			
	3:2			Lincode			
	5.2	Video Mode Value Name					
		00b	T UI U	MPEG2	Hume		
		00b		VC1			
		10b		AVC			
		11b		JPEG			



MFX_STATUS_FLAGS - MFX Pipeline Status Flags						
	1	Decoder Sh	ort Format	Mode		
		Value	Name		Description	
	0			AVC/VC1 Short Format Mode is in use		
		1		AVC/	VC1 Long Format Mode is in use	
	0 Stitch Mode					
		Value	Na	me	Description	
		0b			Not in Stitch Mode	
		1b			In the Special Stitch Mode	



# **MFX\_SLICE\_PERFORM\_CT - MFX Slice Performance Count**

Register	Space	: MMIO: 0/2/0
Source:		VideoCS
Default V	/alue:	0x0000000
Access:		RO
Size (in b	oits):	32
Trusted 1	Гуре:	1
Address:		12464h
This regi	ster st	ores the number of clock cycles spent decoding/encoding the current slice.
This reg	ister i	not part of hardware context save and restore.
DWord	Word Bit Description	
0	31:0	MFX Frame Performance Count
		Total number of clocks between slice start and slice end. This count is incremented on crm_clk



# **MISCCPCTL - Misc. Clocking / Reset Control Registers**

Register Space:		e: MMIO: 0/2/0			
Default Value:		0x0000002			
Size (in b	oits):	32			
Address:		09424h	09424h		
Miscellaneous Clocking / Reset Control Registers.					
DWord Bit		Description			
0	31:8	Bonus ECO bits			
		Access: R/W			
		Bonus ECO bits.			
	7:2	/:2 Reserved			
		Access:	RO		
		Reserved.			
	1	L1 Clock Ungate Enabling Control During Reset			
		Default Value:		1b	
		Access:		R/W	
		Control to enable/disable L1 clock gating during soft resets and FLR reset processing: 1 - Disable L1 clock gating during soft resets and FLR. 0 - Enable L1 clock gating during soft resets and FLR (default op).			
	0 DOP Clock Gating Enable for Render Clocks				
		Access:	R/W		
		Controls the Enabling of the DOP-level Render (crclk/cr2xclk) Clock Gating via PM event messages: 1 - Clock gating is enabled. 0 - Clock gating is disabled.			



# **RSTCTL - Misc. Reset Control Register**

Register	Space	: MMIO: 0/2/0					
Default V	'alue:	0x0000000					
Size (in bits): 32							
Address: 09420h							
Miscellar	ieous	reset control registers.					
DWord	Bit	Descrip	tion				
0	31:4	Reserved					
		Access:	RO				
		Reserved					
	3:2	Reset Staggering Period Control					
		Access:	R/W				
		Reset assertion staggering period between res	et domains during FLR and soft-resets:				
		00: 24 cs clocks					
		01: 48 cs clocks					
		10: 72 cs clocks					
		11: 96 cs clocks					
	1:0	Reset Residency Control					
		Access:	R/W				
		Reset assertion residency period for FLR and so	oft-resets.				
		00: 24 cs clocks					
		01: 48 cs clocks					
		10: 96 cs clocks					
		11: 192 cs clocks					



		GAB_MODE - Mode Register for GAB					
Register	Register Space: MMIO: 0/2/0						
Source:		BlitterCS					
Default \	/alue:	0x0000000					
Access:		R/W					
Size (in k	oits):	32					
Address:		220A0h-220A3h					
		E register nation that controls configurations in the GAB.					
DWord	Bit	Description					
0	31:16	Reserved					
		Access: WO					
		This register has bit-wise masking applied for writes. The register consists of 16 bits of data in [15:0], and 16 bits of corresponding masks in [31:16].					
		To set bit0, for example, the data would be 0x0001_0001.					
		To clear bit0, for example, the data would be 0x0001_0000.					
		Note that mask bit is the data bit offset + 16.					
	15:6	Reserved					
		Read/Write					
	5:3 BLB Arbitration Priority						
		Format: U3					
	2:0	BCS Arbitration Priority					
		Format: U3					



# **GAC\_MODE** - Mode Register for GAC

Register Space: MMIO: 0/2/0		MMIO: 0/2/0				
Source:		VideoCS				
Default Va	alue:	0x0000000				
Access:		R/W				
Size (in bi	ts):	32				
Address:		120A0h-120A3h				
The GAC_	MODE r	egister contains information that contr	ols configurations	in the GAC.		
DWord	Bit		Description			
0	31:16	Masks				
		Format:	Mask[15:0]			
		A 1 in a bit in this field allows the modification of the corresponding bit in Bits 15:0.				
	15:0	0 Reserved				
		Access:		R/W		



	<b>GAFS_MODE - Mode Register for GAFS</b>						
Register	Space	MMIO: 0/2/0					
Source:		RenderCS					
Default V	Value:	0x0000000					
Access:		R/W					
Size (in b	oits):	32					
Trusted	Туре:	1					
Address	:	0212Ch					
DWord	Bit		Description				
0	31:16	Mask Bits					
		Format:	Mask[15:0]				
		Masks: These bits serve as write enables for bits 15:0. If this register is written with any of these bits clear the corresponding bit in the field 15:0 will not be modified. Reading these bits always returns 0s.					
	15:11	Reserved					
		Format:	MBZ				
	10	Reserved					
		Format:	MBZ				
	9	Reserved					
	8:2	Reserved					
		Format: MBZ					
	1 Reserved						
	0	Selection of Arbitration for GAFS	election of Arbitration for GAFS				
		Format:		MBZ			
		GAFS data return policy from FFROB is a round-robin. This bit freezes the round robin to FF pipeline order.					



# **MTTLB\_VA - MT Virtual Page Address Registers**

Register Space:	MMIO: 0/2	MMIO: 0/2/0				
Source:	RenderCS					
Default Value:	0x0000000	0				
Access:	RO					
Size (in bits):	32					
Trusted Type:	1					
Address:	04800h-04803h					
DWord	Bit		Descri	iption		
0	31:12	Address				
		Format:	GraphicsAddress[3	31:12]		
		Page virtual ac	ldress.			
	11:0	Reserved				
		Format:			MBZ	



	NOPI	D - NOP Id	entificatio	n <mark>Reg</mark> iste	r	
Register Space:	MMIO: 0/	2/0				
Source:	RenderCS					
Default Value:	0x000000	00				
Size (in bits):	32					
Trusted Type:	1					
Address:	02094h	02094h				
		Descri	ption			
Access: RW						
The NOPID regist that enabled this		e Noop Identificatior updated.	value specified by t	he last MI_NOOP	' instruction	
DWord		Bit		Description		
0		31:22	Reserved			
			Format:		MBZ	
		21:0	Reserved			



# PAK\_ERR - PAK\_Stream-Out Report (Errors)

Register Space:		MMIO: 0/2/0				
Source:		VideoCS				
Default Valu	e:	0x0000000				
Access:		RO				
Size (in bits):	:	32				
Trusted Type	e:	1				
Address:		124E8h				
DWord	Bit	Description				
0	31:22	Reserved				
		Format:	MBZ			
	21	Incorrect IntraMBFlag in I-slice(AVCf)				
	20	Out of Range Symbol Code(AVC/mpeg2)				
	19	Incorrect MBType(AVC/mpeg2				
	18	Motion Vectors are not inside the frame boundary(mpeg2)				
17		Scale code is zero(mpeg2)				
	16	Incorrect DCTtype for given motionType(mpeg2)				
	15:8	MB Y-position				
	7:0	MB X-position				



# PAK\_WARN - PAK\_Stream-Out Report (Warnings)

Register Space:		MMIO: 0/2/0				
Source:		VideoCS				
Default Value	e:	0x0000000				
Access:		RO				
Size (in bits):		32				
Trusted Type	:	1				
Address:		124E4h				
DWord	Bit		Description			
0	31:22	Reserved				
		Format:	MB	3Z		
	21	Skip Run > 8192 (AVC)				
	20	Incorrect SkipMB (AVC and mpeg2)				
	19	Incorrect MV difference for dual-prime MB (mpeg2)				
	18	End of Slice signal missing on last MB of a Row(mpeg2)				
	17	Incorrect DCT type for field picture				
	16	MVs are not within defined range by fcode				
	15:8	MB Y-position				
	7:0	MB X-position				



# **PAK\_REPORT\_STAT - PAK Report Running Status**

Register Space:		MMIO: 0/2	2/0				
Source:		VideoCS					
Default Valu	e:	0x000000	00				
Access:		RO					
Size (in bits):		32					
Trusted Type	e:	1	1				
Address:		124ECh	124ECh				
DWord	Bit			Description			
0	31:1	Reserved					
	0	PAK Status					
			Name	Description			
				PAK engine is IDLE			
		1		PAK engine is currently generating bit stream.			



#### **PAT\_INDEX - PAT Index**

Register Space:	MMI	D: 0/2/0			
Source:	BSpe	с			
Default Value:	0x000	00003			
Access:	R/W				
Size (in bits):	32				
DWord	Bit			Description	
0	31:10	Reserved			
		Default Value:		000000000000000000000000000b	
	7:6	Reserved			
		Default Value:			00b
	5:4	LRU AGE			
		Default Value:			00b
		00: Take the age v		m Uncore CRs	
		01: Assign the age			
		10: Do not change		e on a hit	
		11: Assign the age	e of "3"		
	1:0	Mem Type			
		Default Value:			11b
		00: Uncacheable(U	JC)		
		01: Write Combini	ng(WC)		
		10: Write through(			
		11: Write back(WB	3)		



# **PDTLB\_VA - PDTLB\_VA Virtual page Address Registers**

Register Space:	MMIO: 0/2	MMIO: 0/2/0				
Source:	BlitterCS					
Default Value:	0x0000000	0				
Access:	RO					
Size (in bits):	32					
Trusted Type:	1					
Address:	24B00h-24	B03h				
This register is direc	tly mapped to	o the current Virtu	ual Addresses in the PD TLB.			
DWord	Bit		Description			
0	31:12	ADDRESS				
		Format:	GraphicsAddress[31:12]			
		PAGE VIRTUAL ADDRESS				
	11:0	RESERVED				
		Format:		MBZ		



# **UHPTR - Pending Head Pointer Register**

Register	Space	te: MMIO: 0/2/0						
Source:			BSpec					
Default \	Value:		0x00000	000				
Access: R/W								
Size (in b	oits):	rs): 32						
Address:	:	02134h						
Name:			RCS Pen	ding Head Pointer Register				
ShortNa	me:		RCS_UHI	PTR				
Address:	:		12134h					
Name:		,	VCS Pen	ding Head Pointer Register				
ShortNa	me:	,	VCS_UH	PTR				
Address:	:		22134h					
Name:			BCS Pen	ding Head Pointer Register				
ShortNa	me:		BCS_UHI	PTR				
				Programming Notes				
Render	CS On	ly: Once	SW uses	UHPTR to preempt the existing workload, should explicitly program				
MI_SET_	_CON	TEXT to s	save the	preempted context status before submitting the new workload.				
DWord	Bit			Description				
0	31:3		ointer A					
		Format	:	GraphicsAddress[31:3]				
				Description				
		This re	aister rei	presents the GFX address offset where execution should continue in the				
			•	wing execution of an MI_ARB_CHECK command.				
	2:1	Reserve	ed					
		Format	:	MBZ				
	0	Head P	ointer V	alid				
				Description				
		This bit	t is set b	y the software to request a pre-emption.				
	It is reset by hardware when an MI_ARB_CHECK command is parsed by the command							
		streamer. The hardware uses the head pointer programmed in this register at the time						
		the reset is generated.						
		Value	Name	Description				
		0		No valid updated head pointer register, resume execution at the current location in the ring buffer				
		1		Indicates that there is an updated head pointer programmed in this register				
				· · · · · · · · · · · · · · · · · · ·				





#### **PP\_DCLV - PPGTT Directory Cacheline Valid Register**

Register Space:	MMIO: 0/2/0
Source:	RenderCS
Default Value:	0x0000000, 0x0000000
Size (in bits):	64
Address:	02220h

Description

#### Access: R/W

This register controls update of the on-chip PPGTT Directory Cache during a context restore. Bits that are set will trigger the load of the corresponding 16 directory entry group. This register is restored with context (prior to restoring the on-chip directory cache itself). This register is also restored when switching to a context whose LRCA matches the current CCID if the **Force PD Restore** bit is set in the context descriptor.

The context image of this register must be updated and maintained by SW; SW should not normally need to read this register.

This register can also effectively be used to limit the size of a process's virtual address space. Any access by a process that requires a PD entry in a set that is not enabled in this register will cause a fatal error, and no fetch of the PD entry will be attempted.

#### **Programming Notes**

Page Directory Base Register is a Global Context Register (power context) and not maintained per context in ring buffer mode of submission. One should explicitly load PP\_DCLV followed by PP\_DIR\_BASE register through Load Register Immediate commands in Ring Buffer before submitting a context. One should program these registers after ensuring the pipe is completely flushed with TLB's invalidated.

DWord	Bit		Description
0	63:32	Reserved	
		Format:	MBZ
	31:0	PPGTT Directory Cache Restor	e [132] 16 entries
		Format:	BitMask[Enable]
			of the directory cache are considered valid and will be brought ese entries are considered invalid and fetch of these entries will



#### **PP\_PFD[0:31] - PPGTT Page Fault Data Registers**

Register Space:	MMIO: 0/2/0				
Source:	RenderCS				
Default Value:	0x00000000				
Access:	RO				
Size (in bits):	32				
Address:	04580h				
		 	• .		

The GTT Page Fault Log entries can be read from these registers. 4580h-4583h: Fault Entry 0

•••

45FCh-45FFh: Fault Entry 31

DWord	Bit		Description	
0	31:12	Fault Entry Page Addres	S	
		Format:	GraphicsAddress[31:12]	
			the bit in the GTT Page Fault Ind	ult Log entry. This field will contain a ication Register corresponding with the
	11:0	Reserved		
		Format:		MBZ



#### **MI\_PREDICATE\_RESULT - Predicate Rendering Data Result** Register Space: MMIO: 0/2/0 Source: RenderCS Default Value: 0x00000000 Access: R/W Size (in bits): 32 Address: 02418h DWord Bit Description 0 31:1 Reserved Format: MBZ MI\_PREDICATE\_RESULT 0 This bit is the result of the last MI\_PREDICATE.



# **MI\_PREDICATE\_DATA - Predicate Rendering Data Storage**

Register	Space	e: MMIO: 0/2/0
Source:		RenderCS
Default \	/alue:	0x0000000, 0x0000000
Access:		R/W
Size (in b	oits):	64
Address:		02410h-02417h
DWord	Bit	Description
0	63:0	MI_PREDICATE_DATA
		This register is used either as computed value based off the MI_PREDICATE_SRC0 and
		MI_PREDICATE_SRC1 or a temporary register. See Predicate Rendering section for more details.



MI_PREDICATE_SRC0 - Predicate Rendering Temporary
Register0

Register	Space	e: MMIO: 0/2/0
Source:		RenderCS
Default \	Value	0x0000000, 0x0000000
Access:		R/W
Size (in b	oits):	64
Address	:	02400h-02407h
DWord	Bit	Description
0	63:0	MI_PREDICATE_SRC0
		This register is a temporary register for Predicate Rendering. See Predicate Rendering section for more details.



M	[_P	REDICATE_SRC1 - Predicate Rendering Temporary Register1
Register	Space	e: MMIO: 0/2/0
Source:		RenderCS
Default <b>'</b>	Value:	0x0000000, 0x0000000
Access:		R/W
Size (in l	oits):	64
Address		02408h-0240Fh
DWord	Bit	Description
0	63:0	MI_PREDICATE_SRC1
		This register is a temporary register for Predicate Rendering. See Predicate Rendering section for

more details.



# **IA\_PRIMITIVES\_COUNT** - **Primitives Generated By VF**

Register	Spac	e: MMIO: 0/2/0
Source:		RenderCS
Default \	/alue:	0x0000000, 0x0000000
Access:		R/W
Size (in b	oits):	64
Trusted	Type:	1
Address:		02318h
This reg	ister	stores the count of primitives generated by VF. This register is part of the context save and restore.
DWord	Bit	Description
0	63:0	IA Primitives Count Report
		Total number of primitives output by the Vertex Fetch (IA) stage. This count is updated for every
		primitive output by the VF stage, as long as Statistics Enable is set in VF_STATE (see the Vertex
		Fetch Chapter in the 3D Volume.)



#### MIDARB\_PRIO\_HIT\_REGISTER - Priority Field in Programmable Arbitration for Hit

Register Spa	ace:	MMIO: 0/2	/0			
Source:		RenderCS				
Default Valu	ie:	0x0000000	0			
Access:		R/W				
Size (in bits)	:	32				
Trusted Typ	e:	1				
Address:		043A0h				
DWord	Bit				Descri	iption
0	31:12	Reserved				
	11:9	Encoded P	rogramma	ble Priority	for MIDAR	RB_GOTOFIELD_HIT3 Register
		Encoding	Priority 1	Priority 2	Priority 3	
		000	CS/VF/ISC	MT/CTC	RCC	
		001	CS/VF/ISC	RCC	MT/CTC	
		010	RCC	CS/VF/ISC	MT/CTC	
		011	RCC	MT/CTC	CS/VF/ISC	
		100	MT/CTC	CS/VF/ISC	RCC	
		101	MT/CTC	RCC	CS/VF/ISC	
		110	Reserved	Reserved	Reserved	
		111	Reserved	Reserved	Reserved	
	8:6	Encoded P	rogramma	ble Priority	for MIDAR	RB_GOTOFIELD_HIT2 Register
	5:3	Encoded P	rogramma	ble Priority	for MIDAR	RB_GOTOFIELD_HIT1 Register
	2:0	Encoded P	rogramma	ble Priority	for MIDAR	RB_GOTOFIELD_HIT0 Register



#### MIDARB\_PRIO\_NP\_REGISTER - Priority Field in Programmable Arbitration for Hit-NP

Register S	oace:	MMIO: 0/2	/0					
Source:		RenderCS						
Default Value: Access:		0x0000000						
		R/W	R/W					
Size (in bit	s):	32						
Trusted Type: Address:		1						
		043A4h						
Address:		04208h						
DWord	Bit			Des	scription			
0	31:20	Reserved						
	19:15		grammable Pri					
		Encoding	Priority 1	Priority 2	Priority 3	Priority 4		
		00000	CS/VF/ISC	MT_CTC	RCC	RCZ_HiZ_Stnc		
		00001	CS/VF/ISC	RCC	MT_CTC	RCZ_HiZ_Stnc		
		00010	RCC	CS/VF/ISC	MT_CTC	RCZ_HiZ_Stnc		
		00011	RCC	MT_CTC	CS/VF/ISC	RCZ_HiZ_Stnc		
		00100	MT_CTC	CS/VF/ISC	RCC	RCZ_HiZ_Stnc		
		00101	MT_CTC	RCC	CS/VF/ISC	RCZ_HiZ_Stnc		
		01000	CS/VF/ISC	MT_CTC	RCZ_HiZ_Stnc			
		01001	CS/VF/ISC	RCC	RCZ_HiZ_Stnc			
		01010	RCC	CS/VF/ISC	RCZ_HiZ_Stnc			
		01011	RCC	MT_CTC	RCZ_HiZ_Stnc			
		01100	MT_CTC	CS/VF/ISC	RCZ_HiZ_Stnc			
		01101	MT_CTC	RCC	RCZ_HiZ_Stnc			
		10000	CS/VF/ISC	RCZ_HiZ_Stnc		RCC		
		10001	CS/VF/ISC	RCZ_HiZ_Stnc		MT_CTC		
		10010	RCC	RCZ_HiZ_Stnc		MT_CTC		
		10011	RCC	RCZ_HiZ_Stnc		CS/VF/ISC		
		10100	MT_CTC	RCZ_HiZ_Stnc		RCC		
		10101	MT_CTC	RCZ_HiZ_Stnc		CS/VF/ISC		
		11000	RCZ_HiZ_Stnc		MT_CTC	RCC		
		11001	RCZ_HiZ_Stnc		RCC	MT_CTC		
		11010	RCZ_HiZ_Stnc	RCC	CS/VF/ISC	MT_CTC		



#### MIDARB\_PRIO\_NP\_REGISTER - Priority Field in Programmable Arbitration for Hit-NP

	11011	RCZ_HiZ_Stnc	RCC	MT_CTC	CS/VF/ISC
	11100	RCZ_HiZ_Stnc	MT_CTC	CS/VF/ISC	RCC
	11101	RCZ_HiZ_Stnc	MT_CTC	RCC	CS/VF/ISC
	Other values	Reserved			
14:10	Encoded Prog	grammable Pri	ority for MID	ARB_GOTOFIEL	D_NP2 Registe
9:5	Encoded Prog	grammable Pri	ority for MID	ARB_GOTOFIEL	D_NP1 Registe
4:0	Encoded Prog	grammable Pri	ority for MID	ARB_GOTOFIEL	D_NP0 Registe



#### MIDARB\_PRIO\_MISS\_REGISTER - Priority Field in Programmable Arbitration for Miss

Register Space:		MMIO: 0/2/0			
Source:		RenderCS			
Default Value:		x0000000			
Access:		2/W			
Size (in bits):		2			
Trusted Type:					
Address:		4204h			
DWord	Bit	Description			
0	31:20	Reserved			
	19:15	Encoded Programmable Priority for MIDARB_GOTOFIELD_MISS3 Register			
14:10		Encoded Programmable Priority for MIDARB_GOTOFIELD_MISS2 Register			
	9:5	Encoded Programmable Priority for MIDARB_GOTOFIELD_MISS1 Register			
	4:0	Encoded Programmable Priority for MIDARB_GOTOFIELD_MISS0 Register			



#### **PRIV\_PAT - Private PAT**

Register Space:	MM	O: 0/2/0		
Source:	BSpec			
Default Value:	0x00	0x0000000		
Size (in bits):	32	32		
Address:	0408	8h		
DWord	Bit		Description	
0	31:0	Private PAT		
		Default Value:	0000000h	
		Access:	R/W	
		Bit[31:16]: Reserved. Bit[15:8]: PPGTT Private PA Bit[7:6]: Reserved. Bit[5:4]: (See below.) 00b: Age is 0. 01b: Age is 1. 10b: Age is 2. 11b: Age is 3. Bit[1:0]: (see below): 00b: Uncached with fence. 01b: Write Combining (tra- 10b: Write Through. 11b: Write Back.		



		PS_DEPTH_COUNT - PS Depth Count
Register Space:		e: MMIO: 0/2/0
Source:		RenderCS
Default \	/alue:	0x0000000, 0x0000000
Access:		R/W
Size (in b	oits):	64
Trusted <sup>-</sup>	Гуре:	1
Address:		02350h
context s	save a	stores the value of the count of samples that have passed the depth test. This register is part of the ind restore. Note that the value of this register can be obtained in a pipeline-synchronous fashion eline flush by using the 3DCONTROL command. See 3D Overview in the 3D volume.
DWord	Bit	Description
01	63:0	Depth Count
		This register reflects the total number of samples that have passed the depth test (i.e., will be
		visible). All samples are counted when Statistics Enable is set in the Windower State. See the
Windower chapter of the 3D volume for details. Samples that pass the depth test but fa stencil test will not be counted.		



# **PS\_INVOCATION\_COUNT - PS Invocation Count**

Register Space: MMIO: 0/2/0		e: MMIO: 0/2/0			
Source: RenderCS					
Default Value: 0x0000000, 0x0000000		0x0000000, 0x0000000			
Access:		R/W			
Size (in l	oits):	64			
Trusted Type: 1		1			
Address:		02348h			
DWord	Bit	Description			
01	63:0	<b>Description</b> <b>nvocation Count</b> ects a count of the total number of pixels (including unlit "helper pixels" within a subspan that I to go through the PS shader to provide 2x2 gradients) that are dispatched to pixel shader cations while Statistics Enable is set in the Windower. See the Windower chapter of the 3D me for details. This count will generally be much greater than the actual count of PS threads a single thread may process up to 32 pixels.			



# **RCGCTL1 - RAM Clock Gating Control 1**

Register Space: Default Value:				
Size (in bits): 32				
Address:		09410h		
RAM Clo	ck G	ating Control Registers.		
Word	Bit	De	escription	
0	31	USBunit RAM Clock Gating Disable		
		Access:	R/W	
		USBunit RAM Clock Gating Disable Contro '0': Clock Gating Enabled. (i.e., clocks can b for functionality) '1': Clock Gating Disabled. (i.e., clocks are t	be gated when they are not required to toggle	
	30	VLFunit RAM Clock Gating Disable		
	50	Access:	R/W	
-	20	for functionality) '1': Clock Gating Disabled. (i.e., clocks are t	be gated when they are not required to toggle toggling, always)	
	29	VISunit RAM Clock Gating Disable	R/W	
		VISunit RAM Clock Gating Disable Control		
		-	be gated when they are not required to toggle	
	28	STCunit RAM Clock Gating Disable		
		Access:	R/W	
		STCunit RAM Clock Gating Disable Contro '0': Clock Gating Enabled. (i.e., clocks can b for functionality) '1': Clock Gating Disabled. (i.e., clocks are t	be gated when they are not required to toggle	
-	27	TDSunit RAM Clock Gating Disable		
		Access:	R/W	
		TDSunit RAM Clock Gating Disable Contro '0': Clock Gating Enabled. (i.e., clocks can k for functionality) '1': Clock Gating Disabled. (i.e., clocks are t	be gated when they are not required to toggle	



# **RCGCTL1 - RAM Clock Gating Control 1**

26	VMCunit RAM Clock Gating Disable			
	Access:	R/W		
	VMCunit RAM Clock Gating Disable Control:			
	'0': Clock Gating Enabled. (i.e., clocks can be ga for functionality) '1': Clock Gating Disabled. (i.e., clocks are toggl			
25	QRCunit RAM Clock Gating Disable			
	Access:	R/W		
	QRCunit RAM Clock Gating Disable Control: '0': Clock Gating Enabled. (i.e., clocks can be ga for functionality) '1': Clock Gating Disabled. (i.e., clocks are toggl			
24	SCunit RAM Clock Gating Disable			
	Access:	R/W		
	SCunit RAM Clock Gating Disable Control: '0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to toggle for functionality) '1': Clock Gating Disabled. (i.e., clocks are toggling, always)			
23	SVLunit RAM Clock Gating Disable			
	Access:	R/W		
	SVLunit RAM Clock Gating Disable Control: '0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to toggle for functionality) '1': Clock Gating Disabled. (i.e., clocks are toggling, always)			
	5.	ing, always)		
22	5.	ing, always)		
22	'1': Clock Gating Disabled. (i.e., clocks are toggl	ing, always) R/W		
22	'1': Clock Gating Disabled. (i.e., clocks are toggl VFunit RAM Clock Gating Disable			
22	'1': Clock Gating Disabled. (i.e., clocks are toggl VFunit RAM Clock Gating Disable Access: VFunit RAM Clock Gating Disable Control: '0': Clock Gating Enabled. (i.e., clocks can be ga	R/W		
22	'1': Clock Gating Disabled. (i.e., clocks are toggl <b>VFunit RAM Clock Gating Disable</b> Access: VFunit RAM Clock Gating Disable Control: '0': Clock Gating Enabled. (i.e., clocks can be ga for functionality)	R/W ted when they are not required to toggle		
22	'1': Clock Gating Disabled. (i.e., clocks are toggl VFunit RAM Clock Gating Disable Access: VFunit RAM Clock Gating Disable Control: '0': Clock Gating Enabled. (i.e., clocks can be ga	R/W ted when they are not required to toggle		
	'1': Clock Gating Disabled. (i.e., clocks are toggl <b>VFunit RAM Clock Gating Disable</b> Access: VFunit RAM Clock Gating Disable Control: '0': Clock Gating Enabled. (i.e., clocks can be ga for functionality)	R/W ted when they are not required to toggle		
22	<ul> <li>'1': Clock Gating Disabled. (i.e., clocks are toggl</li> <li>VFunit RAM Clock Gating Disable</li> <li>Access:</li> <li>VFunit RAM Clock Gating Disable Control:</li> <li>'0': Clock Gating Enabled. (i.e., clocks can be ga for functionality)</li> <li>'1': Clock Gating Disabled. (i.e., clocks are toggl)</li> </ul>	R/W ted when they are not required to toggle		



	RCGCTL1 - RAM Clock Ga		
	for functionality)		
	'1': Clock Gating Disabled. (i.e., clocks are togglir	ng, always)	
20	Reserved		
	Access:	RO	
	Reserved.		
19	SVGunit RAM Clock Gating Disable		
	Access:	R/W	
	SVGunit RAM Clock Gating Disable Control: '0': Clock Gating Enabled. (i.e., clocks can be gate for functionality) '1': Clock Gating Disabled. (i.e., clocks are togglin		
18	RCZunit RAM Clock Gating Disable		
	Access:	R/W	
	RCZunit RAM Clock Gating Disable Control:		
	<ul> <li>'0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to toggl for functionality)</li> <li>'1': Clock Gating Disabled. (i.e., clocks are toggling, always)</li> </ul>		
17	RCPBEunit RAM Clock Gating Disable		
	Access:	R/W	
	RCPBEunit RAM Clock Gating Disable Control: '0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to toggl for functionality) '1': Clock Gating Disabled. (i.e., clocks are toggling, always)		
16	RCCunit RAM Clock Gating Disable		
	Access:	R/W	
	RCCunit RAM Clock Gating Disable Control: '0': Clock Gating Enabled. (i.e., clocks can be gate for functionality) '1': Clock Gating Disabled. (i.e., clocks are togglin		
15	PSDunit RAM Clock Gating Disable		
	Access:	R/W	
	PSDunit RAM Clock Gating Disable Control: '0': Clock Gating Enabled. (i.e., clocks can be gate	d when they are not required to too	
	for functionality)		



# **RCGCTL1 - RAM Clock Gating Control 1**

	'1': Clock Gating Disabled. (i.e., clocks are toge	gling, always)		
14	MTunit RAM Clock Gating Disable			
	Access:	R/W		
	MTunit RAM Clock Gating Disable Control:			
	'0': Clock Gating Enabled. (i.e., clocks can be g	ated when they are not required to toggle		
	for functionality)			
	'1': Clock Gating Disabled. (i.e., clocks are toge	gling, always)		
13	SBEunit RAM Clock gating Disable			
	Access:	R/W		
	SBEunit RAM Clock Gating Disable Control:			
	'0': Clock Gating Enabled. (i.e., clocks can be g	ated when they are not required to toggle		
	for functionality)			
	'1': Clock Gating Disabled. (i.e., clocks are toge	gling, always)		
12	IZunit RAM Clock Gating Disable			
	Access:	R/W		
	IZunit RAM Clock Gating Disable Control:			
	'0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to toggle			
l	for functionality)			
	'1': Clock Gating Disabled. (i.e., clocks are tog	gling, always)		
11		gling, always)		
11	'1': Clock Gating Disabled. (i.e., clocks are tog	gling, always) R/W		
11	'1': Clock Gating Disabled. (i.e., clocks are togo			
11	'1': Clock Gating Disabled. (i.e., clocks are togo <b>IECPunit RAM Clock Gating Disable</b> Access:	R/W		
11	'1': Clock Gating Disabled. (i.e., clocks are togg <b>IECPunit RAM Clock Gating Disable</b> Access: IECPunit RAM Clock Gating Disable Control: '0': Clock Gating Enabled. (i.e., clocks can be g for functionality)	R/W gated when they are not required to toggle		
11	'1': Clock Gating Disabled. (i.e., clocks are toge <b>IECPunit RAM Clock Gating Disable</b> Access: IECPunit RAM Clock Gating Disable Control: '0': Clock Gating Enabled. (i.e., clocks can be g	R/W gated when they are not required to toggle		
	'1': Clock Gating Disabled. (i.e., clocks are togg <b>IECPunit RAM Clock Gating Disable</b> Access: IECPunit RAM Clock Gating Disable Control: '0': Clock Gating Enabled. (i.e., clocks can be g for functionality)	R/W gated when they are not required to toggle		
	<ul> <li>'1': Clock Gating Disabled. (i.e., clocks are togg</li> <li>IECPunit RAM Clock Gating Disable</li> <li>Access:</li> <li>IECPunit RAM Clock Gating Disable Control:</li> <li>'0': Clock Gating Enabled. (i.e., clocks can be g for functionality)</li> <li>'1': Clock Gating Disabled. (i.e., clocks are togg</li> </ul>	R/W gated when they are not required to toggle		
	<ul> <li>'1': Clock Gating Disabled. (i.e., clocks are togg</li> <li>IECPunit RAM Clock Gating Disable</li> <li>Access:</li> <li>IECPunit RAM Clock Gating Disable Control:</li> <li>'0': Clock Gating Enabled. (i.e., clocks can be g for functionality)</li> <li>'1': Clock Gating Disabled. (i.e., clocks are togg</li> <li>IECPUNIT RAM Clock Gating Disable</li> </ul>	R/W gated when they are not required to toggle gling, always)		
	<ul> <li>'1': Clock Gating Disabled. (i.e., clocks are togg</li> <li>IECPunit RAM Clock Gating Disable</li> <li>Access:</li> <li>IECPunit RAM Clock Gating Disable Control:</li> <li>'0': Clock Gating Enabled. (i.e., clocks can be g for functionality)</li> <li>'1': Clock Gating Disabled. (i.e., clocks are togg</li> <li>ICunit RAM Clock Gating Disable</li> <li>Access:</li> <li>ICunit RAM Clock Gating Disable Control:</li> <li>'0': Clock Gating Enabled. (i.e., clocks can be g</li> </ul>	R/W gated when they are not required to toggle gling, always) R/W		
	<ul> <li>'1': Clock Gating Disabled. (i.e., clocks are togg</li> <li>IECPunit RAM Clock Gating Disable         <ul> <li>Access:</li> <li>IECPunit RAM Clock Gating Disable Control:</li> <li>'0': Clock Gating Enabled. (i.e., clocks can be g for functionality)</li> <li>'1': Clock Gating Disabled. (i.e., clocks are togg</li> </ul> </li> <li>ICunit RAM Clock Gating Disable         <ul> <li>Access:</li> <li>ICunit RAM Clock Gating Disable</li> <li>Access:</li> <li>ICunit RAM Clock Gating Disable Control:</li> <li>'0': Clock Gating Enabled. (i.e., clocks can be g for functionality)</li> </ul> </li> </ul>	R/W gated when they are not required to toggle gling, always) R/W gated when they are not required to toggle		
	<ul> <li>'1': Clock Gating Disabled. (i.e., clocks are togg</li> <li>IECPunit RAM Clock Gating Disable</li> <li>Access:</li> <li>IECPunit RAM Clock Gating Disable Control:</li> <li>'0': Clock Gating Enabled. (i.e., clocks can be g for functionality)</li> <li>'1': Clock Gating Disabled. (i.e., clocks are togg</li> <li>ICunit RAM Clock Gating Disable</li> <li>Access:</li> <li>ICunit RAM Clock Gating Disable Control:</li> <li>'0': Clock Gating Enabled. (i.e., clocks can be g</li> </ul>	R/W gated when they are not required to toggle gling, always) R/W gated when they are not required to toggle		
	<ul> <li>'1': Clock Gating Disabled. (i.e., clocks are togg</li> <li>IECPunit RAM Clock Gating Disable         <ul> <li>Access:</li> <li>IECPunit RAM Clock Gating Disable Control:</li> <li>'0': Clock Gating Enabled. (i.e., clocks can be g for functionality)</li> <li>'1': Clock Gating Disabled. (i.e., clocks are togg</li> </ul> </li> <li>ICunit RAM Clock Gating Disable         <ul> <li>Access:</li> <li>ICunit RAM Clock Gating Disable</li> <li>Access:</li> <li>ICunit RAM Clock Gating Disable Control:</li> <li>'0': Clock Gating Enabled. (i.e., clocks can be g for functionality)</li> </ul> </li> </ul>	R/W gated when they are not required to toggle gling, always) R/W gated when they are not required to toggle		
10	<ul> <li>'1': Clock Gating Disabled. (i.e., clocks are togg</li> <li>IECPunit RAM Clock Gating Disable</li> <li>Access:</li> <li>IECPunit RAM Clock Gating Disable Control:</li> <li>'0': Clock Gating Enabled. (i.e., clocks can be g for functionality)</li> <li>'1': Clock Gating Disabled. (i.e., clocks are togg</li> <li>ICunit RAM Clock Gating Disable</li> <li>Access:</li> <li>ICunit RAM Clock Gating Disable Control:</li> <li>'0': Clock Gating Enabled. (i.e., clocks can be g for functionality)</li> <li>'1': Clock Gating Enabled. (i.e., clocks are togg</li> <li>ICunit RAM Clock Gating Disable Control:</li> <li>'0': Clock Gating Enabled. (i.e., clocks can be g for functionality)</li> <li>'1': Clock Gating Disabled. (i.e., clocks are togg</li> </ul>	R/W gated when they are not required to toggle gling, always) R/W gated when they are not required to toggle		



	<b>RCGCTL1 - RAM Clock Ga</b>	ting Control 1		
	'0': Clock Gating Enabled. (i.e., clocks can be gated for functionality)			
	'1': Clock Gating Disabled. (i.e., clocks are toggling	g, always)		
8	GAMunit RAM Clock Gating Disable			
	Default Value:	0b		
	Access:	R/W		
	GAMunit RAM Clock Gating Disable Control:	L		
	'0': Clock Gating Enabled. (i.e., clocks can be gate	d when they are not required to toggle		
	for functionality)			
	'1': Clock Gating Disabled. (i.e., clocks are toggling	g, always)		
7	BCunit RAM Clock Gating Disable			
	Access:	R/W		
	BCunit RAM Clock Gating Disable Control:			
	'0': Clock Gating Enabled. (i.e., clocks can be gate	d when they are not required to toggle		
	for functionality)			
	'1': Clock Gating Disabled. (i.e., clocks are toggling	g, always)		
6	HDCunit RAM Clock Gating Disable			
	Access:	R/W		
	GAFSunit RAM Clock Gating Disable Control:			
	'0': Clock Gating Enabled. (i.e., clocks can be gate	d when they are not required to toggle		
	for functionality)			
	'1': Clock Gating Disabled. (i.e., clocks are toggling	g, aiways)		
5	DMunit RAM Clock Gating Disable			
	Access:	R/W		
	DMunit RAM Clock Gating Disable Control:			
	'0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to toggle			
	for functionality)			
	'1': Clock Gating Disabled. (i.e., clocks are toggling	g, always)		
4	Reserved			
	Access:	RO		
	Reserved.			
3	CSunit RAM Clock Gating Disable			
	Access:	R/W		
	CSunit RAM Clock Gating Disable Control:			



# **RCGCTL1 - RAM Clock Gating Control 1**

	'0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to toggle for functionality) '1': Clock Gating Disabled. (i.e., clocks are toggling, always)		
2	BLBunit RAM Clock Gating Disable		
	Access:	R/W	
	BLBunit RAM Clock Gating Disable Contro '0': Clock Gating Enabled. (i.e., clocks can for functionality) '1': Clock Gating Disabled. (i.e., clocks are	be gated when they are not r	equired to toggle
1	MPCunit RAM Clock Gating Disable		
	Access:	R/W	
MPCunit RAM Clock Gating Disable Control: '0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required for functionality) '1': Clock Gating Disabled. (i.e., clocks are toggling, always)			
0	BFunit RAM Clock Gating Disable		
	Access:	R/W	
	BFunit RAM Clock Gating Disable Control: '0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to toggle for functionality) '1': Clock Gating Disabled. (i.e., clocks are toggling, always)		



# **RCGCTL2 - RAM Clock Gating Control 2**

Register	Space	e: MMIO: 0/2/0			
Default \	/alue:	0x0000000			
Size (in bits): 32					
Address:		09414h			
RAM Clo	ck Ga	ting Control Registers.			
DWord	Bit	Description	Description		
0	31	SPARE 2 clock gate disable			
		Access:	R/W		
		SPARE 2 Unit Clock Gating Disable Control:			
		0: Clock Gating Enabled. (I.e., clocks can be gated whe for functionality.)	en they are not required to toggle		
		1: Clock Gating Disabled. (I.e., clocks are toggling, alw	avs)		
		1. Clock Guting Disubled. (i.e., clocks are togginig, and	ays.)		
	30:2	Reserved			
		Access:	RO		
		Reserved.			
	1	VSunit RAM Clock Gating Disable			
		Access:	R/W		
		VSunit RAM Clock Gating Disable Control:			
		0: Clock Gating Enabled. (I.e., clocks can be gated whe	en they are not required to toggle		
		for functionality.)			
		1: Clock Gating Disabled. (I.e., clocks are toggling, alw	ays.)		
	0	Reserved			
		Access:	RO		
		Reserved.			



#### RCC\_LRA\_0 - RCC LRA 0

Register Space:		D: 0/2/0	
Default Value:	0x3F100F00		
Size (in bits):	32		
Address:	04058	3h-0405Bh	
RCC LRA 0			
DWord	Bit		Description
0	31:30	Reserved	
		Default Value:	00b
		Access:	RO
		Reserved	
		Format: U1	
	29:24	RCC LRA1 Max	
		Default Value:	111111b
		Access:	R/W
		RCC LRA1 Max	
		Format: U6	
_		Maximum value of programma	able LRA1
	23:22	Reserved	
		Default Value:	00b
		Access:	RO
		Reserved	
-		Format: U1	
	21:16	RCC LRA1 Min	
		Default Value:	010000b
		Access:	R/W
		RCC LRA1 Min	
		Format: MBZ	
		Minimum value of programma	ble LRA1
	15:14	Reserved	
		Default Value:	00Ь
		Access:	RO
		Reserved	



#### RCC\_LRA\_0 - RCC LRA 0

	Format: U1			
13:8	RCC LRA0 Max			
15.0	Default Value: 001111b		h	
	Access:		R/W	
	RCC LRA0 Max			
	Format: U1			
	Maximum value of programmable LRA0			
7:6	Reserved			
	Default Value:		00b	
	Access:		RO	
	Reserved			
	Format: U1			
5:0	RCC LRA0 Min			
	Default Value:	000000	000000b	
	Access:	R/W	R/W	
	RCC LRA0 Min			
	Format: U6			
	Minimum value of programmable LRA0			



# RCC\_LRA\_1 - RCC LRA 1

Register Space:	MMIO: (	)/2/0				
Default Value:	0x00010000					
Size (in bits):	32					
	0405Ch-0405Fh					
Address:	0405Ch-	0405Fn				
RCC LRA 1						
DWord	Bit			Description		
0	31:20	Reserved		F		
		Default Value	:	00000000000b		
		Access:		RO		
		Reserved				
		Format:	MBZ			
	19:18	MSC LRA				
		Default Value:			00b	
		Access:			R/W	
		MSC LRA				
		Format:	U1			
		Which LRA sh	nould MSC use			
	17:16	RCC LRA				
		Default Value	:		01b	
		Access:			R/W	
		RCC LRA				
		Format: MBZ	2			
		Which LRA should RCC use				
	15:0	Reserved				
		Default Value	:	000000000000000b		
		Access:		RO		
		Reserved				
		Format:	MBZ			



# **RCCTLB\_VA - RCC Virtual page Address Registers**

Register Space:	MMIO: 0/2/0				
Source:	RenderCS				
Default Value:	0x0000000	)			
Access:	RO				
Size (in bits):	32				
Trusted Type:	1				
Address:	04A00h-04A	A03h			
These registers are	directly mappe	ed to the curren	it Virtual Addresses i	in the RCCTLB	(Render Cache for Color TLB).
DWord	Bit		0	Description	
0	31:12	Address			
		Format:	GraphicsAdd	dress[31:12]	
		Page virtual a	ddress.		
	11:0	11:0 Reserved			
		Format:			MBZ



# **RCS\_BB\_STATE - RCS Batch Buffer State Register**

Register Sp	ace:	MMIO: 0/2/0					
Source:		RenderCS					
Default Val	ue:	0x0000	00000				
Access:		RO					
Size (in bits	5):	32					
Address:		02110ł	١				
This regist	er conta	ains the at	tributes of the current	batch buffer initiated fro	om the Ring Buffer.		
Software s buffer.	hould a	lways set t	,	, ,	et written by a context restore. command when initiating a batch		
DWord	Bit	Description					
0	31:9	Reserved			<u>.</u>		
		Format:			MBZ		
	8	Reserved					
		Format:			MBZ		
	7	Reserved					
		Format:			MBZ		
	6	Reserved					
	5	Address Space Indicator					
		Value	Name		Description		
		0h	GGTT <b>[Default]</b>	This batch buffer is loca	ated in GGTT memory		
		1h	PPGTT	This batch buffer is loca	ated in PPGTT memory.		
	4	Reserved					
	3:0	Reserved					
		Format:			MBZ		



# **RCZTLB\_VA - RCZ Virtual Page Address Registers**

Register Space:	MMIO: 0/2/0				
Source:	RenderCS				
Default Value:	0x0000000				
Access:	RO				
Size (in bits):	32				
Trusted Type:	1				
Address:	04B00h-04B	03h			
These registers are Hi Z, and Stencil TLI		d to the curren	t Virtual Addresses in the RCZTI	B (Render Cache for Z (Depth),	
DWord	Bit		Description		
0	31:12	Address			
		Format:	GraphicsAddress[31:12]		
		Page virtual address.			
	11:0	Reserved			
		Format:		MBZ	



# TLBPEND\_RDY0 - Ready Bit Vector 0 for TLBPEND registers

Register Space:	MMIO: 0/2/0				
Source:	Rende	erCS			
Default Value:	0x000	00000			
Access:	R/W				
Size (in bits):	32				
Trusted Type:	1				
Address:	04708	h-0470Bh			
This register contains	This register contains the ready bits for entries 0-31 of TLBPEND structure (Cycles pending TLB translation).				
DWord		Bit	Description		
0 31:0			Ready bits per entry		



TLBPEND_RDY1 - Ready Bit Vector 1 for TLBPEND registers				
Register Space:	MMIO	: 0/2/0		
Source:	Rende	erCS		
Default Value:	0x0000000			
Access:	R/W			
Size (in bits):	32			
Trusted Type:	1			
Address:	0470C	h-0470Fh		
This register contains the ready bits for entries 32-63 of TLBPEND structure (Cycles pending TLB translation).				
DWord		Bit	Description	
0		31:0	Ready bits per entry	



# **RBSYNC - Render/Blitter Semaphore Sync Register**

Register S	pace:	MMIO: 0/2/0			
Source:		RenderCS			
Default Va	lue:	0x0000000			
Access:		R/W			
Size (in bit	s):	32			
Trusted Ty	pe:	1			
Address:		02044h			
This regist	This register is written by BCS, read by CS.				
DWord	Bit	Description			
0	31:0	Semaphore Data			
		Semaphore data for synchronization between render engine and blitter engine.			



# **MI\_MODE** - Render Mode Register for Software Interface

Register	Space:	e: MMIO: 0/2/0							
Source:		R	enderCS						
Default \	Value:	0	x00000000						
Access:		R	/W						
Size (in b	oits):	3	2						
Address:	:	0	209Ch						
The MI_I function		register	contains informatic	on that controls so	oftware interface aspects of the Memory Interface				
DWord	Bit			E.	Description				
0	31:16	Masks							
		Format	t:	N	ask[15:0]				
		A 1 in a	bit in this field allo	ws the modificat	on of the corresponding bit in Bits 15:0				
	15	Suspen	d Flush						
		Format	t:		U1				
		Value	Name		Description				
		0h	No Delay	HW will not dela	, flush, this bit will get cleared by				
			[Default]	MI_SUSPEND_FL	USH as well				
		1h	h Delay Flush Suspend flush is active						
		Programming Notes							
		This should only be written to from the ring using MI_SUSPEND_FLUSH. It is considered undefined if written by software through MMIO							
	14	Async Flip Performance mode							
		Format: U1							
		Value		me	Description				
		0h	Performance mode [Default]	e enabled	The stall of the flip event is in the windower				
		1h	Performance mode	e disabled	The stall of the flip event is in the command stream				
				Due que ma	sing Notes				
		Thiski	t must be set to 11		ning Notes				
					lip Performance mode.				
		comma this bro	ands following the leaks the usage mod	MI_WAIT_FOR_EV del of controlling	bled stall is in the Windower allowing the ENT to be parsed by command streamer, the display message generation in display ATE commands from ring buffer.				



# **MI\_MODE** - Render Mode Register for Software Interface

13	Flush Performance mode						
	Format: U1						
	Value	Name				Description	
	0h	run fast resto	estore [Default]			onPipelined SV flush.	
	1h	run slow lega	icy restore		With N	NonPipelined SV flush.	
12	MI_FLUSH	/I_FLUSH Enable					
	Format:				Enable		
	PIPE_CON	TROL is a supe	rset of MI_FLU	SH. Since MI	_FLUSH	l is redundant, by default, it is disabled	
11	Invalidate	UHPTR enab	e				
	Format:				Enable		
	If bit set H equal to U		alid bit of UHI	PTR (2134h, Ł	oit 0) wł	hen current active head pointer is	
10	10 Reserved						
	Format:					MBZ	
9	Rings Idle	1					
	Format:					U1	
	Read Only	Status bit		I			
	Value	Na	me			Description	
	0h	Not Idle [Def	ault]	Parser not Idle or Ring Arbiter not Idle.			
	1h	Idle		Parser Idle a	nd Ring	g Arbiter Idle.	
	Programming Notes						
	Writes to this bit are not allowed.						
8	Stop Ring	s					
	Format:					U1	
	Value	Name			Des	cription	
	0h	[Default]	Normal Oper				
	1h   Parser is turned off and Ring arbitration is turned off.						
	Programming Notes						
	Software must set this bit to force the Rings and Command Parser to Idle. Software must read a 1 in the Ring Idle bit after setting this bit to ensure that the hardware is idle.						
	Software	must clear this	bit for Rings t	o resume no	rmal op	peration.	
7	Reserved						
	Format:					MBZ	



# **MI\_MODE - Render Mode Register for Software Interface**

6	Vertex	Shader Time	er Dispatch Enable		
	Forma	t:		Enabl	e
	Value	Name		Des	cription
	0h	Disable [ <b>Default]</b>	•	e the timer for dispatch of single vertices from the vertex shader. shader will try to collect 2 vertices before a dispatch	
	1h	Enable	Enable the timer for dispa shader thread after the time		ngle vertices. Dispatch a single verte es.
5	Reserv	ed			
	Forma	t:			MBZ
4	Reserved				
	Forma	t:			MBZ
3:1	Reserved				
	Format:				MBZ
0	Mask IIR disable				
	Forma	t:		Disable	
	interrup	ot acknowledg	· ·	pending.	interrupts in the IIR register if an Setting this bit to a 1 allows interru acknowledge is pending.



# **RVSYNC - Render/Video Semaphore Sync Register**

Register S	pace:	MMIO: 0/2/0			
Source:		RenderCS			
Default Va	lue:	0x0000000			
Access:		R/W			
Size (in bit	s):	32			
Trusted Ty	pe:	1			
Address:		02040h			
This regist	This register is written by VCS, read by CS.				
DWord	Bit	Description			
0	31:0	Semaphore Data			
		Semaphore data for synchronization between render engine and blitter engine.			

Г



٦

		PR_CTR - Re	nder Watchdog Counter		
Register Space: MMIO: 0/2/0					
Source:		RenderCS			
Default V	alue:	0x0000000			
Access:		RO			
Size (in bits):		32			
Address:		02190h			
DWord	Bit		Description		
0	31:0	Counter Value			
		Format:	U32		
		This register reflects the rend	der watchdog counter value itself. It cannot be written to.		



# **PR\_CTR\_THRSH** - Render Watchdog Counter Threshold

Register Space:		e: MMIO: 0/2/0						
Source:		RenderCS						
Default \	Value:	0x00150000						
Access:		R/W						
Size (in b	oits):	32						
Address:		0217Ch						
DWord	Bit		Description					
0	31:0	ounter logic Threshold						
		Default Value:	00150000h					
		Format:	U32					
		This field specifies the threshold that the hardware checks against for the value of the render						
		clock counter before generating an interrupt. The counter in hardware generates an interrupt						
		when the threshold is reached, rolls over and starts counting again. The interrupt generated is the						
		3 , 1	this watchdog timer is intended primarily to remedy VLD					
		hangs on the main pipeline.						



# MFC\_BITSTREAM\_SE\_BITCOUNT\_FRAME - Reported Bitstream Output Bit Count for Syntax Elements Only Register

Register Space:		e: MMIO: 0/2/0						
Source:		VideoCS						
Default V	/alue:	0x0000000						
Access:		RO						
Size (in b	oits):	32						
Trusted 7	Гуре:	1						
Address: 124A4h								
This reg	ister s	tores the count of number of bits in the bitstream due to syntax elements only. This excludes						
header/	byte a	alignment /tail/EMU/CABAC-0word/padding bits but includes the stop-one-bit. This register is part						
of the co	ntext	save and restore.						
DWord	Bit	Description						
0	31:0	MFC Bitstream Syntax Element Only Bit Count						
		Total number of bits in the bitstream output due to syntax elements only. It includes the data						
	bytes only. This count is updated for every time the internal bitstream counter is incremented							
		its reset at image start.						



### MFC\_BITSTREAM\_BYTECOUNT\_FRAME - Reported Bitstream Output Byte Count per Frame Register

Register Space:		e: MMIO: 0/2/0					
Source:		VideoCS					
Default Value:		0x0000000					
Access:		RO					
Size (in b	oits):	32					
Trusted <sup>-</sup>	Type:	1					
Address:		124A0h					
This regi	ster s	tores the count of bytes of the bitstream output per frame					
DWord	Bit	Description					
0	31:0	MFC Bitstream Byte Count per Frame					
	Total number of bytes in the bitstream output per frame from the encoder. This includes						
		header/tail/byte alignment/data bytes/EMU (emulation) bytes/cabac-zero word insertion/padding					
insertion. This count is updated for every time the internal bitstream counter is incremented							
		its reset at image start.					



### MFC\_AVC\_CABAC\_BIN\_COUNT\_FRAME - Reported Bitstream Output CABAC Bin Count Register

Register Space:		e: MMIO: 0/2/0			
Source:		VideoCS			
Default \	/alue:	0x0000000			
Access:		RO			
Size (in b	oits):	32			
Trusted <sup>-</sup>	Type:	1			
Address:		124A8h			
This regi	ster s	tores the count of number of bins per frame.			
DWord	Bit	Description			
0	31:0	MFC AVC Cabac Bin Count			
		Total number of BINs in the bitstream output per frame from the encoder. This count is updated			
for every time the bin counter is incremented and its reset at image start.					



#### **TIMESTAMP - Reported Timestamp Count**

Register Space:	MMIO: 0/2/0
Source:	RenderCS
Default Value:	0x0000000, 0x0000000
Access:	RO. This register is not set by the context restore.
Size (in bits):	64
Address:	02358h

This register provides an elapsed real-time value that can be used as a timestamp for GPU events over short periods of time. Note that the value of this register can be obtained in a 3D pipeline-synchronous fashion without a pipeline flush by using the PIPE\_CONTROL command. See 3D Geometry Pipeline in the "3D and Media" volume. This register (effectively) counts at a constant frequency by adjusting the increment amount according to the actual reference clock frequency. SW therefore does not need to know the reference clock frequency. This register is not reset by a graphics reset. It will maintain its value unless a full chipset reset is performed.

DWord	Bit	Description			
0	63:36	Reserved			
		Format:	N	/IBZ	
	35:0	Timestamp Value			
		Format: U36			
		Description			
		This register toggles every 80 ns. The	e upper 28 bits	are zero.	



# **RSTFCTLMSG - Reset Flow Control Messages**

			-				
Register	Space:	MMIO: 0/2/0					
Default \	/alue:	0x0000000					
Size (in b	oits):	32					
Address:		08108h					
Soft-Res	et and	FLR Flow Control Message Registers					
DWord	Bit	Desci	ription				
0	31:16	Message Mask					
		Access:	RO				
		Message Mask	·				
		In order to write to bits 15:0, the correspond For example, for bit 14 to be set, bit 30 need	5 5				
	15:8	Reserved					
		Access:	RO				
		Reserved					
	7	Blitter Reset Flow Acknowledgement Messag	les				
		Access:	R/W				
		PM Acknowledgement Messages for Blitter	reset:				
		'1': PREP_RST_BLIT_ACK					
		- Acknowledgement that graphics blitter is prepared for reset assertion.					
		'0': DONE_BLIT_RST_ACK					
		- Acknowledgement that graphics blitter reset is de-asserted					
	6	Media Reset Flow Acknowledgement Messag	jes				
		Access:	R/W				
		PM Acknowledgement Messages for Media	reset:				
		'1': PREP_RST_MEDIA_ACK					
		- Acknowledgement that graphics media block is prepared for reset assertion.					
		'0': DONE_MEDIA_RST_ACK					
		- Acknowledgement that the graphics media reset is de-asserted					
	5	Render Reset Flow Acknowledgement Messages					
		Access:	R/W				
		PM Acknowledgement Messages for Render reset:					
		'1': PREP_RST_RENDER_ACK					
		- Acknowledgement that the graphics rende	er block is prepared for reset assertion.				
		'0': DONE_RENDER_RST_ACK - Acknowledgement that the graphics render reset is de-asserted					
		- Acknowledgement that the graphics rende					



# **RSTFCTLMSG - Reset Flow Control Messages**

4	GTI-Device Reset Flow Acknowledgement Messages						
	Access: R/W						
	PM Acknowledgement Me	essages for GTI-Device reset:					
	'1': PREP_RST_GTIDEV_ACK	Κ					
	- Acknowledgement that t	he GTI device is prepared for reset assertion.					
	'0': DONE_GTIDEV_RST_AC						
	- Acknowledgement that t	he GTI device reset is de-asserted					
3:2	Reserved						
	Access:	RO					
	Reserved.						
1	Global Resource Arbitration	n Acknowledgement Messages					
	Access:	R/W					
	Global Resource Arbitratio	n Acknowledgement Message from PM:					
	'1': CP_ARB_REQ_ACK - Acknowledgement for CPunit's global resource arbitration request						
	'0': CP_ARB_RELEASE_ACK - Acknowledgement to CPunit's release of global resources						
0	CP Busy / Idle Status Acknowledgement Messages						
	Access: R/W						
	CP Busy / Idle Status Acknowledgement Message from PM:						
	'0': CP_NOT_BUSY_ACK - Acknowledgement that the CPunit is idle.						
	'1': CP_BUSY_ACK - Acknowledgement that the CPunit is busy.						



#### RING\_BUFFER\_HEAD\_PREEMPT\_REG -RING\_BUFFER\_HEAD\_PREEMPT\_REG

Register Space:	MMIO: 0/2/0
Source:	BSpec
Default Value:	0x0000000
Access:	R/W
Size (in bits):	32
Address:	0214Ch
Name:	RCS RING_BUFFER_HEAD_PREEMPT_REG
ShortName:	RCS RING BUFFER HEAD PREEMPT REG

This register contains the Head pointer offset in the ring when the last PREEMPTABLE command was executed and caused the head pointer to move due to the UHPTR register being valid. If the PREEMPTABLE command is executed as part of the batch buffer then the value of the register will be the offset in the ring of the command past the batch buffer start that contained the preemptable command.

This is a global register and context save/restored as part of power context image.

	Programming Notes						
-	Programming Restriction: This register should NEVER be programmed by driver. This is for HW internal use only.						
DWord	Bit			Description			
0 31:21 <b>Reserved</b>							
		Forma	t:	MBZ			
	20:2	Preem	oted He	ad Offset			
		Forma	t:	U19			
		This field contains the Head pointer offset in the ring when the last MI_ARB_CHEC command was executed and caused the head pointer to move due to the UHPTR register being valid.					
	1:0		atch In				
		Forma	t:	Enabled			
		Value	Name	Description			
		0h	Ring	Preemptable command was executed in ring and caused head pointer to be updated.			
		1h	Batch	Preemptable command was executed in batch and caused head pointer to be updated.			



#### **RING\_BUFFER\_CTL - Ring Buffer Control**

Register Space:	MMIO: 0/2/0
Source:	BSpec
Default Value:	0x0000000
Access:	R/W
Size (in bits):	32
Address:	0203Ch
Name:	RCS Ring Buffer Control
ShortName:	RCS_RING_BUFFER_CTL
Address:	1203Ch
Name:	VCS Ring Buffer Control
ShortName:	VCS_RING_BUFFER_CTL
Address:	2203Ch
Name:	BCS Ring Buffer Control
ShortName:	BCS_RING_BUFFER_CTL
	Description
instructions to the ring buffer is defined	e used to define and operate the ring buffer mechanism which can be used to pass e command interface. The buffer itself is located in a physical memory region. The ned by a 4 Dword register set that includes starting address, length, head offset, tail

offset, and control information. Refer to the Programming Interface chapter for a detailed description of the parameters specified in this ring buffer register set, restrictions on the placement of ring buffer memory, arbitration rules, and in how the ring buffer can be used to pass instructions.

# Ring Buffer Head and Tail Offsets must be properly programmed before it is enabled. A Ring Buffer can be enabled when empty.

Graphics Engine doesn't go IDLE when head offset is not equal to tail offset when ring buffer is disabled. PSMI controller waits for HW to go Idle as part of the PSMI flow. When PSMI flow happens in middle of ring buffer initialization where in Head offset is not equal to Tail offset and Ring Buffer disabled, PMSI flow will hang waiting for Graphics Engine to go IDLE. (During ring buffer initialization SW programs Head and Tail offsets prior to enabling Ring Buffer). In order to avoid this dead lock PSMI controller must detect this case and program head and tail offset to be equal to allow Graphics Engine to go IDLE, before exiting PSMI flow original head and tail offsets should be restored.

DWord	Bit	Description					
0	31:21	Reserved					
		Format: MBZ					
	20:12	Buffer Length					
		Format: U9-1 in 4 KB pages - 1					
		This field is written by SW to specify the length of the ring buffer in 4 KB Pages.					
		Range = [0 = 1 page = 4 KB, 1FFh = 512 pages = 2 MB]					
		Value Name Description					
		0 1 page = 4 KB					



# **RING\_BUFFER\_CTL** - **Ring Buffer Control**

	1FFh				512 pages =	2 MB	
11	RBWait	1					
	Description						
	Indicates that this ring has executed a WAIT_FOR_EVENT instruction and is						
	currently waiting. Software can write a "1" to clear this bit, write of "0" has no effect. When the RB is waiting for an event and this bit is cleared, the						
		wait will be terminated and the RB will be returned to arbitration.					
	RenderCS: RBWait is not set on executing WAIIT_FOR_EVENT instruct waiting on Async Flip Pending.					OR_EVENT instruction	
10	Semaph	Semaphore Wait					
				D	escription		
			5		ed a MI_SEMAPH ently waiting.	HORE_MBOX instruction	
9	Reserve	ed					
	Format	:				MBZ	
8	Reserve	ed				1	
	Format	Format:			MBZ		
7:3	Reserve	ed				1	
	Format	:				MBZ	
2:1		atic Repor	i i				
	Source:			BlitterCS, VideoCS			
	Exists I			//BCS	, VCS		
				D	escription		
	This fie	ld is writte	n by softw	/are to	control the auto	omatic "reporting" (write)	
		-				er DWord 1) to the	
					e Hardware Statu or enabled at 4KE	s Page. Automatic	
		aries within				, 0 110 01 12010	
	Value	1	Name			Description	
	0	MI_AUTO	REPORT_O	FF	Automatic repor	ting disabled	
	1	MI_AUTO	REPORT_64	4KB	Report every 16	pages (64KB)	
	2 MI_AUTOREPOR		REPORT_4	KB	Report every page	-	
						not be enabled in Ring Buffe ling to minimize the auto rep	
	3	MI_AUTO	REPORT_12	28KB	Report every 32	pages (128KB)	
	3 MI_AUTOREPORT_128KB Report every 32 pages (128KB) Automatic Report Head Pointer						
2:1	Automa	atic Repor	<u>t Head Pc</u>	ointer			



# **RING\_BUFFER\_CTL** - Ring Buffer Control

		1			1	
		Exists If:	/	/RCS		
			5	l the automatic "reporting" (write) of th		
		ring buffer's "Head Pointer" register (register DWord 1) to the corresponding location				
		within the Hardware Status Page. Automatic reporting can either be disabled or enabled at 4KB, 64KB or 128KB boundaries within the ring buffer.				
		Value	Name	Description		
	0h		MI_AUTOREPORT_OFF	Automatic reporting disabled		
		1h	Reserved	Reserved		
		2h	Reserved			
		3h	MI_AUTOREPORT_128KB	Report every 32 pages (128KB)		
			Progran	nming Notes		
-	0	Ring Buf	fer Enable			
		Format:		Enable		
		This field	is used to enable or disable this	s ring buffer. It can be enabled or disabl	ed	
		regardless of whether there are valid instructions pending. If disabled and the ring				
		head equals ring tail, all state currently loaded in hardware is considered invalid.				
		Programming Notes				
		-	fer Mode of Scheduling:			
			t follow the below programming	g notes during SW initialization ring buffer for the first time, this		
			e coming out of boot, standby,	-		
			N must sat the Force Wakeup h	it to provent GT from entering C6		
		SW must set the Force Wakeup bit to prevent GT from entering C6.				
		<ul> <li>SW must dispatch workload (dummy context) to initialize render engine with default state such that any context switches that occur</li> </ul>				
		subsequently (Power Save) will save and restore coherent device				
		st	ate. Indirect pointers used in 3D	O states must point to valid		
		-	raphics surface existing in mem			
	PP_DIR_BASE register should be programmed as part of init workload if PPGTT is enabled in GFX_MODE register.					
	<ul> <li>SW must ensure all the register (MMIO) initialization/pro</li> </ul>					
		st	arough CPU happens in this bloc ate is save/restored on subsequ equences).	ck or latter, this ensures the MMIO lent context switches (Power		
			nce the render engine is progra	mmed with valid state and the		
				should be reset to enable C6 entry.		
		<u> </u>				



# **BCS\_RCCID** - Ring Buffer Current Context ID Register

Register Space:	MMIO: 0/2/0				
Source:	BlitterCS				
Default Value:	0x0000000, 0x000	00000			
Access:	R/W				
Size (in bits):	64				
Address:	22190h-22197h				
This register contain	s the current ring co	ntext ID associated with the ring buffer.			
		Programming Notes			
	The current context registers must not be written directly (via MMIO). The RCCID register should only be updated indirectly from RNCID.				
DWord	Bit	Description			
0	63:0	Unnamed See Context Descriptor for BCS.			



#### **RING\_BUFFER\_HEAD** - Ring Buffer Head

Register Space:	MMIO: 0/2/0
Source:	BSpec
Default Value:	0x0000000
Access:	R/W
Size (in bits):	32
Address:	02034h
Name:	RCS Ring Buffer Head
ShortName:	RCS_RING_BUFFER_HEAD
Address:	12034h
Name:	VCS Ring Buffer Head
ShortName:	VCS_RING_BUFFER_HEAD
Address:	22034h
Name:	BCS Ring Buffer Head
ShortName:	BCS_RING_BUFFER_HEAD

This register is used to define and operate the ring buffer mechanism which can be used to pass instructions to the command interface. The buffer itself is located in a physical memory region. The ring buffer is defined by a 4 Dword register set that includes starting address, length, head offset, tail offset, and control information. Refer to the Programming Interface chapter for a detailed description of the parameters specified in this ring buffer register set, restrictions on the placement of ring buffer memory, arbitration rules, and in how the ring buffer can be used to pass instructions.

# Ring Buffer Head Offsets must be properly programmed before ring is enabled. A Ring Buffer can be enabled when empty.

DWord	Bit		Description				
0	31:21	Wrap Count					
		Format:	U11 count of ring buffer wraps				
		This field is incremented by 1 whenever the <b>Head Offset</b> wraps from the end of the buffer back to the start (i.e., whenever it wraps back to 0). Appending this field to the <b>Head Offset</b> field effectively creates a virtual 4GB Head "Pointer" which can be used as a tag associated with					
		instructions placed	in a ring buffer. The Wrap Count itself will wrap to 0 upon overflow.				
	20:2	Head Offset					
		Format: GraphicsAddress[20:2] DWord Offset					
		this field to select th while the RB is enab	the offset of the <i>next</i> instruction DWord to be parsed. Software will initialize the first DWord to be parsed once the RB is enabled. (Writing the Head Offset bled is UNDEFINED). Subsequently, the device will increment this offset as it is - until it reaches the QWord specified by the <b>Tail Offset</b> . At this point the lered "empty".				
		Programming Notes					
		A RB can be enable	ed empty or containing some number of valid instructions.				
	1	Reserved					



# **RING\_BUFFER\_HEAD** - Ring Buffer Head

	Format:			MBZ	
0	Wait for Condition Indicator				
	Source:		RenderCS		
	Exists If:		//RCS		
	This is a read only value used to waiting for a conditional code t			e command streamer is currently	
0	Reserved				
	Source:	BlitterCS, VideoCS			
	Exists If:	//BCS, VCS			
	Format:	MBZ			



# **RING\_BUFFER\_START** - **Ring Buffer Start**

Register Sp	bace:	MMIO: 0/2/0						
Source:		BSpec	BSpec					
Default Va	lue:	0x00000000	0x0000000					
Access:		R/W						
Size (in bit	s):	32						
Address:		02038h	02038h					
Name:		RCS Ring Buffer	Start					
ShortName	e:	RCS_RING_BUFF	ER_START					
Address:		12038h						
Name:		VCS Ring Buffer	Start					
ShortName	e:	VCS_RING_BUFF	ER_START					
Address:		22038h						
Name:		BCS Ring Buffer	Start					
ShortName	e:	BCS_RING_BUFF	ER_START					
instruction defined by informatio this ring bu	s to the a 4 Dwo n. Refer uffer reg	are used to define and operate the "ring buffer" mechanism which can be used to pass the command interface. The buffer itself is located in a physical memory region. The ring buffer word register set that includes starting address, length, head offset, tail offset, and control er to the Programming Interface chapter for a detailed description of the parameters specified egister set, restrictions on the placement of ring buffer memory, arbitration rules, and in how t be used to pass instructions.						
DWord	Bit		Description					
0	31:12	Starting Address	-					
		Format:	GraphicsAddress[31:12]RingBuffer					
		This field specifies Bits 31:12 of the 4KB-aligned starting Graphics Address of the ring buffe Address bits 31 down to 29 must be zero.						
			ges must map to Main Memory (unca	ched) pages. Ring Buffer addresses				
			ted through the global GTT.					
	11:0	Reserved						
		Format:		MBZ				



#### **RING\_BUFFER\_TAIL** - Ring Buffer Tail

Register	Space:	MMIO: 0/2/0					
Source:		BSpec					
Default \	Value:	0x0000000					
Access:		R/W					
Size (in b	oits):	32					
Address:		02030h					
Name:		RCS Ring Buffer Tail					
ShortNa	me:	RCS_RING_BUFFER_TAIL					
Address:		12030h					
Name:		VCS Ring Buffer Tail					
ShortNa	me:	VCS_RING_BUFFER_TAIL					
Address:	:	22030h					
Name:		BCS Ring Buffer Tail					
ShortNa	me:	BCS_RING_BUFFER_TAIL					
instruction defined informat Refer to	ons to t by a 4 [ ion. the Pro	are used to define and operate the "ring buffer" mechanism which can be used to pass the command interface. The buffer itself is located in a linear memory region. The ring buffer Dword register set that includes starting address, length, head offset, tail offset, and control ogramming Interface chapter for a detailed description of the parameters specified in this rin set, restrictions on the placement of ring buffer memory, arbitration rules, and in how the rin	ng				
instruction defined informat Refer to buffer re buffer ca Ring Bu when em	ons to t by a 4 I ion. the Pro gister s an be us ffer Tail npty.	the command interface. The buffer itself is located in a linear memory region. The ring buffer Dword register set that includes starting address, length, head offset, tail offset, and control ogramming Interface chapter for a detailed description of the parameters specified in this rin set, restrictions on the placement of ring buffer memory, arbitration rules, and in how the rin sed to pass instructions. I Offsets must be properly programmed before ring is enabled. A Ring Buffer can be enabled	ng g				
instruction defined informat Refer to buffer re buffer ca Ring Bu when em <b>DWord</b>	ons to t by a 4 I cion. the Pro egister s an be us ffer Tail npty. Bit	the command interface. The buffer itself is located in a linear memory region. The ring buffer Dword register set that includes starting address, length, head offset, tail offset, and control ogramming Interface chapter for a detailed description of the parameters specified in this rin set, restrictions on the placement of ring buffer memory, arbitration rules, and in how the rin sed to pass instructions. I Offsets must be properly programmed before ring is enabled. A Ring Buffer can be enabled Description	ng g				
instruction defined informat Refer to buffer re buffer ca Ring Bu when em	ons to t by a 4 I ion. the Pro gister s an be us ffer Tail npty.	the command interface. The buffer itself is located in a linear memory region. The ring buffer Dword register set that includes starting address, length, head offset, tail offset, and control ogramming Interface chapter for a detailed description of the parameters specified in this rin set, restrictions on the placement of ring buffer memory, arbitration rules, and in how the rin sed to pass instructions. I Offsets must be properly programmed before ring is enabled. A Ring Buffer can be enabled Description Reserved	ng g				
instruction defined informat Refer to buffer re buffer ca Ring Bu when em <b>DWord</b>	ons to t by a 4 I cion. the Pro egister s an be us ffer Tail npty. Bit	the command interface. The buffer itself is located in a linear memory region. The ring buffer Dword register set that includes starting address, length, head offset, tail offset, and control ogramming Interface chapter for a detailed description of the parameters specified in this rin set, restrictions on the placement of ring buffer memory, arbitration rules, and in how the rin sed to pass instructions. I Offsets must be properly programmed before ring is enabled. A Ring Buffer can be enabled Description	ng g				
instruction defined informat Refer to buffer re buffer ca Ring Bu when em <b>DWord</b>	ons to t by a 4 I cion. the Pro egister s an be us ffer Tail npty. Bit	the command interface. The buffer itself is located in a linear memory region. The ring buffer Dword register set that includes starting address, length, head offset, tail offset, and control ogramming Interface chapter for a detailed description of the parameters specified in this rin set, restrictions on the placement of ring buffer memory, arbitration rules, and in how the rin sed to pass instructions. I Offsets must be properly programmed before ring is enabled. A Ring Buffer can be enabled <b>Description</b> <b>Reserved</b> Format: <b>MBZ</b>	ng g				
instruction defined informat Refer to buffer re buffer ca Ring Bu when em <b>DWord</b>	ons to t by a 4 [ ion. the Pro- gister s an be us ffer Tail npty. Bit 31:21	the command interface. The buffer itself is located in a linear memory region. The ring buffer Dword register set that includes starting address, length, head offset, tail offset, and control ogramming Interface chapter for a detailed description of the parameters specified in this rins set, restrictions on the placement of ring buffer memory, arbitration rules, and in how the rins sed to pass instructions. I Offsets must be properly programmed before ring is enabled. A Ring Buffer can be enabled <b>Description</b> <b>Reserved</b> Format: MBZ <b>Tail Offset</b> Format: GraphicsAddress[20:3]	ng g d				
instruction defined informat Refer to buffer re buffer ca Ring Bu when em <b>DWord</b>	ons to t by a 4 [ ion. the Pro- gister s an be us ffer Tail npty. Bit 31:21	the command interface. The buffer itself is located in a linear memory region. The ring buffer Dword register set that includes starting address, length, head offset, tail offset, and control ogramming Interface chapter for a detailed description of the parameters specified in this rin set, restrictions on the placement of ring buffer memory, arbitration rules, and in how the rin sed to pass instructions. I Offsets must be properly programmed before ring is enabled. A Ring Buffer can be enabled <b>Description</b> <b>Reserved</b> Format: <b>MBZ</b>	ng g d ffer ner				
instruction defined informat Refer to buffer re buffer ca Ring Bu when em <b>DWord</b>	ons to t by a 4 [ ion. the Pro- gister s an be us ffer Tail npty. Bit 31:21	the command interface. The buffer itself is located in a linear memory region. The ring buffer Dword register set that includes starting address, length, head offset, tail offset, and control ogramming Interface chapter for a detailed description of the parameters specified in this rin- set, restrictions on the placement of ring buffer memory, arbitration rules, and in how the rin- sed to pass instructions. I Offsets must be properly programmed before ring is enabled. A Ring Buffer can be enabled Format: Tail Offset Format:	ng g d ffer ner				



# **TLBPEND\_SEC0 - Section 0 of TLBPEND Entry**

Register	Space:	MMIO: 0/2/0			
Source:		RenderCS			
Default \	/alue:	0x0000000			
Access:		R/W			
Size (in b	oits):	32			
Trusted	Туре:	1			
Address:	: 04400h-04403h				
This regi	ster is c	directly mapped to the TLBPEND Array in the Graphic Arbiter.			
DWord	Bit	Description			
0	31	<b>vtstatus</b> This bit will be used in conjunction with the ready bit to determine the stage of the translation. See table below.			
	30:28	<b>T bits</b> 5 3:1 of the GTT entry used to translate the Virtual Address. 000 if translation is pending.			
	27:0	<b>Current address</b> The value of this field depends on the stage of the TLB translation for this entry: VA - bits 27:20 = 00, bits 19:0 = Bits 31:12 of the Virtual Address of the cycle.			



# **TLBPEND\_SEC1 - Section 1 of TLBPEND Entry**

Register	Space	MMIO: 0/2/0					
Source:		RenderCS					
Default V	Value:	0x0000000					
Access:		R/W					
Size (in b	bits): 32						
Trusted	Trusted Type: 1						
Address:	:	04500h-04503h					
		directly mapped to the cur he for Z (Depth), Hi Z, and	rent Virtual Addresses in the MTTLB (Texture and constant cache Stencil TLB).				
DWord	Bit		Description				
0	31:28	Current address					
		Bits 9:6 of the Virtual Add	ress of the cycle.				
	27:24	<b>Cacheability Control Bits</b>	5				
		Bits 3:1 of the GTT entry are used to translate the Virtual Address. 000 if translation is					
		pending.					
		3 Reserved					
		2 Graphics Data Type (GFDT). This field contains the GFDT bit for this surface when					
		writes occur. GFDT can	also be set by the GTT. The effective GFDT is the logical OR of				
		this field with the GFD	I from the GTT entry. This field is ignored for reads.				
		1:0 <b>Cacheability Control.</b> This field controls cacheability.					
			ntrol bits from GTT entry.				
		01: Data is not cached.					
		11: Data is cached.					
	23	ZLR bit					
		Flag to indicate this is a ze	ero length read, a read used to calculate a physical address for a write.				
22:4							
	22:4	TAG					
		Cycle identification TAG.					
	22:4 3:0	Cycle identification TAG. SRC ID					
		Cycle identification TAG. <b>SRC ID</b> Encoding of unit generation	ng this cycle.				
		Cycle identification TAG. SRC ID Encoding of unit generativ Value	ng this cycle.				
		Cycle identification TAG. SRC ID Encoding of unit generatin Value 0000b	ng this cycle.   Name  CS_RD_SRCID				
		Cycle identification TAG. SRC ID Encoding of unit generation Value 0000b 0001b	ng this cycle. Name CS_RD_SRCID VF_RD_SRCID				
		Cycle identification TAG. SRC ID Encoding of unit generation Value 0000b 0001b 0010b	ng this cycle.				
		Cycle identification TAG. SRC ID Encoding of unit generation Value 0000b 0001b	ng this cycle. Name CS_RD_SRCID VF_RD_SRCID				
		Cycle identification TAG. SRC ID Encoding of unit generation Value 0000b 0001b 0010b	ng this cycle.				
		Cycle identification TAG. SRC ID Encoding of unit generatin Value 0000b 0001b 0010b 0011b	ng this cycle.				
		Cycle identification TAG. SRC ID Encoding of unit generation Value 0000b 0001b 0010b 0011b 0100b	ng this cycle.          Name         CS_RD_SRCID         VF_RD_SRCID         ISC_SRCID         MT_SRCID         RCC_SRCID				



# **TLBPEND\_SEC1 - Section 1 of TLBPEND Entry**

		1000b	CS_WR_SRCID
	1001b	MBC_SRCID	
		1010b	CS_RD_PROBE
		1011b	CS_RD_PWRCTX
		1100b	RC_R4WRCMP
		1101b	RESRVD2_SRCID
		1110b	RESRVD1_SRCID
		1111b	RESRVD0_SRCID



# **TLBPEND\_SEC2 - Section 2 of TLBPEND Entry**

Register Space	e: N	1MIO: 0/2/0			
Source:	R	RenderCS			
Default Value:	: 0:	0x0000000			
Access:	R	/W			
Size (in bits):	3.	2			
Trusted Type:	1				
Address:	04	4600h-04603h			
This register is	s directly r	directly mapped to the current Virtual Addresses in the MTTLB (Texture and constant cache TLB).			
DWord	Bit		Description		
0	31:14	Reserved			
	13	<b>Big Page Attribu</b> This entry is using			
	12:8	Current Address			
		Format:	GraphicsAddress[14:10]		
		Bits 14:10 of the Virtual Address of the cycle.			
	7:0	PAT Entry			
		Location of Physi	cal Address in Physical Address Table.		



### SO\_NUM\_PRIMS\_WRITTEN[0:3] - Stream Output Num Primitives Written Counter

Register Space:		MMIO: 0/2/0				
Source:		RenderCS				
Default Value:		0x0000000, 0x0000000				
Access:		R/W				
Size (in bits):		64				
Address:		05200h-0521Fh				
There is one 64-bit register for each of the 4 supported streams:5200h-5207h SO_NUM_PRIMS_WRITTEN0 (for Stream Out Stream #0)5208h-520Fh SO_NUM_PRIMS_WRITTEN1 (for Stream Out Stream #1)5210h-5217h SO_NUM_PRIMS_WRITTEN2 (for Stream Out Stream #2)5218h-521Fh SO_NUM_PRIMS_WRITTEN3 (for Stream Out Stream #3)These registers are used to count the number of primitives (aka objects: points, lines, triangles) which the SO stage has successfully written to a particular "stream's" Streamed Vertex Output buffers, subject to buffer overflow detection. (See the Stream Output section of the 3D pipeline volume). These registers are part of the context save and restore.						
DWord	Bit	Description				
0	63:0	Num Prims Written Count				
		Format: U64				
		This count is incremented (by one) every time a GS thread outputs a DataPort Streamed Ve Buffer Write message with the Increment Num Prims Written bit set in the message header the Geometry Shader and Data Port chapters in the 3D Volume.)				



# SO\_PRIM\_STORAGE\_NEEDED[0:3] - Stream Output Primitive Storage Needed Counters

Register Space:	MMIO: 0/2/0
Source:	RenderCS
Default Value:	0x0000000, 0x0000000
Access:	RO. This register is set by the context restore
Size (in bits):	64

Address: 05240h-0525Fh

There is one 64-bit register for each of the 4 supported streams:

5240h-5247h SO\_PRIM\_STORAGE\_NEEDED0 (for Stream Out Stream #0) 5248h-524Fh SO\_PRIM\_STORAGE\_NEEDED1 (for Stream Out Stream #1) 5250h-5257h SO\_PRIM\_STORAGE\_NEEDED2 (for Stream Out Stream #2) 5258h-525Fh SO\_PRIM\_STORAGE\_NEEDED3 (for Stream Out Stream #3)

These registers are used to count the number of primitives (aka objects: points, lines, triangles) which the SO stage has or would have written to a particular "stream's" Streamed Vertex Output buffers if all buffers had been large enough to accommodate the writes. (See the Stream Output section of the 3D pipeline volume).

These registers are part of the context save and restore.

DWord	Bit	Description		
0	63:0	Prim Storage Needed Count		
		Format:	U64	
		This count is incremented (by one) by the SOL stage for each object (point, line, triangle) it writes or attempts to write to the corresponding stream's output buffers. The count is not affected by the actual number of buffers bound to the stream.		



# **SO\_WRITE\_OFFSET[0:3] - Stream Output Write Offsets**

		Format: MBZ				
	1:0	Reserved				
0	31:2	Format:U30This field contains a DWord offset from the corresponding SO buffer's Base Address value. The SOL stage uses this value as a write offset when performing writes to the buffer. The SOL stage will increment this value as a part of performing stream output to the buffer. Note that the SOL stage uses the buffer's Surface Pitch to advance the Write Offset, without regard to the buffer's Base Address (see Programming Notes above).				
DWord	Bit	Description				
• 1	he po The SC vritter vould	are must ensure that no HW stream output operations can be in process or otherwise pending at int that the MI_LOAD/STORE commands are processed. This will likely require a pipeline flush. OL stage will effectively advance the write offset by the buffer's Surface Pitch after each vertex is a (assuming no overflow is detected in any targetted SO buffer). Under "normal" conditions one expect software to initialize the WriteOffset to some (possibly zero) multiple of Surface Pitch in to align vertex writes to the buffer's Base Address, though it is not required to do so.				
Programming Notes						
5280h- 5284h- 5288h- 528Ch- These r Output will incr memor	5283F 5287F 528BF 528FF egiste Buffe emer y via	R/W 32-bit register for each of the 4 supported stream output buffer slots: a SO_WRITE_OFFSET0 (for Stream Out Buffer #0) a SO_ WRITE_OFFSET1 (for Stream Out Buffer #1) a SO_ WRITE_OFFSET2 (for Stream Out Buffer #2) a SO_ WRITE_OFFSET3 (for Stream Out Buffer #3) ers are used to set and track a DWord-granular Write Offset for each of the 4 Stream er slots. Software can directly write them via MI_LOAD_REGxxx commands. The SOL stage at them as part of stream output processing. Software can cause them to be written to MI_STORE_REGxxx commands. (See the Stream Output section of the 3D pipeline volume). rs are part of the context save and restore.				
Address		05280h-0528Fh				
Size (in bits):		32				
Default Value: Access:		0x00000000 RW. This register is set by the context restore.				
Source:		RenderCS				
Register	Space	e: MMIO: 0/2/0				



#### **FF\_MODE - Thread Mode Register Register Space:** MMIO: 0/2/0 Source: RenderCS Default Value: 0x28A01010 Access: R/W Size (in bits): 32 020A0h Address: This register is used to program the FF shader Mode. **DWord** Bit Description 0 31 Reserved Format: MBZ 30 Reserved Format: MBZ 29:26 **DS Hit Max Value** U4 Format: **Description** If the number of hits reaches the DS Hit Max Value and there is a pending miss to be dispatched, the DS will dispatch the pending miss vertex as a single dispatch. Programming the value beyond the range will have undefined behavior. Value Name 10 [Default] [1,11] 25:20 VS Hit Max Value U6 Format: Description If the number of hits reaches the VS Hit Max Value and there is a pending miss to be dispatched, the VS will dispatch the pending miss vertex as a single dispatch. Programming the value beyond the range will have undefined behavior. Value Name 10 [Default] [1,58] 19 **DS Reference Count Full Force Miss Enable** Format: Enable Value Name **Description**



## **FF\_MODE - Thread Mode Register**

	1	1		
	0b	[Default]		hit to the DS cache and the associated handle's reference count is ful stall until a derefernce.
	1b			hit to the DS cache and the associated handle's reference count is fu force the cycle as a miss and allocate a new handle.
18:17	TS Thre	ead Dispate	ch Mo	de
	Forma	t:		U2
	Value	Name		Description
	0h	Load Balanced [ <b>Default]</b>	t	Thread Dispatch will load balance the half slices of the threads. Note: his will cause possible corruption if input handles are reused due to nstancing or topologies that reuse vertices(i.e. strips and fans)
	1h	Half Slice (	) 4	All threads will be dispatched to Half Slice 0.
	2h	Half Slice 1	All threads will be dispatched to Half Slice 1.	
	3h	Reserved		
16	5 TS Thread Dispatch Override Enable		erride Enable	
	Forma	t:		Enable
				1
	Value	Nam		Description
	0h	_	efault	Hardware will decide which half slice the thread will dispatch.
	1h	Enable		The value in the TS Thread Dispatch Mode will be used for dispate
15	VS Ref	erence Cou	nt Ful	l Force Miss Enable
	Forma	t:		U1
	Value	Name		Description
	[0,1]			
	0b	[Default]		hit to the VS cache and the associated handle's reference count is fu stall until a derefernce.
	1b			hit to the VS cache and the associated handle's reference count is fu force the cycle as a miss and allocate a new handle.
14:13	VS Thr	ead Dispat	ch Mo	de
	Forma	t:		U2
	Value	Name		Description
	Oh	Load		•
	UII	Balanced		hread Dispatch will load balance the half slices of the threads. Note: his will cause possible corruption if input handles are reused due to
		[Default]		nstancing or topologies that reuse vertices (i.e. strips and fans)
	1h	Half Slice (	) /	All threads will be dispatched to Half Slice 0.
	2h	Half Slice 1	LA	All threads will be dispatched to Half Slice 1.



12	VS Thre	ead Dispatch Ov	verr	ide Enable		
	Format:			Enable		
					• .•	
	Value	Name Disable			scriptio	
	0h 1h	Enable [Default]		Hardware will decide which half The value in the VS Thread Disp		•
11:7	Reserve	1 -				
11./	Format				MBZ	
6:5		ead Dispatch Mo	ode	I		
	Format: U2			U2		
	Value	Name	De		cription	
	0h	Balanced	Thread Dispatch will load balance the half slices of the threads. Note this will cause possible corruption if input handles are reused due instancing or topologies that reuse vertices (i.e., strips and fans).			nandles are reused due to
	1h	Half Slice 0	All t	threads will be dispatched to Ha	alf Slice	0.
	2h	Half Slice 1	All t	threads will be dispatched to Ha	patched to Half Slice 1.	
	3h	Reserved				
4	DS Thr	ead Dispatch Ov	verr	ride Enable		
	Format	:		Enable		
	Value	Name		De	scriptio	n
	0h	Disable	ŀ	Hardware will decide which half		
	1h	Enable <b>[Default</b>	<b>t]</b> 1	The value in the DS Thread Disp	atch Mo	ode will be used for dispatcl
2.0	Recerve	ed				
3:0	ILCSCI V	Reserved				



## **TLB\_RD\_ADDR - TLB\_RD\_ADDRESS Register**

Register Sp	ace:	MMIO: 0/2/0						
Default Valu		0x00000000						
Size (in bits)	):	32 04700h-04703h						
Address:								
TLB Read Address								
DWord	Bit			Descri	ption			
0	31:10	Reserved						
		Default Value:		000000000000	0000000000b			
		Access:		RO				
		Reserved Bits						
	9:0	TLB Read Address						
		Default Value:			00000000b			
		Access:			R/W			
		TLB Read Address						
		<u>MSB&lt;9:X&gt; :</u>						
		TLB Select	<9:X>	PAT MSB: Sectior	n of the PAT used.			
		PAT_MSB_VLFTLB	00000	32 entries - 32				
		PAT_MSB_CVSTLB	00001	32 entries - 32				
		PAT_MSB_RCCTLB	0001	64 entries - 64				
		PAT_MSB_ZTLB	001	128 entries - 128				
		PAT_MSB_L3TLB	01	160 entries - 256				
		PAT_MSB_CASCTLB	10	140 entries - 256				
		LSB <x:0> :</x:0>						
		GEN RAM ADDRES	in Sele	ected TLB				



#### **TLB\_RD\_DATA - TLB\_RD\_DATA Register**

Register Space:MMIO: 0/2/0Default Value:0x00000000

Size (in bits): 32

Address: 04704h-04707h

#### TLB\_READ\_DATA Register

DWord	Bit		Description	
0	31:0	TLB_READ_DATA Register		
		Default Value:	0000000h	
		Access:	RO	
		Return data		



## **TLB064\_VA - TLB064\_VA Virtual Page Address Registers**

Register Space:	MMIO: 0/2	/0		
Source:	VideoCS			
Default Value:	0x0000000	0		
Access:	RO			
Size (in bits):	32			
Trusted Type:	1			
Address:	14800h-14	803h		
This register is direc	tly mapped t	o the current Vir	tual Addresses in the TLB064 (V	CS and VMC TLB).
DWord	Bit	Description		
0	31:12	Address		
		Format:	GraphicsAddress[31:12]	
		Page virtual a	ddress.	
	11:0	Reserved		
		Format:		MBZ



## TLB132\_VA - TLB132\_VA Virtual Page Address Registers

Register Space:	MMIO: 0/2/	0			
Source:	VideoCS				
Default Value:	0x0000000	)			
Access:	RO				
Size (in bits):	32				
Address:	14900h-149	03h			
These registers are Default Value = UU	• • • •		Virtual Addresses in	the TLB132	2 (All The Media Clients TLB).
DWord	Bit		Des	scription	
0	31:12	31:12 Address			
		Format:	GraphicsAddre	ess[31:12]	
		Page virtual ad	dress.		
	11:0	Reserved			
		Format:			MBZ



# TLB232\_VA - TLB232\_VA Virtual Page Address Registers

Register Space:	MMIO: 0/2	/0			
Source:	VideoCS				
Default Value:	0x0000000	0			
Access:	RO				
Size (in bits):	32				
Trusted Type:	1				
Address:	14A00h-14	4A03h			
This register is direc	tly mapped t	o the current Virt	ual Addresses in the TLB232 (VI	DS and VLF FW TLB).	
DWord	Bit	Description			
0	31:12	Address			
		Format:	GraphicsAddress[31:12]		
		Page virtual ac	ldress.		
	11:0	Reserved			
		Format:		MBZ	



## TLB304\_VA - TLB304\_VA Virtual Page Address Registers

MMIO: 0/2/	0		
VideoCS			
0x0000000	)		
RO			
32			
1			
14B00h-14E	303h		
tly mapped to	the current Virtu	al Addresses in the TLB304 (V	(CR TLB).
Bit		Description	
31:12	1:12 Address		
	Format:	GraphicsAddress[31:12]	
	Page virtual add	lress.	
11:0	Reserved		
	Format:		MBZ
	VideoCS 0x00000000 RO 32 1 14B00h-14E tly mapped to <u>Bit</u> 31:12	0x0000000 RO 32 1 14B00h-14B03h tly mapped to the current Virtu Bit 31:12 Address Format: Page virtual add 11:0 Reserved	VideoCS 0x0000000 RO 32 1 14B00h-14B03h tly mapped to the current Virtual Addresses in the TLB304 (V Bit Description 31:12 Address Format: GraphicsAddress[31:12] Page virtual address. 11:0 Reserved



## **GFX\_PEND\_TLB - TLBPEND Control Register**

Register	Space:	MMIO: 0/2/0					
Source:		VideoCS					
Default <b>\</b>	Value:	0x0000000					
Access:		R/W					
Size (in b	oits):	32					
Trusted	Туре:	1					
Address		14040h					
Max Out	standi	ng Media pending TLB requests					
DWord	Bit		Description				
0	31	Reserved					
		Format:	MBZ				
	30	Reserved					
		Format:	MBZ				
	29:24	VMX BS Limit Count					
		Format:	U6				
		This is the MAX number of Allowed internal pending read requests which require a TLB read.					
	23	VMC Limit Enable bit					
		Format:	U1				
		This bit is used to enable the pendi	ng TLB requests limitation function for the VMC.				
		When set, the number of internal pending read requests which require a TLB read will not exceed the programmed counter value.					
		not exceed the programmed count	er value.				
	22	Reserved					
		Format:	MBZ				
	21:16	VMC TLB Limit Count					
		Format:	U6				
		This is the MAX number of Allowed internal pending read requests which require a TLB read.					
	15	VMXRS Limit Enable bit					
		Format:	U1				
		This bit is used to enable the pendi	ng TLB requests limitation function for the VMX Row				
		store.					
		When set, the number of internal p	ending read requests which require a TLB read will				
		not exceed the programmed count					
	14	Reserved					
	74						



## **GFX\_PEND\_TLB - TLBPEND Control Register**

	Format:	MBZ	
13:8	VMX RS Random Accsess TLB Limit Count		
	Format:		U6
	This is the MAX number of Allowed internal pending read	requests	which require a TLB read.
7	VCS Limit Enable bit		
	Format:		U1
	This bit is used to enable the pending TLB requests li	mitatio	n function for the Command
	Streamer.		
	When set, the number of internal pending read requi	ests whi	ich require a TLB read will
	not exceed the programmed counter value.		
6	Reserved		
	Format:	MBZ	
5:0	VCS TLB Limit Count		
	Format:		U6
	This is the MAX number of Allowed internal pending read	requests	which require a TLB read.

Look Inside."

			-				
Register	Spa	ace: MMIO: 0/2/0					
Default \	Value	ıe: 0x02800000					
Size (in b	oits):	: 32					
Address:		09400h					
Unit Lev	vel C	Clock Gating Control Registers.					
DWord	Bit	Descripti	ion				
0	31	Sarbunit Clock Gating Disable					
		Access:	R/W				
		SARB unit Clock Gating Disable Control:					
		'0': Clock Gating Enabled. (i.e., clocks can be gate	ed when they are not required to toggle				
		for functionality)	an always)				
		'1': Clock Gating Disabled. (i.e., clocks are togglir	ig, always)				
	30	IEFunit Clock Gating Disable					
		Access:	R/W				
		IEFunit Clock Gating Disable Control:					
		'0': Clock Gating Enabled. (i.e., clocks can be gate	ed when they are not required to toggle				
		for functionality)					
		'1': Clock Gating Disabled. (i.e., clocks are togglir	ng, always)				
	29	IECPunit Clock Gating Disable					
		Access:	R/W				
		IECPunit Clock Gating Disable Control:					
		'0': Clock Gating Enabled. (i.e., clocks can be gate	ed when they are not required to toggle				
		for functionality) '1': Clock Gating Disabled. (i.e., clocks are togglir					
		1. Clock Gating Disabled. (i.e., clocks are toggin	iy, aiways)				
	28	ICunit Clock Gating Disable					
		Access:	R/W				
		ICunit Clock Gating Disable Control:	<b>-</b>				
		'0': Clock Gating Enabled. (i.e., clocks can be gate	ed when they are not required to toggle				
		for functionality)					
		'1': Clock Gating Disabled. (i.e., clocks are togglir	ng, always)				
	27	HIZunit Clock Gating Disable					
		Access:	R/W				
		HIZunit Clock Gating Disable Control:					
		'0': Clock Gating Enabled. (i.e., clocks can be gate	ed when they are not required to toggle				
		for functionality)					
		'1': Clock Gating Disabled. (i.e., clocks are togglir	iy, aiways)				



26	GWunit Clock Gating Disable				
20	Access:	R/W			
	GWunit Clock Gating Disable Co				
		ocks can be gated when they are not required to t			
	for functionality)				
	'1': Clock Gating Disabled. (i.e., c	locks are toggling, always)			
25	GTIunit Clock Gating Disable				
	Default Value:	1b			
	Access:	R/W			
	GTI Units Clock Gating Disable C	Control:			
	3	ocks can be gated when they are not required to t			
	for functionality)				
	'1': Clock Gating Disabled. (i.e., c	locks are toggling, always)			
24	GSunit Clock Gating Disable				
	Access:	R/W			
	GSunit Clock Gating Disable Control:				
	'0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to toggle				
	for functionality)				
	'1': Clock Gating Disabled. (i.e., c	locks are toggling, always)			
	GPMunit Clock Gating Disable				
23	Default Value:	1b			
23	Default value.				
23	Access:	R/W			
23					
23	Access: GPMunit Clock Gating Disable C '0': Clock Gating Enabled. (i.e., cl	Control:			
23	Access: GPMunit Clock Gating Disable C '0': Clock Gating Enabled. (i.e., cl for functionality)	control: ocks can be gated when they are not required to			
23	Access: GPMunit Clock Gating Disable C '0': Clock Gating Enabled. (i.e., cl	control: ocks can be gated when they are not required to t			
	Access: GPMunit Clock Gating Disable C '0': Clock Gating Enabled. (i.e., cl for functionality)	control: ocks can be gated when they are not required to t			
	Access: GPMunit Clock Gating Disable C '0': Clock Gating Enabled. (i.e., cl for functionality) '1': Clock Gating Disabled. (i.e., c	control: ocks can be gated when they are not required to			
23	Access: GPMunit Clock Gating Disable C '0': Clock Gating Enabled. (i.e., cl for functionality) '1': Clock Gating Disabled. (i.e., c GAMunit Clock Gating Disable Default Value: Access:	Control: ocks can be gated when they are not required to clocks are toggling, always) 0b R/W			
	Access: GPMunit Clock Gating Disable C '0': Clock Gating Enabled. (i.e., cl for functionality) '1': Clock Gating Disabled. (i.e., c GAMunit Clock Gating Disable Default Value: Access: GAMunit Clock Gating Disable C	Control: ocks can be gated when they are not required to t clocks are toggling, always) 0b R/W Control:			
	Access: GPMunit Clock Gating Disable C '0': Clock Gating Enabled. (i.e., cl for functionality) '1': Clock Gating Disabled. (i.e., c GAMunit Clock Gating Disable Default Value: Access: GAMunit Clock Gating Disable C	Control: ocks can be gated when they are not required to t clocks are toggling, always) 0b R/W			



		-			
	Default Value:	Ob			
	Access:	R/W			
	GACunit Clock Gating Disable Control:	· ·			
	'0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to toggle				
	for functionality)				
	'1': Clock Gating Disabled. (i.e., clocks are	toggling, always)			
20	GABunit Clock Gating Disable				
	Default Value:	0b			
	Access:	R/W			
	GABunit Clock Gating Disable Control:				
	'0': Clock Gating Enabled. (i.e., clocks can	be gated when they are not required to toggle			
	for functionality)				
	'1': Clock Gating Disabled. (i.e., clocks are toggling, always)				
19	FTunit Clock Gating Disable				
	Access:	R/W			
	FTunit Clock Gating Disable Control:				
	'0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to toggle				
	for functionality)				
	'1': Clock Gating Disabled. (i.e., clocks are toggling, always)				
18	FLunit Clock Gating Disable				
	Access:	R/W			
	FLunit Clock Gating Disable Control:				
	'0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to toggle				
	for functionality)				
	'1': Clock Gating Disabled. (i.e., clocks are toggling, always)				
17	EU_FPUunit Clock Gating Disable				
	Access:	R/W			
	EU_FPUunit Clock Gating Disable Control:				
	'0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to toggle				
	for functionality)				
	'1': Clock Gating Disabled. (i.e., clocks are toggling, always)				
16	EU_TCunit Clock Gating Disable				
	Access:	R/W			
	EU_TCunit Clock Gating Disable Control:				
	'0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to toggle				



	for functionality) '1': Clock Gating Disabled. (i.e., clocks are to	ading always)	
	1. Clock Galling Disabled. (i.e., clocks are to	gginig, always)	
15	EU_EMunit Clock Gating Disable		
	Access:	R/W	
	EU_EMunit Clock Gating Disable Control:		
	'0': Clock Gating Enabled. (i.e., clocks can be	gated when they are not required to to	
	for functionality)		
	'1': Clock Gating Disabled. (i.e., clocks are to	ggling, always)	
14	EU_GAunit Clock Gating Disable		
	Access:	R/W	
	EU_GAunit Clock Gating Disable Control:	I	
	'0': Clock Gating Enabled. (i.e., clocks can be	gated when they are not required to to	
	for functionality)		
	'1': Clock Gating Disabled. (i.e., clocks are toggling, always)		
13	EUunit Clock Gating Disable		
13	Access:	R/W	
	EUunit Clock Gating Disable Control:	.,	
	'0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to toggle		
	for functionality)		
	'1': Clock Gating Disabled. (i.e., clocks are toggling, always)		
12	2 SVLunit Clock Gating Disable		
	Access:	R/W	
	SVLunit Clock Gating Disable Control:		
	'0': Clock Gating Enabled. (i.e., clocks can be	gated when they are not required to to	
	5	gated when they are not required to to	
	'0': Clock Gating Enabled. (i.e., clocks can be		
11	'0': Clock Gating Enabled. (i.e., clocks can be for functionality) '1': Clock Gating Disabled. (i.e., clocks are to		
11	'0': Clock Gating Enabled. (i.e., clocks can be for functionality) '1': Clock Gating Disabled. (i.e., clocks are to		
11	<ul> <li>'0': Clock Gating Enabled. (i.e., clocks can be for functionality)</li> <li>'1': Clock Gating Disabled. (i.e., clocks are to <b>DTunit Clock Gating Disable</b></li> <li>Access:</li> </ul>	ggling, always)	
11	'0': Clock Gating Enabled. (i.e., clocks can be for functionality) '1': Clock Gating Disabled. (i.e., clocks are to <b>DTunit Clock Gating Disable</b>	ggling, always)	
11	<ul> <li>'0': Clock Gating Enabled. (i.e., clocks can be for functionality)</li> <li>'1': Clock Gating Disabled. (i.e., clocks are to</li> <li><b>DTunit Clock Gating Disable</b></li> <li>Access:</li> <li>DTunit Clock Gating Disable Control:</li> </ul>	ggling, always)	
11	<ul> <li>'0': Clock Gating Enabled. (i.e., clocks can be for functionality)</li> <li>'1': Clock Gating Disabled. (i.e., clocks are to</li> <li><b>DTunit Clock Gating Disable</b></li> <li>Access:</li> <li>DTunit Clock Gating Disable Control:</li> <li>'0': Clock Gating Enabled. (i.e., clocks can be</li> </ul>	ggling, always) R/W gated when they are not required to to	
11	<ul> <li>'0': Clock Gating Enabled. (i.e., clocks can be for functionality)</li> <li>'1': Clock Gating Disabled. (i.e., clocks are to</li> </ul> <b>DTunit Clock Gating Disable</b> Access:   DTunit Clock Gating Disable Control:   '0': Clock Gating Enabled. (i.e., clocks can be for functionality)   '1': Clock Gating Disabled. (i.e., clocks can be for functionality)	ggling, always) R/W gated when they are not required to to	



	DMunit Clock Gating Disable Control: '0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to toggle for functionality) '1': Clock Gating Disabled. (i.e., clocks are toggling, always)				
9	DGunit Clock Gating Disable				
9	Access: R/W				
	DGunit Clock Gating Disable Control: '0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to toggle for functionality) '1': Clock Gating Disabled. (i.e., clocks are toggling, always)				
8	DAPunit Clock Gating Disable				
	Access: R/W				
	DAPunit Clock Gating Disable Control:				
	DAPunit Clock Gating Disable Control: '0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to toggle for functionality) '1': Clock Gating Disabled. (i.e., clocks are toggling, always)				
7	CSunit Clock Gating Disable				
	Access: R/W				
	CSunit Clock Gating Disable Control: '0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to toggle for functionality) '1': Clock Gating Disabled. (i.e., clocks are toggling, always)				
G	CLunit Clock Gating Disable				
6	Clunit Clock Gating Disable				
0	Access: R/W				
0					
0	Access: R/W				
5	Access:       R/W         CLunit Clock Gating Disable Control:       '0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to toggle for functionality)				
	Access:       R/W         CLunit Clock Gating Disable Control:       '0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to toggle for functionality)         '1': Clock Gating Disabled. (i.e., clocks are toggling, always)				
	Access:       R/W         CLunit Clock Gating Disable Control:       '0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to toggle for functionality)         '1': Clock Gating Disabled. (i.e., clocks are toggling, always)         BLBunit Clock Gating Disable         Access:       R/W         BLBunit Clock Gating Disable Control:         '0': Clock Gating Disable Control:         '0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to toggle for functionality)				
	Access:       R/W         CLunit Clock Gating Disable Control:       '0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to toggle for functionality)         '1': Clock Gating Disabled. (i.e., clocks are toggling, always)         BLBunit Clock Gating Disable         Access:       R/W         BLBunit Clock Gating Disable Control:         '0': Clock Gating Disable Control:         '0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to toggle				



	Access:	R/W	
	BFunit Clock Gating Disable Control:	k	
	'0': Clock Gating Enabled. (i.e., clocks ca	an be gated when they are not required to	togg
	for functionality)		
	'1': Clock Gating Disabled. (i.e., clocks a	re toggling, always)	
3	BDunit Clock Gating Disable		
	Access:	R/W	
	BDunit Clock Gating Disable Control:		
	'0': Clock Gating Enabled. (i.e., clocks ca	an be gated when they are not required to	tog
	for functionality)		
	'1': Clock Gating Disabled. (i.e., clocks are toggling, always)		
2	BCSunit Clock Gating Disable		
	Access:	R/W	
	BCSunit Clock Gating Disable Control: '0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to toggle for functionality) '1': Clock Gating Disabled. (i.e., clocks are toggling, always)		
1	AVSunit Clock Gating Disable		
	Access:	R/W	
	AVSunit Clock Gating Disable Control:		
	'0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to toggl		
	for functionality)		
	'1': Clock Gating Disabled. (i.e., clocks are toggling, always)		
0	Reserved		
0	Reserved Access:	RO	

Look Inside."

Register	Spa	Ice: MMIO: 0/2/0					
Default V Size (in b	Value	e: 0x0000000					
	oits):	32					
Address	:	09404h					
Unit Lev	el Clock Gating Control Registers.						
DWord	Bit	Description					
0	31	VFunit Clock Gating Disable					
		Access: R/W					
		VFunit Clock Gating Disable Control: '0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to toggle for functionality)					
		'1': Clock Gating Disabled. (i.e., clocks are toggling, always)					
	30	VDSunit Clock Gating Disable					
		Access: R/W					
		VDSunit Clock Gating Disable Control: '0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to toggle					
		for functionality) '1': Clock Gating Disabled. (i.e., clocks are toggling, always)					
	29 VDIunit Clock Gating Disable						
		Access: R/W					
		VDIunit Clock Gating Disable Control: '0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to toggle for functionality) '1': Clock Gating Disabled. (i.e., clocks are toggling, always)					
	28	VCSunit Clock Gating Disable					
	20	Access: R/W					
		Access: R/W VCSunit Clock Gating Disable Control: '0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to toggle for functionality) '1': Clock Gating Disabled. (i.e., clocks are toggling, always)					
	1. Clock Guting Disubled. (i.e., clocks are togging, aways)						
	27	DTOunit Clock Gating Disable					
		Access: R/W					
		DTOunit Clock Gating Disable Control: '0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to toggle for functionality)					
		'1': Clock Gating Disabled. (i.e., clocks are toggling, always)					



#### UCGCTL2 - Unit Level Clock Gating Control 2 26 VCPunit Clock Gating Disable R/W Access: VCPunit Clock Gating Disable Control: '0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to toggle for functionality) '1': Clock Gating Disabled. (i.e., clocks are toggling, always) 25 VCDunit Clock Gating Disable Access: R/W VCDunit Clock Gating Disable Control: '0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to toggle for functionality) '1': Clock Gating Disabled. (i.e., clocks are toggling, always) 24 URBMunit Clock Gating Disable R/W Access: URBMunit Clock Gating Disable Control: '0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to toggle for functionality) '1': Clock Gating Disabled. (i.e., clocks are toggling, always) 23 **TSGunit Clock Gating Disable** R/W Access: TSGunit Clock Gating Disable Control: '0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to toggle for functionality) '1': Clock Gating Disabled. (i.e., clocks are toggling, always) 22 **TDLunit Clock Gating Disable** Access: R/W TDLunit Clock Gating Disable Control: '0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to toggle for functionality) '1': Clock Gating Disabled. (i.e., clocks are toggling, always) 21 **TDSunit Clock Gating Disable** R/W Access: TDSunit Clock Gating Disable Control: '0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to toggle



for functionality) '1': Clock Gating Disabled. (i.e., clocks are toggling, always)

20	SVSMunit Clock Gating Disable				
	Access:	R/W			
	SVSMunit Clock Gating Disable Control:				
	'0': Clock Gating Enabled. (i.e., clocks can be ga	ted when they are not required to	o toggle		
	for functionality)				
	'1': Clock Gating Disabled. (i.e., clocks are togg	ling, always)			
19	SVGunit Clock Gating Disable				
	Access:	R/W			
	SVGunit Clock Gating Disable Control:				
	'0': Clock Gating Enabled. (i.e., clocks can be ga	ted when they are not required to	o toggle		
	for functionality)				
	'1': Clock Gating Disabled. (i.e., clocks are togg	ling, always)			
18	SOunit Clock Gating Disable				
	Access:	R/W			
	SOunit Clock Gating Disable Control:				
	'0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to toggle				
	for functionality)				
	'1': Clock Gating Disabled. (i.e., clocks are toggling, always)				
17	SIunit Clock Gating Disable				
	Access:	R/W			
	SIunit Clock Gating Disable Control:				
	'0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to toggle				
	for functionality)				
	'1': Clock Gating Disabled. (i.e., clocks are toggling, always)				
16	SFunit Clock Gating Disable				
	Access:	R/W			
	SFunit Clock Gating Disable Control:				
	'0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to toggle				
	for functionality)				
	'1': Clock Gating Disabled. (i.e., clocks are togg	ling, always)			
	SECunit Clock Gating Disable				
15	SECunit Clock Gating Disable				



	SECunit Clock Gating Disable Control:		
	'0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to toggle for functionality)		
	'1': Clock Gating Disabled. (i.e., clocks are toggling, always)		
14	4 SCunit Clock Gating Disable		
	Access: R/W		
	SCunit Clock Gating Disable Control:		
	'0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to toggle		
	for functionality)		
	'1': Clock Gating Disabled. (i.e., clocks are toggling, always)		
13	RCZunit Clock Gating Disable		
	Access: R/W		
	RCZunit Clock Gating Disable Control:		
	'0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to toggle		
	for functionality)		
	'1': Clock Gating Disabled. (i.e., clocks are toggling, always)		
12	2 RCPBunit Clock Gating Disable		
	Access: R/W		
	RCPBunit Clock Gating Disable Control:		
	'0': Clock Gating Enabled. (i.e., clocks can be gated when they are not	required to toge	
	for functionality)		
	'1': Clock Gating Disabled. (i.e., clocks are toggling, always)		
11	RCCunit Clock Gating Disable		
	Access: R/W		
	RCCunit Clock Gating Disable Control:		
	'0': Clock Gating Enabled. (i.e., clocks can be gated when they are not	required to toge	
	for functionality) '1': Clock Gating Disabled. (i.e., clocks are toggling, always)		
	1. Clock Gating Disabled. (i.e., clocks are togging, always)		
10	QCunit Clock Gating Disable		
	Access: R/W		
	QCunit Clock Gating Disable Control:		
	'0': Clock Gating Enabled. (i.e., clocks can be gated when they are not	required to toge	
	for functionality)		
	'1': Clock Gating Disabled. (i.e., clocks are toggling, always)		



	Access:	R/W		
	PSDunit Clock Gating Disable Control:			
	'0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to toggle			
	for functionality)			
	'1': Clock Gating Disabled. (i.e., clocks are	toggling, always)		
8	PLunit Clock Gating Disable			
	Access:	R/W		
	PLunit Clock Gating Disable Control:			
	'0': Clock Gating Enabled. (i.e., clocks can	be gated when they are not required to toggle		
	for functionality)			
	'1': Clock Gating Disabled. (i.e., clocks are	toggling, always)		
7	MTunit Clock Gating Disable			
	Access:	R/W		
	MTunit Clock Gating Disable Control:			
	'0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to toggle			
	for functionality)			
	'1': Clock Gating Disabled. (i.e., clocks are toggling, always)			
6	MPCunit Clock Gating Disable			
	A			
	Access:	R/W		
	MPCunit Clock Gating Disable Control:	R/W		
	MPCunit Clock Gating Disable Control:			
	MPCunit Clock Gating Disable Control:	R/W be gated when they are not required to toggle		
	MPCunit Clock Gating Disable Control: '0': Clock Gating Enabled. (i.e., clocks can	be gated when they are not required to toggle		
5	MPCunit Clock Gating Disable Control: '0': Clock Gating Enabled. (i.e., clocks can for functionality)	be gated when they are not required to toggle		
5	MPCunit Clock Gating Disable Control: '0': Clock Gating Enabled. (i.e., clocks can for functionality) '1': Clock Gating Disabled. (i.e., clocks are	be gated when they are not required to toggle		
5	MPCunit Clock Gating Disable Control: '0': Clock Gating Enabled. (i.e., clocks can for functionality) '1': Clock Gating Disabled. (i.e., clocks are <b>TDGunitClock Gating Disable</b>	be gated when they are not required to toggle toggling, always)		
5	MPCunit Clock Gating Disable Control: '0': Clock Gating Enabled. (i.e., clocks can for functionality) '1': Clock Gating Disabled. (i.e., clocks are <b>TDGunitClock Gating Disable</b> Access: TDGunit Clock Gating Disable Control:	be gated when they are not required to toggle toggling, always) R/W		
5	MPCunit Clock Gating Disable Control: '0': Clock Gating Enabled. (i.e., clocks can for functionality) '1': Clock Gating Disabled. (i.e., clocks are <b>TDGunitClock Gating Disable</b> Access: TDGunit Clock Gating Disable Control:	be gated when they are not required to toggle toggling, always) R/W		
5	MPCunit Clock Gating Disable Control: '0': Clock Gating Enabled. (i.e., clocks can for functionality) '1': Clock Gating Disabled. (i.e., clocks are <b>TDGunitClock Gating Disable</b> Access: TDGunit Clock Gating Disable Control: '0': Clock Gating Enabled. (i.e., clocks can	be gated when they are not required to toggle toggling, always) R/W be gated when they are not required to toggle		
5	MPCunit Clock Gating Disable Control: '0': Clock Gating Enabled. (i.e., clocks can for functionality) '1': Clock Gating Disabled. (i.e., clocks are <b>TDGunitClock Gating Disable</b> Access: TDGunit Clock Gating Disable Control: '0': Clock Gating Enabled. (i.e., clocks can for functionality) '1': Clock Gating Disabled. (i.e., clocks are	be gated when they are not required to toggle toggling, always) R/W be gated when they are not required to toggle		
	MPCunit Clock Gating Disable Control: '0': Clock Gating Enabled. (i.e., clocks can for functionality) '1': Clock Gating Disabled. (i.e., clocks are <b>TDGunitClock Gating Disable</b> Access: TDGunit Clock Gating Disable Control: '0': Clock Gating Enabled. (i.e., clocks can for functionality)	be gated when they are not required to toggle toggling, always) R/W be gated when they are not required to toggle		
	MPCunit Clock Gating Disable Control: '0': Clock Gating Enabled. (i.e., clocks can for functionality) '1': Clock Gating Disabled. (i.e., clocks are <b>TDGunitClock Gating Disable</b> Access: TDGunit Clock Gating Disable Control: '0': Clock Gating Enabled. (i.e., clocks can for functionality) '1': Clock Gating Disabled. (i.e., clocks are <b>MSCunit Clock Gating Disable</b> Access:	be gated when they are not required to toggle toggling, always) R/W be gated when they are not required to toggle toggling, always)		
	MPCunit Clock Gating Disable Control: '0': Clock Gating Enabled. (i.e., clocks can for functionality) '1': Clock Gating Disabled. (i.e., clocks are <b>TDGunitClock Gating Disable</b> Access: TDGunit Clock Gating Disable Control: '0': Clock Gating Enabled. (i.e., clocks can for functionality) '1': Clock Gating Disabled. (i.e., clocks are <b>MSCunit Clock Gating Disable</b> Access: MSCunit Clock Gating Disable Control:	be gated when they are not required to toggle toggling, always) R/W be gated when they are not required to toggle toggling, always) R/W		
	MPCunit Clock Gating Disable Control: '0': Clock Gating Enabled. (i.e., clocks can for functionality) '1': Clock Gating Disabled. (i.e., clocks are <b>TDGunitClock Gating Disable</b> Access: TDGunit Clock Gating Disable Control: '0': Clock Gating Enabled. (i.e., clocks can for functionality) '1': Clock Gating Disabled. (i.e., clocks are <b>MSCunit Clock Gating Disable</b> Access: MSCunit Clock Gating Disable Control:	be gated when they are not required to toggle toggling, always) R/W be gated when they are not required to toggle toggling, always)		



3	TEunit Clock Gating Disable	R/W	
	TEunit Clock Gating Disable Contro		
	3		
2	'0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to toggle for functionality)		
	'1': Clock Gating Disabled. (i.e., cloc	ks are toggling, always)	
	TETGunit Clock Gating Disable		
	Access:	R/W	
	TETGunit Clock Gating Disable Con	trol:	
	'0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to toggle		
	for functionality)		
	'1': Clock Gating Disabled. (i.e., clocks are toggling, always)		
1	MAunit Clock Gating Disable		
	Access:	R/W	
	MAunit Clock Gating Disable Contr	ol:	
	'0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to toggle		
	'0': Clock Gating Enabled. (i.e., clock	cs can be gated when they are not required to togg	
	for functionality)		
	5		
0	for functionality) '1': Clock Gating Disabled. (i.e., cloc		
0	for functionality) '1': Clock Gating Disabled. (i.e., cloc		
0	for functionality) '1': Clock Gating Disabled. (i.e., cloc IZunit Clock Gating Disable	ks are toggling, always)	
0	for functionality) '1': Clock Gating Disabled. (i.e., clock <b>IZunit Clock Gating Disable</b> Access: IZunit Clock Gating Disable Control '0': Clock Gating Enabled. (i.e., clock	ks are toggling, always)          R/W	
0	for functionality) '1': Clock Gating Disabled. (i.e., cloc <b>IZunit Clock Gating Disable</b> Access: IZunit Clock Gating Disable Control	ks are toggling, always)          R/W         R/W     <	



Jelault va	alue:	0x0000000				
Size (in bi	ts):	32				
Address:		09408h				
Jnit Level	evel Clock Gating Control Registers.					
DWord E		<u> </u>	Description			
0 3	31 FI	lunits 2nd Clock Gating Disable				
	A	Access:	R/W			
	'0 fc	Flunits 2nd Clock Gating Disable Control: '0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to toggle for functionality) '1': Clock Gating Disabled. (i.e., clocks are toggling, always)				
:	30 <b>S</b> '	VRRunit Clock Gating Disable				
		Access:	R/W			
		VRRunits' Clock Gating Disable Contro				
	fc	'0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to toggle for functionality) '1': Clock Gating Disabled. (i.e., clocks are toggling, always)				
	29 🔽	CRunit Clock Gating Disable				
	A	Access:	R/W			
	'0 fc	VCRunits' Clock Gating Disable Control: '0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to toggle for functionality) '1': Clock Gating Disabled. (i.e., clocks are toggling, always)				
-	28 EI	DTunit Clock Gating Disable				
	A	Access:	R/W			
	'0	DTunits' Clock Gating Disable Control )': Clock Gating Enabled. (i.e., clocks ca or functionality)	: an be gated when they are not required to toggle			
	'1	': Clock Gating Disabled. (i.e., clocks a	re toggling, always)			
		Clunit Clock Gating Disable				
		Access:	R/W			
	VCIunits' Clock Gating Disable Control: '0': Clock Gating Enabled. (i.e., clocks can be gated when for functionality) '1': Clock Gating Disabled. (i.e., clocks are toggling, alway					



26	Reserved		
	Access:	RO	
	Reserved.		
25	HSunit Clock Gating Disable		
	Access:	R/W	
	HSunit Clock Gating Disable Control: '0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to toggl for functionality) '1': Clock Gating Disabled. (i.e., clocks are toggling, always)		
24	SOLunit Clock Gating Disable		
	Access:	R/W	
	SOLunit Clock Gating Disable Control: '0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to toggle for functionality) '1': Clock Gating Disabled. (i.e., clocks are toggling, always)		
23	QRCunit Clock Gating Disable		
	Access:	R/W	
	QRCunit Clock Gating Disable Control: '0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to toggle for functionality) '1': Clock Gating Disabled. (i.e., clocks are toggling, always)		
22	MSPBISTunit Clock Gating Disable		
	Access:	R/W	
	MSPBISTunit Clock Gating Disable Control: '0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to toggl for functionality) '1': Clock Gating Disabled. (i.e., clocks are toggling, always)		
	for functionality)		
21	for functionality)		
21	for functionality) '1': Clock Gating Disabled. (i.e., clo		



20	Reserved				
19	SBEunit Clock Gating Disable				
	Access: R/W				
	SBEunit Clock Gating Disable Control:				
	'0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to tog	ggle			
	for functionality)				
	'1': Clock Gating Disabled. (i.e., clocks are toggling, always)				
18	BCunit Clock Gating Disable				
	Access: R/W				
	BCunit Clock Gating Disable Control:				
	'0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to tog	ggle			
	for functionality)				
	'1': Clock Gating Disabled. (i.e., clocks are toggling, always)	'1': Clock Gating Disabled. (i.e., clocks are toggling, always)			
17	WMBE Clock Gating Disable				
	Access: R/W				
	WMBEunit Clock Gating Disable Control:				
	'0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to toggle				
	for functionality)				
	'1': Clock Gating Disabled. (i.e., clocks are toggling, always)				
16	WMFEunit Clock Gating Disable				
	Access: R/W				
	WMFEunit Clock Gating Disable Control:				
	'0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to toggle				
	for functionality)				
	'1': Clock Gating Disabled. (i.e., clocks are toggling, always)				
15	VSCunit Clock Gating Disable				
	Access: R/W				
	VSCunit Clock Gating Disable Control:				
	'0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to toggle				
	for functionality)				
	'1': Clock Gating Disabled. (i.e., clocks are toggling, always)				
	4 Reserved				
14	Reserved	<u> </u>			
14 13					



	'0': Clock Gating Enabled. (i.e., clocks can for functionality)	be gated when they are not required to tog		
	'1': Clock Gating Disabled. (i.e., clocks are	toggling, always)		
12	STCunit Clock Gating Disable			
	Access:	R/W		
	STCunit Clock Gating Disable Control: '0': Clock Gating Enabled. (i.e., clocks can for functionality) '1': Clock Gating Disabled. (i.e., clocks are	be gated when they are not required to tog toggling, always)		
11	VSunit Clock Gating Disable			
	Access:	R/W		
	VSunit Clock Gating Disable Control:	·		
	'0': Clock Gating Enabled. (i.e., clocks can for functionality) '1': Clock Gating Disabled. (i.e., clocks are	be gated when they are not required to tog		
		togginig, always)		
10	VOPunit Clock Gating Disable			
	Access:	R/W		
	VOPunit Clock Gating Disable Control: '0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to toggle for functionality) '1': Clock Gating Disabled. (i.e., clocks are toggling, always)			
9	VMXunit Clock Gating Disable			
	Access:	R/W		
	VMXunit Clock Gating Disable Control:	1		
	'0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to togg for functionality) '1': Clock Gating Disabled. (i.e., clocks are toggling, always)			
	1. Clock Gating Disabled. (i.e., clocks are	togginig, always)		
8	VMEunit Clock Gating Disable			
	Access:	R/W		
	VMEunit Clock Gating Disable Control: '0': Clock Gating Enabled, (i.e., clocks can	be gated when they are not required to tog		
	for functionality)	5 7 . 5		



	Access:	R/W				
	VMDunit Clock Gating Disable Control:					
	'0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to toggle					
	for functionality)					
	'1': Clock Gating Disabled. (i.e., clocks are toggling, alv	vays)				
6	5 VMCunit Clock Gating Disable					
	Access:	R/W				
	VMCunit Clock Gating Disable Control:	•				
	'0': Clock Gating Enabled. (i.e., clocks can be gated wh	en they are not required to toggle				
	for functionality)					
	'1': Clock Gating Disabled. (i.e., clocks are toggling, alv	vays)				
5	VLFunit Clock Gating Disable					
	Access:	R/W				
	VLFunit Clock Gating Disable Control:					
	5	'0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to toggle				
	for functionality)					
	'1': Clock Gating Disabled. (i.e., clocks are toggling, alv	vays)				
4	VITunit Clock Gating Disable					
	Access:	R/W				
	VITunit Clock Gating Disable Control:					
	'0': Clock Gating Enabled. (i.e., clocks can be gated wh	en they are not required to toggle				
	for functionality)					
	'1': Clock Gating Disabled. (i.e., clocks are toggling, alv	vays)				
3	VIPunit Clock Gating Disable					
	Access:	R/W				
	VIPunit Clock Gating Disable Control:					
	'0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to toggle					
	for functionality)					
	'1': Clock Gating Disabled. (i.e., clocks are toggling, always)					
	VINunit Clock Gating Disable					
2		DAM				
2	Access:	R/W				
2	Access: VINunit Clock Gating Disable Control:	K/ W				
2						
2	VINunit Clock Gating Disable Control:					



1	VFTunit Clock Gating Disable				
	Access:	R/W			
	VFTunit Clock Gating Disable Contr	ol:			
	'0': Clock Gating Enabled. (i.e., clock	ks can be gated when they are not required to to	oggle		
	for functionality)				
	'1': Clock Gating Disabled. (i.e., clocks are toggling, always)				
0	VFEunit Clock Gating Disable				
	Access:	R/W			
	VFEunit Clock Gating Disable Control:				
	'0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to toggle				
	for functionality)				
	'1': Clock Gating Disabled. (i.e., clocks are toggling, always)				



Register	Space:	:: MMIO: 0/2/0				
Default \	/alue:	0x00F80003				
Size (in b	oits):	32				
Address:		0940Ch				
Unit Leve	el Cloc	k Gating Control Registers.				
DWord	Bit	Description				
0	31:30	Reserved				
		Access: RO				
		Reserved.				
	29	GAFSRRB unit Clock Gate Disable				
		Access: R/W				
		GAFSRRB units Clock Gating Disable Control:				
		'0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to toggle				
		for functionality)				
		'1': Clock Gating Disabled. (i.e., clocks are toggling, always)				
	28	RAMDFT units Clock Gate Disable				
		Access: R/W				
		RAMDFT units Clock Gating Disable Control:				
		'0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to toggle				
		for functionality)				
		'1': Clock Gating Disabled. (i.e., clocks are toggling, always)				
	27	Reserved				
		Access: RO				
		Reserved.				
	26	L3 CBR 1x Clock Gate Disable				
		Access: R/W				
		L3 CBR units 1x Clock Gating Disable Control:				
		'0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to toggle				
		for functionality)				
		'1': Clock Gating Disabled. (i.e., clocks are toggling, always)				
	25	L3 BANK 2x Clock Gate Disable				
		Access: R/W				
		L3 BANK units 2x Clock Gating Disable Control:				
	'0': Clock Gating Enabled. (i.e., clocks can be gated when they are not require					



	for functionality) '1': Clock Gating Disabled. (i.e., clocks are tog	gling, always)			
24	L3 BANK 1x Clock Gate Diable				
	Access:	R/W			
	L3 BANK units 1x Clock Gating Disable Contro '0': Clock Gating Enabled. (i.e., clocks can be g for functionality) '1': Clock Gating Disabled. (i.e., clocks are tog	ated when they are not required to t			
23	MBGFunit Clock Gate Disable				
	Default Value:	1b			
	Access:	R/W			
	MBGFunits Clock Gating Disable Control:				
22	for functionality) '1': Clock Gating Disabled. (i.e., clocks are toggling, always)				
22	MSQDunit 2x Clock Gate Disable Default Value:	1b			
	Access:	R/W			
	MSQD units cu2x Clock Gating Disable Control: '0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to togo for functionality) '1': Clock Gating Disabled. (i.e., clocks are toggling, always)				
21	MSQDunit Clock Gate Disable				
	Default Value:	1b			
	Access:	R/W			
	MSQD units 1x Clock Gating Disable Control: '0': Clock Gating Enabled. (i.e., clocks can be g for functionality) '1': Clock Gating Disabled. (i.e., clocks are tog				
20	MISDunits 2x Clock Gate Disable				
	Default Value:	1b			
	Access:	R/W			
	MISDunits cu2x Clock Gating Disable Control: '0': Clock Gating Enabled. (i.e., clocks can be g for functionality)				



	'1': Clock Gating Disabled. (i.e., clock	s are toggling, always)			
19	MISDunit Clock Gate Disable				
	Default Value:	1b			
	Access:	R/W			
	MISDunits 1x Clock Gating Disable C	Control:			
	5	s can be gated when they are not required to			
	for functionality)				
	'1': Clock Gating Disabled. (i.e., clock	s are toggling, always)			
.8	GAFMunit Clock Gate Disable				
	Access:	R/W			
	GAFMunit' Clock Gating Disable Cor	ntrol:			
	5	s can be gated when they are not required to			
	for functionality)				
	'1': Clock Gating Disabled. (i.e., clock	s are toggling, always)			
.7	GAPCunit Clock Gate Disable				
	Access:	D AA/			
		R/W			
	GAPCunits Clock Gating Disable Cor	ntrol:			
	'0': Clock Gating Enabled. (i.e., clocks	ntrol:			
	'0': Clock Gating Enabled. (i.e., clocks for functionality)	ntrol: s can be gated when they are not required to			
	'0': Clock Gating Enabled. (i.e., clocks	ntrol: s can be gated when they are not required to			
.6	'0': Clock Gating Enabled. (i.e., clocks for functionality)	ntrol: s can be gated when they are not required to			
.6	'0': Clock Gating Enabled. (i.e., clocks for functionality) '1': Clock Gating Disabled. (i.e., clock	ntrol: s can be gated when they are not required to			
.6	<ul> <li>'0': Clock Gating Enabled. (i.e., clocks for functionality)</li> <li>'1': Clock Gating Disabled. (i.e., clock</li> <li>GAPZunit Clock Gate Disable</li> <li>Access:</li> <li>GAPZunits' Clock Gating Disable Contemport</li> </ul>	ntrol: s can be gated when they are not required to ts are toggling, always) R/W ntrol:			
.6	<ul> <li>'0': Clock Gating Enabled. (i.e., clocks for functionality)</li> <li>'1': Clock Gating Disabled. (i.e., clock</li> </ul> <b>GAPZunit Clock Gate Disable</b> Access: GAPZunits' Clock Gating Disable Control of Clock Gating Enabled. (i.e., clocks)	ntrol: s can be gated when they are not required to ts are toggling, always) R/W ntrol:			
.6	<ul> <li>'0': Clock Gating Enabled. (i.e., clocks for functionality)</li> <li>'1': Clock Gating Disabled. (i.e., clock</li> <li>GAPZunit Clock Gate Disable</li> <li>Access:</li> <li>GAPZunits' Clock Gating Disable Control</li> <li>'0': Clock Gating Enabled. (i.e., clocks for functionality)</li> </ul>	ntrol: s can be gated when they are not required to s are toggling, always) R/W ntrol: s can be gated when they are not required to			
.6	<ul> <li>'0': Clock Gating Enabled. (i.e., clocks for functionality)</li> <li>'1': Clock Gating Disabled. (i.e., clock</li> <li>GAPZunit Clock Gate Disable</li> <li>Access:</li> <li>GAPZunits' Clock Gating Disable Control</li> <li>'0': Clock Gating Enabled. (i.e., clocks</li> </ul>	ntrol: s can be gated when they are not required to s are toggling, always) R/W ntrol: s can be gated when they are not required to			
	<ul> <li>'0': Clock Gating Enabled. (i.e., clocks for functionality)</li> <li>'1': Clock Gating Disabled. (i.e., clock</li> <li>GAPZunit Clock Gate Disable</li> <li>Access:</li> <li>GAPZunits' Clock Gating Disable Control</li> <li>'0': Clock Gating Enabled. (i.e., clocks for functionality)</li> </ul>	ntrol: s can be gated when they are not required to s are toggling, always) R/W ntrol: s can be gated when they are not required to			
	<ul> <li>'0': Clock Gating Enabled. (i.e., clocks for functionality)</li> <li>'1': Clock Gating Disabled. (i.e., clock</li> <li>GAPZunit Clock Gate Disable</li> <li>Access:</li> <li>GAPZunits' Clock Gating Disable Conductionality)</li> <li>'0': Clock Gating Enabled. (i.e., clocks for functionality)</li> <li>'1': Clock Gating Disabled. (i.e., clocks for functionality)</li> </ul>	ntrol: s can be gated when they are not required to s are toggling, always) R/W ntrol: s can be gated when they are not required to s are toggling, always) R/W			
	<ul> <li>'0': Clock Gating Enabled. (i.e., clocks for functionality)</li> <li>'1': Clock Gating Disabled. (i.e., clock</li> <li>GAPZunit Clock Gate Disable</li> <li>Access:</li> <li>GAPZunits' Clock Gating Disable Conditionality)</li> <li>'1': Clock Gating Enabled. (i.e., clocks for functionality)</li> <li>'1': Clock Gating Disabled. (i.e., clocks for functionality)</li> </ul>	ntrol: s can be gated when they are not required to ts are toggling, always) R/W ntrol: s can be gated when they are not required to ts are toggling, always) R/W ontrol:			
	<ul> <li>'0': Clock Gating Enabled. (i.e., clocks for functionality)</li> <li>'1': Clock Gating Disabled. (i.e., clock</li> <li>GAPZunit Clock Gate Disable</li> <li>Access:</li> <li>GAPZunits' Clock Gating Disable Cond</li> <li>'0': Clock Gating Enabled. (i.e., clocks for functionality)</li> <li>'1': Clock Gating Disabled. (i.e., clocks for functionality)</li> </ul>	ntrol: s can be gated when they are not required to ts are toggling, always) R/W ntrol: s can be gated when they are not required to ts are toggling, always) R/W ontrol:			
	<ul> <li>'0': Clock Gating Enabled. (i.e., clocks for functionality)</li> <li>'1': Clock Gating Disabled. (i.e., clock</li> <li>GAPZunit Clock Gate Disable</li> <li>Access:</li> <li>GAPZunits' Clock Gating Disable Conditionality)</li> <li>'1': Clock Gating Enabled. (i.e., clocks for functionality)</li> <li>'1': Clock Gating Disabled. (i.e., clocks for functionality)</li> <li>'1': Clock Gating Enabled. (i.e., clocks for functionality)</li> </ul>	Introl:         s can be gated when they are not required to         ts are toggling, always)         R/W         ntrol:         s can be gated when they are not required to         ts are toggling, always)         R/W         R/W         It toggling, always)         R/W         It toggling, always)         R/W         S can be gated when they are not required to         S can be gated when they are not required to			
	<ul> <li>'0': Clock Gating Enabled. (i.e., clocks for functionality)</li> <li>'1': Clock Gating Disabled. (i.e., clock</li> <li>GAPZunit Clock Gate Disable</li> <li>Access:</li> <li>GAPZunits' Clock Gating Disable Cond</li> <li>'0': Clock Gating Enabled. (i.e., clocks for functionality)</li> <li>'1': Clock Gating Disabled. (i.e., clocks for functionality)</li> </ul>	Introl:         s can be gated when they are not required to         ts are toggling, always)         R/W         ntrol:         s can be gated when they are not required to         ts are toggling, always)         R/W         R/W         It toggling, always)         R/W         It toggling, always)         R/W         S can be gated when they are not required to         S can be gated when they are not required to			
15	<ul> <li>'0': Clock Gating Enabled. (i.e., clocks for functionality)</li> <li>'1': Clock Gating Disabled. (i.e., clock</li> <li>GAPZunit Clock Gate Disable</li> <li>Access:</li> <li>GAPZunits' Clock Gating Disable Conditionality)</li> <li>'1': Clock Gating Enabled. (i.e., clocks for functionality)</li> <li>'1': Clock Gating Disabled. (i.e., clocks for functionality)</li> <li>'1': Clock Gating Enabled. (i.e., clocks for functionality)</li> </ul>	Introl:         s can be gated when they are not required to         ts are toggling, always)         R/W         ntrol:         s can be gated when they are not required to         ts are toggling, always)         R/W         R/W         It toggling, always)         R/W         It toggling, always)         R/W         S can be gated when they are not required to         S can be gated when they are not required to			



1					
	GAFSunits' Clock Gating Disable Control:				
	'0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to toggle				
	for functionality)				
	'1': Clock Gating Disabled. (i.e., clocks are toggling, always)				
13	GAHSunit Clock Gate Disable				
	Access: R/W				
	GAHSunits' Clock Gating Disable Control:				
	'0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to toggle				
	for functionality)				
	'1': Clock Gating Disabled. (i.e., clocks are toggling, always)				
12	VISunit Clock Gate Disable				
	Access: R/W				
	VISunits' Clock Gating Disable Control:				
	'0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to toggle				
	for functionality)				
	'1': Clock Gating Disabled. (i.e., clocks are toggling, always)				
11	VACunit Clock Gate Disable				
	Access: R/W				
	VACunits' Clock Gating Disable Control:				
	'0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to toggle				
	for functionality)				
	'1': Clock Gating Disabled. (i.e., clocks are toggling, always)				
10	VAMunit Clock Gate Disable				
	Access: R/W				
	VAMunits Clock Gating Disable Control:				
	'0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to toggle				
	for functionality)				
	'1': Clock Gating Disabled. (i.e., clocks are toggling, always)				
9	VADuit Clock Gating Disable				
	Access: R/W				
	VADunits Clock Gating Disable Control:				
	'0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to toggle				
	for functionality)				
	'1': Clock Gating Disabled. (i.e., clocks are toggling, always)				
8	JPGunit Clock Gating Disable				



	Access: R/W			
	JPGunits Clock Gating Disable Control:			
	'0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to			
	for functionality)			
	'1': Clock Gating Disabled. (i.e., clocks are toggling, always)			
7	VBPunits Clock Gating Disable			
	Access: R/W			
	VBPunits Clock Gating Disable Control:			
	'0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to			
	for functionality)			
	'1': Clock Gating Disabled. (i.e., clocks are toggling, always)			
6	VHRunit Clock Gating Disable			
	Access: R/W			
	VHRunits Clock Gating Disable Control:			
	'0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to			
	for functionality)			
	'1': Clock Gating Disabled. (i.e., clocks are toggling, always)			
5	'1': Clock Gating Disabled. (i.e., clocks are toggling, always)         VID4 VINunit Clock Gating Disable			
5				
5	VID4 VINunit Clock Gating Disable			
5	VID4 VINunit Clock Gating Disable       Access:   R/W			
5	VID4 VINunit Clock Gating Disable         Access:       R/W         VID4 VINunits' Clock Gating Disable Control:         '0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to for functionality)			
5	VID4 VINunit Clock Gating Disable         Access:       R/W         VID4 VINunits' Clock Gating Disable Control:         '0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to			
5	VID4 VINunit Clock Gating Disable         Access:       R/W         VID4 VINunits' Clock Gating Disable Control:         '0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to for functionality)			
	VID4 VINunit Clock Gating Disable         Access:       R/W         VID4 VINunits' Clock Gating Disable Control:       '0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to for functionality)         '1': Clock Gating Disabled. (i.e., clocks are toggling, always)			
	VID4 VINunit Clock Gating Disable         Access:       R/W         VID4 VINunits' Clock Gating Disable Control:         '0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to for functionality)         '1': Clock Gating Disabled. (i.e., clocks are toggling, always)         VID3 VINunit Clock Gating Disable			
	VID4 VINunit Clock Gating Disable         Access:       R/W         VID4 VINunits' Clock Gating Disable Control:         '0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to for functionality)         '1': Clock Gating Disabled. (i.e., clocks are toggling, always)         VID3 VINunit Clock Gating Disable         Access:       R/W			
	VID4 VINunit Clock Gating Disable         Access:       R/W         VID4 VINunits' Clock Gating Disable Control:         '0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to for functionality)         '1': Clock Gating Disabled. (i.e., clocks are toggling, always)         VID3 VINunit Clock Gating Disable         Access:       R/W         VID3 VINunits' Clock Gating Disable Control:			
	VID4 VINunit Clock Gating Disable         Access:       R/W         VID4 VINunits' Clock Gating Disable Control:       '0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to for functionality)         '1': Clock Gating Disabled. (i.e., clocks are toggling, always)         VID3 VINunit Clock Gating Disable         Access:       R/W         VID3 VINunits' Clock Gating Disable Control:         '0': Clock Gating Disable			
	VID4 VINunit Clock Gating Disable         Access:       R/W         VID4 VINunits' Clock Gating Disable Control:       '0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to for functionality)         '1': Clock Gating Disabled. (i.e., clocks are toggling, always)         VID3 VINunit Clock Gating Disable         Access:       R/W         VID3 VINunits' Clock Gating Disable Control:         '0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to for functionality)			
4	VID4 VINunit Clock Gating Disable         Access:       R/W         VID4 VINunits' Clock Gating Disable Control:       '0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to for functionality)         '1': Clock Gating Disabled. (i.e., clocks are toggling, always)         VID3 VINunit Clock Gating Disable         Access:       R/W         VID3 VINunits' Clock Gating Disable         Access:       R/W         VID3 VINunits' Clock Gating Disable Control:         '0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to for functionality)         '1': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to for functionality)         '1': Clock Gating Disabled. (i.e., clocks are toggling, always)			
4	VID4 VINunit Clock Gating Disable         Access:       R/W         VID4 VINunits' Clock Gating Disable Control:       '0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to for functionality)         '1': Clock Gating Disabled. (i.e., clocks are toggling, always)         VID3 VINunit Clock Gating Disable         Access:       R/W         VID3 VINunits' Clock Gating Disable         Access:       R/W         VID3 VINunits' Clock Gating Disable Control:         '0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to for functionality)         '1': Clock Gating Disabled. (i.e., clocks are toggling, always)         VID3 VINunits' Clock Gating Disable Control:         '0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to for functionality)         '1': Clock Gating Disabled. (i.e., clocks are toggling, always)         VID2 VINunit Clock Gating Disable			
4	VID4 VINunit Clock Gating Disable         Access:       R/W         VID4 VINunits' Clock Gating Disable Control:       '0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to for functionality)         '1': Clock Gating Disabled. (i.e., clocks are toggling, always)         VID3 VINunit Clock Gating Disable         Access:       R/W         VID3 VINunits' Clock Gating Disable Control:         '0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to for functionality)         '1': Clock Gating Disable Control:         '0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to for functionality)         '1': Clock Gating Disabled. (i.e., clocks are toggling, always)         VID2 VINunit Clock Gating Disable         Access:       R/W			
4	VID4 VINunit Clock Gating Disable         Access:       R/W         VID4 VINunits' Clock Gating Disable Control:       '0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to for functionality)         '1': Clock Gating Disabled. (i.e., clocks are toggling, always)         VID3 VINunit Clock Gating Disable         Access:       R/W         VID3 VINunits' Clock Gating Disable Control:         '0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to for functionality)         '1': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to for functionality)         '1': Clock Gating Disabled. (i.e., clocks can be gated when they are not required to for functionality)         '1': Clock Gating Disabled. (i.e., clocks are toggling, always)         VID2 VINunit Clock Gating Disable         Access:       R/W         VID2 VINunit Clock Gating Disable         Access:       R/W         VID2 VINunits' Clock Gating Disable			



2	VID1 VINunit Clock Gating Disable				
	Access:	R/W			
	VID1 VINunits' Clock Gating Disab	le Control:			
	'0': Clock Gating Enabled. (i.e., cloc	ks can be gated when they are	e not required to toggle		
	for functionality)				
	'1': Clock Gating Disabled. (i.e., clo	cks are toggling, always)			
1:0	MSQCunit Clock Gating Disable				
	Default Value:		11b		
	Access:		R/W		
	MSQCunits' Clock Gating Disable Control:				
	'0': Clock Gating Enabled. (i.e., clocks can be gated when they are not required to toggle				
	for functionality)				
	'1': Clock Gating Disabled. (i.e., cloo	cks are toggling, always)			



#### CASCTLB\_VLD\_0 - Valid Bit Vector 0 for CASC

Register Space: MMIO: 0/2/0							
Default Value:	0x00000000						
Size (in bits):	e (in bits): 32						
Address:	04760h-0476	53h					
This register contains the valid bits for entries 0-31 of CASCTLB							
DWord	Bit		Description				
0	31:0	Valid Bit Vector 0 for CAS	c				
		Default Value:	0000000h				
		Access:	RO				
		Valid bits per entry					



#### CVSTLB\_VLD\_0 - Valid Bit Vector 0 for CVS

Register Space: MMIO: 0/2/0								
Default Value: 0x0000000								
Size (in bits):	Size (in bits): 32							
Address:	04724h-04727	h						
This register contains the valid bits for entries 0-31 of CVSTLB								
DWord	Bit		Description					
0	31:0	Valid Bit Vector 0 for CV	s					
		Default Value:	0000000h					
		Access:	RO					
		Valid bits per entry						



# L3TLB\_VLD\_0 - Valid Bit Vector 0 for L3

Register Space:	MMIO: 0/2/0			
Default Value:	0x00000000			
Size (in bits):	32			
Address:	04740h-04743	3h		
This register contains the valid bits for entries 0-31 of L3TLB				
DWord	Bit		Description	
0	31:0	Valid Bit Vector 0 for L3		
		Default Value:	0000000h	
		Access:	RO	
		Valid bits per entry		



MTTLB_VLD0 - Valid Bit Vector 0 for MTTLB				
Register Space:	MMIC	): 0/2/0		
Source:	Rende	erCS		
Default Value:	0x0000000			
Access:	RO			
Size (in bits):	32			
Trusted Type:	1			
Address:	04780h-04783h			
This register contains the valid bits for entries 0-31 of MTTLB (Texture and constant cache TLB).				
DWord		Bit	Description	
0		31:0	Valid bits per entry	



# VICTLB\_VLD0 - Valid Bit Vector 0 for MTVICTLB

Register Space:	MMIO: 0/2/0			
Source:	Render	CS		
Default Value: 0	0x0000	0000		
Access:	RO			
Size (in bits):	32			
Trusted Type:	1			
Address: 0	04788h-0478Bh			
This register contains t Streamer TLB).	This register contains the valid bits for entries 0-31 of VICTLB (Vertex Fetch, Instruction Cache, and Command Streamer TLB).			
DWord	Bit Description			
0	31:0 Valid bits per entry			



# **RCCTLB\_VLD\_0** - Valid Bit Vector 0 for RCC

Register Space:	MMIO: 0/2/0			
Default Value:	0x0000000			
Size (in bits):	32			
Address:	04728h-0472E	ßh		
This register contains the valid bits for entries 0-31 of RCCTLB				
DWord	Bit		Description	
0	31:0	Valid Bit Vector 0 for RCC		
		Default Value:	0000000h	
		Access:	RO	
		Valid bits per entry		



Г

RCCTLB_VLD0 - Valid Bit Vector 0 for RCCTLB				
Register Space:	MMIC	): 0/2/0		
Source:	Rende	erCS		
Default Value:	0x000	00000		
Access:	RO			
Size (in bits):	32			
Trusted Type:	1			
Address:	04790h-04793h			
This register contains the valid bits for entries 0-31 of RCCTLB (Render Cache for Color TLB).				
DWord		Bit	Description	
0	31:0 Valid bits per entry			

Г



<b>RCZTLB_VLD0</b> - Valid Bit Vector 0 for RCZTLB				
Register Space:	MMIC	D: 0/2/0		
Source:	Rende	erCS		
Default Value:	0x000	0x0000000		
Access:	RO			
Size (in bits):	32			
Trusted Type:	1			
Address:	04798h-0479Bh			
This register contains	This register contains the valid bits for entries 0-31 of RCZTLB (Render Cache for Z (Depth), Hi Z, and Stencil TLB).			
DWord	Bit Description			
0	31:0 Valid bits per entry			



Г

MTTL	<b>B06</b> 4	\$_VLD0 - \	/alid Bit Vector 0 for TLB064	
Register Space:	MMIO:	0/2/0		
Source:	VideoC	5		
Default Value:	0x00000	0x0000000		
Access:	RO			
Size (in bits):	32			
Trusted Type:	1			
Address:	14780h-14783h			
This register contains the valid bits for entries 0-31 of MTTLB (Texture and constant cache TLB).				
DWord		Bit	Description	
0	31:0 Valid bits per entry			

Г



MTTLB132_VLD0 - Valid Bit Vector 0 for TLB132				
Register Space:	MMIO:	0/2/0		
Source:	VideoC	5		
Default Value:	0x0000000			
Access:	RO			
Size (in bits):	32			
Trusted Type:	1			
Address:	14788h-1478Bh			
This register contains the valid bits for entries 0-31 of MTTLB (Texture and constant cache TLB).				
DWord	Bit Description			
0		31:0	Valid bits per entry	



MTTLB232_VLD0 - Valid Bit Vector 0 for TLB232				
Register Space:	MMIO: 0/2/0			
Source:	VideoCS			
Default Value:	0x0000000			
Access:	RO			
Size (in bits):	32			
Trusted Type:	1			
Address:	14790h-14793h			

This register contains the valid bits for entries 0-31 of MTTLB (Texture and constant cache TLB).			
DWord Bit Description			
0	31:0	Valid bits per entry	

Г



MTTLB304_VLD0 - Valid Bit Vector 0 for TLB304				
Register Space:	MMIO:	0/2/0		
Source:	VideoC	5		
Default Value:	0x0000000			
Access:	RO			
Size (in bits):	32			
Trusted Type:	1			
Address:	14798h-1479Bh			
This register contains the valid bits for entries 0-31 of MTTLB (Texture and constant cache TLB).				
DWord	Bit Description			
0		31:0	Valid bits per entry	



# **TLBPEND\_VLD0 - Valid Bit Vector 0 for TLBPEND registers**

Register Space:	MMIO: 0/2/0			
Source:	Render	rCS		
Default Value:	0x0000	00000		
Access:	R/W	R/W		
Size (in bits):	32			
Trusted Type:	1			
Address:	04700h-04703h			
This register contains the valid bits for entries 0-31 of TLBPEND structure (Cycles pending TLB translation).				
DWord Bit		Bit	Description	
0		31:0	Valid bits per entry	



# VLFTLB\_VLD\_0 - Valid Bit Vector 0 for VLF

Register Space:	MMIO: 0/2/0	MMIO: 0/2/0				
Default Value:	0x0000000					
Size (in bits):	32					
Address:	04720h-0472	3h				
This register con	tains the valid b	its for entries 0-31 of VLFT	ĽB			
DWord	Bit		Description			
0	31:0	Valid Bit Vector 0 for VLF	· · · · · · · · · · · · · · · · · · ·			
0	31:0	Valid Bit Vector 0 for VLF Default Value:	· · · · · · · · · · · · · · · · · · ·			
0	31:0					
0	31:0	Default Value:	00000000h			
0	31:0	Default Value: Access:	00000000h			



# **ZTLB\_VLD\_0** - Valid Bit Vector 0 for Z

Register Space:	MMIO: 0/2/0					
Default Value:	0x00000000					
Size (in bits):	32					
Address:	04730h-04733	h				
This register contain	This register contains the valid bits for entries 0-31 of ZTLB					
DWord	Bit		Description			
0	31:0	Valid Bit Vector 0 for Z				
		Default Value:	0000000h			
		Access:	RO			
		Valid bits per entry				



# CASCTLB\_VLD\_1 - Valid Bit Vector 1 for CASC

Register Space:	MMIO: 0/2/0				
Default Value:	0x0000000				
Size (in bits):	32				
Address:	04764h-0476	57h			
This register contai	This register contains the valid bits for entries 0-31 of CASCTLB				
DWord	Bit		Description		
0	31:0	Valid Bit Vector 1 for CAS	c		
		Default Value:	0000000h		
		Access:	RO		
		Valid bits per entry			



# L3TLB\_VLD\_1 - Valid Bit Vector 1 for L3

Register Space:	MMIO: 0/2/0				
Default Value:	0x00000000				
Size (in bits):	32				
Address:	04744h-0474	7h			
This register contains the valid bits for entries 0-31 of L3TLB					
DWord	Bit		Description		
0	31:0	Valid Bit Vector 1 for L3			
		Default Value:	0000000h		
		Access:	RO		
		Valid bits per entry			



MTTLB_VLD1 - Valid Bit Vector 1 for MTTLB					
Register Space:	MMIO:	0/2/0			
Source:	Render	CS			
Default Value:	0x0000	0000			
Access:	RO				
Size (in bits):	32				
Trusted Type:	1				
Address:	04784h-04787h				
This register contains the valid bits for entries 0-31 of MTTLB (Texture and constant cache TLBVertex Fetch, Instruction Cache, and Command Streamer TLB).					
DWord		Bit	Description		
0		31:0	Valid bits per entry		



# MTVICTLB\_VLD1 - Valid Bit Vector 1 for MTVICTLB

Register Space:	MMIO:	MMIO: 0/2/0				
Source:	Render	CS				
Default Value:	0x00000	0000				
Access:	RO					
Size (in bits):	32					
Trusted Type:	1					
Address:	0478Ch	-0478Fh				
This register contai Streamer TLB).	ns the valio	d bits for entries	0-31 of VICTLB (Vertex Fetch, Instruction Cache, and Command			
DWord		Bit	Description			
0	31:0 Valid bits per entry					



# **RCCTLB\_VLD\_1** - Valid Bit Vector 1 for RCC

Register Space:	MMIO: 0/2/0					
Default Value:	0x0000000					
Size (in bits):	32					
Address:	0472Ch-0472F	h				
This register contai	ins the valid bi	ts for entries 0-31 of RCCT	LB			
DWord	Bit		Description			
0	31:0	Valid Bit Vector 1 for RCC				
		Default Value:	0000000h			
		Access:	RO			
		Valid bits per entry				



### **RCCTLB\_VLD1** - Valid Bit Vector 1 for RCCTLB

Register Space:	MMIO: 0/2	2/0		
Source:	RenderCS			
Default Value:	0x0000000	00		
Access:	RO			
Size (in bits):	32			
Trusted Type:	1			
Address:	04794h-04797h			
This register is reserv	ed for futur	e RCC TLB extension		
DWord Bit			Descrip	tion
0		31:0	Reserved	
			Format:	MBZ



<b>RCZTLB_VLD1</b> - Valid Bit Vector 1 for RCZTLB					
Register Space:	MMIC	D: 0/2/0			
Source:	Rende	erCS			
Default Value:	0x000	00000			
Access:	RO				
Size (in bits):	32				
Trusted Type:	1				
Address:	0479Ch-0479Fh				
This register contains	This register contains the valid bits for entries 0-31 of RCZTLB (Render Cache for Z (Depth), Hi Z, and Stencil TLB).				
DWord		Bit	Description		
0	0 31:0 Valid bits per entry				



Г

MTTLB064_VLD1 - Valid Bit Vector 1 for TLB064					
Register Space:	MMIO:	0/2/0			
Source:	VideoC	5			
Default Value:	0x00000	0000			
Access:	RO				
Size (in bits):	32				
Trusted Type:	1				
Address:	14784h-14787h				
This register contains the valid bits for entries 0-31 of MTTLB (Texture and constant cache TLB).					
DWord		Bit	Description		
0		31:0	Valid bits per entry		

Г



MTTL	. <b>B13</b> 2	2_VLD1 - V	Valid Bit Vector 1 for TLB132	
Register Space:	MMIO:	0/2/0		
Source:	VideoC	5		
Default Value:	0x00000	0000		
Access:	RO			
Size (in bits):	32			
Trusted Type:	1			
Address:	1478Ch-1478Fh			
This register contains the valid bits for entries 0-31 of MTTLB (Texture and constant cache TLB).				
DWord	Bit Description			
0	31:0 Valid bits per entry			



Г

MTTLB232_VLD1 - Valid Bit Vector 1 for TLB232					
Register Space:	MMIO:	0/2/0			
Source:	VideoC	5			
Default Value:	0x00000	0000			
Access:	RO				
Size (in bits):	32				
Trusted Type:	1				
Address:	14794h-14797h				
This register contains the valid bits for entries 0-31 of MTTLB (Texture and constant cache TLB).					
DWord		Bit	Description		
0		31:0	Valid bits per entry		



MTTL	MTTLB304_VLD1 - Valid Bit Vector 1 for TLB304					
Register Space:	MMIO: (	MMIO: 0/2/0				
Source:	VideoCS	VideoCS				
Default Value:	0x00000	0x0000000				
Access:	RO					
Size (in bits):	32					
Trusted Type:	1					
Address:	1479Ch-1479Fh					
J	This register contains the valid bits for entries 0-31 of MTTLB (Texture and constant cache TLB). Default Value = 00000000h Trusted Type = 1					
DWord		Bit	Description			
0		31:0	Valid bits per entry			



# **TLBPEND\_VLD1 - Valid Bit Vector 1 for TLBPEND registers**

Register Space:	MMIO:	: 0/2/0				
Source:	RenderCS					
Default Value:	0x0000	0x0000000				
Access:	R/W	R/W				
Size (in bits):	32					
Trusted Type:	1					
Address:	04704h-04707h					
This register contains	This register contains the valid bits for entries 32-63 of TLBPEND structure (Cycles pending TLB translation).					
DWord		Bit	Description			
0		31:0	Valid bits per entry			



Register Space:	MMIO: 0/2/0			
Default Value:	0x00000000			
Size (in bits):	32			
Address:	04734h-0473	7h		
	tains the valid b	its for entries 0-31 of ZTLB		
	tains the valid b	its for entries 0-31 of ZTLB	Description	
This register con		its for entries 0-31 of ZTLB	Description	
This register con DWord	Bit		Description 00000000h	
This register con DWord	Bit	Valid Bit Vector 1 for Z		



# CASCTLB\_VLD\_2 - Valid Bit Vector 2 for CASC

Register Space:	MMIO: 0/2/0	)		
Default Value:	0x00000000			
Size (in bits):	32			
Address:	04768h-0476	6Bh		
This register cont	tains the valid	bits for entries 0-31 of CASC	TLB	
DWord	Bit		Description	
0	31:0	Valid Bit Vector 2 for CAS	c	
		Default Value:	0000000h	
		Access:	RO	
		Valid bits per entry		



# L3TLB\_VLD\_2 - Valid Bit Vector 2 for L3

Register Space:	MMIO: 0/2/0			
Default Value:	0x00000000			
Size (in bits):	32			
Address:	04748h-0474	Bh		
This register con	tains the valid b	vits for entries 0-31 of L3TLB	3	
DWord	Bit		Description	
DWord 0	<b>Bit</b> 31:0	Valid Bit Vector 2 for L3	Description	
		Valid Bit Vector 2 for L3 Default Value:	Description 00000000h	
		l l		
		Default Value:	0000000h	
		Default Value: Access:	0000000h	



# **ZTLB\_VLD\_2** - Valid Bit Vector 2 for Z

Register Space:	MMIO: 0/2/0			
Default Value:	0x00000000			
Size (in bits):	32			
Address:	04738h-0473	Bh		
This register contai	ns the valid b	its for entries 0-31 of ZTLB		
DWord	Bit		Description	
0	31:0	Valid Bit Vector 2 for Z		
		Default Value:	0000000h	
		Access:	RO	
		Valid bits per entry		



# **CASCTLB\_VLD\_3** - Valid Bit Vector 3 for CASC

Register Space:	MMIO: 0/2/0		
Default Value:	0x00000000		
Size (in bits):	32		
Address:	0476Ch-0476	ōFh	
This register conta	ins the valid l	oits for entries 0-31 of CASC	CTLB
DWord	Bit		Description
0	31:0	Valid Bit Vector 3 for CAS	c
		Default Value:	0000000h
		Access:	RO
		Valid bits per entry	



# L3TLB\_VLD\_3 - Valid Bit Vector 3 for L3

Register Space:	MMIO: 0/2/0			
Default Value:	0x00000000			
Size (in bits):	32			
Address:	0474Ch-0474F	ĥ		
This register contai	ns the valid bi	ts for entries 0-31 of L3TLI	3	
DWord	Bit		Description	
0	31:0	Valid Bit Vector 3 for L3		
		Default Value:	0000000h	
		Access:	RO	
		Valid bits per entry		



	ZTLB_VI	.D_3 - Valid Bi	t Vector 3 for Z
Register Space:	MMIO: 0/2/0		
Default Value:	0x00000000		
Size (in bits):	32		
Address:	0473Ch-0473F	h	
This register conta	ins the valid bi	ts for entries 0-31 of ZT	_B
DWord	Bit		Description
0	31:0	Valid Bit Vector 3 for Z	
		Default Value:	0000000h
		Access:	RO
		Valid bits per entry	



# CASCTLB\_VLD\_4 - Valid Bit Vector 4 for CASC

Register Space:	MMIO: 0/2/0	)		
Default Value:	0x00000000			
Size (in bits):	32			
Address:	04770h-047	73h		
This register cont	tains the valid	bits for entries 0-31 of CASC	TLB	
DWord	Bit		Description	
0	31:0	Valid Bit Vector 4 for CASC	2	
		Default Value:	0000000h	
		Access:	RO	
		Valid bits per entry		
		Valid bits per entry		



# L3TLB\_VLD\_4 - Valid Bit Vector 4 for L3

Register Space:	MMIO: 0/2/0			
Default Value:	0x00000000			
Size (in bits):	32			
Address:	04750h-04753	3h		
This register cont	ains the valid b	its for entries 0-31 of L3TLI	В	
DWord	Bit		Description	
DWord 0	<b>Bit</b> 31:0	Valid Bit Vector 4 for L3	Description	
	1	Valid Bit Vector 4 for L3 Default Value:	Description 0000000h	
	1			
	1	Default Value:	0000000h	
		Default Value: Access:	0000000h	



# L3TLB\_VLD\_5 - Valid Bit Vector 5 for L3

Register Space:	MMIO: 0/2/0			
Default Value:	0x0000000			
Size (in bits):	32			
Address:	04754h-04757	7h		
This register contai	ns the valid bi	its for entries 0-31 of L3TL	В	
DWord	Bit		Description	
0	31:0	Valid Bit Vector 5 for L3		
		Default Value:	0000000h	
		Access:	RO	
		Valid bits per entry		



Register Space:	MMIO: 0/2/0			
Default Value:	0x0000000			
Size (in bits):	32			
Address:	04758h-0475Bh			
This register con	tains the valid b	its for entries 0-31 of L3TLI	В	
		Description		
DWord	Bit		Description	
0 0	<b>Bit</b> 31:0	Valid Bit Vector 6 for L3	Description	
		Valid Bit Vector 6 for L3 Default Value:	Description 00000000h	
		Default Value:	0000000h	
		Default Value: Access:	0000000h	



#### L3TLB\_VLD\_7 - Valid Bit Vector 7 for L3

Register Space:	MMIO: 0/2/0			
Default Value:	0x00000000			
Size (in bits):	32			
Address:	0475Ch-0475F	ĥ		
This register contain	ns the valid bi	ts for entries 0-31 of L3TLI	3	
DWord	Bit		Description	
0	31:0	Valid Bit Vector 7 for L3		
		Default Value:	0000000h	
		Access:	RO	
		Valid bits per entry		



BCSTLB_VLD - Valid Bit Vector for BCS TLB				
Register Space:	MMIC	): 0/2/0		
Source:	Blitter	CS		
Default Value:	0x000	00000		
Access:	RO			
Size (in bits):	32			
Trusted Type:	1			
Address:	24780	h-24783h		
This register contains the valid bits for entries 0-31 of BCS TLB.				
DWord Bit Description			Description	
0		31:4	Reserved	
		3:0	Valid bits per entry	



Г

BLBTLB_VLD - Valid Bit Vector for BLB TLB					
Register Space:	MMIC	D: 0/2/0			
Source:	Blitter	CS			
Default Value:	0x000	00000			
Access:	RO				
Size (in bits):	32				
Trusted Type:	1				
Address:	24784h-24787h				
This register contain	This register contains the valid bits for entries 0-31 of BLB TLB.				
DWord	Bit Description				
0	31:8		Reserved		
		7:0	Valid bits per entry		



CTX_TLB_VLD - Valid Bit Vector for CTX TLB				
Register Space:	MMIC	): 0/2/0		
Source:	Blitter	CS		
Default Value:	0x000	00000		
Access:	RO			
Size (in bits):	32			
Trusted Type:	1			
Address:	24788	h-2478Bh		
This register contains the valid bits for entries 0-31 of CTX TLB.				
DWord Bit			Description	
0	0 31:1		Reserved	
		0	Valid bits per entry	



PDTLB_VLD - Valid Bit Vector for PD TLB				
Register Space:	MMIC	0: 0/2/0		
Source:	Blitter	CS		
Default Value:	0x000	00000		
Access:	RO			
Size (in bits):	32			
Trusted Type:	1			
Address:	2478Ch-2478Fh			
This register contains the valid bits for entries 0-31 of PD TLB.				
DWord Bit Description			Description	
0	0 31:8		Reserved	
		7:0	Valid bits per entry	



	V	CS_CXT_SIZ	ZE - VCS C	Context Siz	es	
Register Space:	MMIC	): 0/2/0				
Source:	Video	CS				
Default Value:	0x000	40D02				
Access:	Read/	32 bit Write Only				
Size (in bits):	32					
Address:	121A8	3h				
DWord		Bit	Description			
0 31:21 Reserved Format:						
			Format: M		MBZ	<u>Z</u>
		20:16	VCS Context Size			_
			Format:			U5
			Value	Name		]
			4h	[Default]		]
		15:13	Reserved			
			Format:		MBZ	7
		12:8	VCR Context Size			
			Format:			U5
						1
			Value	Name		-
			Dh	[Default]		
		7:5	Reserved			
			Format:		MBZ	Ζ
		4:0	Reserved			



# VCS\_CNTR - VCS Counter for the bit stream decode engine

Register Space: N		MMIO: 0/2/0		
Source:		VideoCS		
Default Valu	le:	OxFFFFFFF		
Access:		R/W		
Size (in bits)	):	32		
Address:		12178h-1217Bh		
DWord	Bit	Description		
0	31:0	Count Value		
		Default Value:	fffffffh	
		Writing a Zero value to this register starts the counting.		
		Writing a Value of FFFF FFFF to this counter stops the counter.		

Г



٦

		VC	S_EI	R - VCS Error Identity	y Register		
Register	Space:	MMI	MMIO: 0/2/0				
Source:		Vide	VideoCS				
Default V	/alue:	0x00	000000				
Access:		R/W	С				
Size (in b	oits):	32					
Address:		120B	120B0h				
register v	will cau	se the Mas	ter Error	tent values of Hardware-Detected Erro bit in the ISR to be set. The EIR registe he appropriate bit(s) except for the unr Description	r is also used by software to clear		
0	31:16	Reserved					
-		Format:			MBZ		
-	15:0	Error Iden	tity Bits				
		Format:	Array of	Error condition bits ee the table titled	Hardware-Detected Error Bits		
		EMR regist of the Inter error by wr	his register contains the persistent values of ESR error status bits that are unmasked via the MR register. The logical OR of all (defined) bits in this register is reported in the Master Error bit f the Interrupt Status Register. In order to clear an error condition, software must first clear the ror by writing a 1 to the appropriate bit(s) in this field. If required, software should then roceed to clear the Master Error bit of the IIR.				
		Value Name Description					
		0h		[Default]			
		1h		Error occurred	Error occurred		
			Programming Notes				
	Writing a 1 to a set bit will cause that error condition to be cleared. However, the Page Error bit (Bit 4) cannot be cleared except by reset (i.e., it is a fatal error).				5		



VCS EM	R - VCS E	rror Masl	k Register

Register Spa	ace:	MMIO: 0/2/0					
Source:		VideoCS					
Default Valu	ie:	0x0000FFFF					
Access:		R/W					
Size (in bits)	:	32					
Address:		120B4h					
"Unmasked interrupt, a	l" bits will nd will pe	l be reported ersist in the E	vare to control which Error Sta in the EIR, thus setting the Ma IR until cleared by software. "N	aster Error ISR bit a Aasked" bits will no	nd possibly triggering a CPU		
			r Error conditions or CPU inter				
Undefined	or reserve	ed bits in the	Hardware Detected Error Bit	Table will always ret	urn a read value of '1'		
DWord	Bit		De	escription			
0	31:16	Reserved					
		Default Val	ue:		0000h		
		Format:			MBZ		
	15:0	Error Mask Bits					
		Format: Array of error condition mask bits See the table titled Hardware-Detected Error Bits.					
		This register contains a bit mask that selects which error condition bits (from the ESR) are reported in the EIR.					
		Value	Value Name Description				
		0000h					
		FFFFh	Masked [Default]	Will not be reporte	ed in the EIR		



VCS_ESR -	VCS Error	<b>Status</b>	Reaister

Register Spa	ce:	MMIO: 0/2/0				
Source:		VideoCS				
Default Valu	e:	0x00000000				
Access:		RO				
Size (in bits)		32				
Address:		120B8h				
definition pe	ersistent).	The EMR reg	ne current values of all Hardware-Detected Error condition bits (these are all by MR register selects which of these error conditions are reported in the persistent EIR d by software) and thereby causing a Master Error interrupt condition to be reported			
0	31:16	Reserved	Description	•		
Ū	51.10	Format:		MBZ		
	15:0	Error Status Bits				
		Format: A	Format: Array of error condition bits See the table titled Hardware-Detected Error Bits.			
		This registe bits.	his register contains the non-persistent values of all hardware-detected error condition bits.			
		Value Name		Description		
		0h	[Default]			
		UN				



#### VCS\_EXCC - VCS Execute Condition Code Register

Register Space:	MMIO: 0/2/0
Source:	VideoCS
Default Value:	0x0000000
Access:	R/W,RO
Size (in bits):	32
Trusted Type:	1
Address:	12028h

This register contains user defined and hardware generated conditions that are used by MI\_WAIT\_FOR\_EVENT commands. An MI\_WAIT\_FOR\_EVENT instruction excludes the executing ring from arbitration if the selected event evaluates to a 1, while instruction is discarded if the condition evaluates to a 0. Once excluded, a ring is enabled into arbitration when the selected condition evaluates to a 0.

This register also contains control for the invalidation of indirect state pointers on context restore.

DWord	Bit	Description						
0	31:16	Mask Bits						
		Format: Mask[15:0]						
		These bits serve as a write enable for bits 15:0. If this register is written with these bits clear the corresponding bit in the field 15:0 will not be modified. Reading these bits always returns 0s.						
	15:5	Reserved						
		Format: MBZ						
	4:0	<b>User Defined Condition Codes</b> The software may signal a Stream Sema to match the bit field specified in a WAI	. , ,	5 5				



#### VCS\_HWSTAM - VCS Hardware Status Mask Register

Register Space:	MMIO: 0/2/0
Source:	VideoCS
Default Value:	OxFFFFFFF
Access:	R/W
Size (in bits):	32
Trusted Type:	1

Address: 12098h

Access: RO for Reserved Control bits

The HWSTAM register has the same format as the Interrupt Control Registers. The bits in this register are "mask" bits that prevent the corresponding bits in the Interrupt Status Register from generating a "Hardware Status Write" (PCI write cycle). Any unmasked interrupt bit (HWSTAM bit set to 0) will allow the Interrupt Status Register to be written to the ISR location (within the memory page specified by the Hardware Status Page Address Register) when that Interrupt Status Register bit changes state.

#### **Programming Notes**

• To write the interrupt to the HWSP, the corresponding IMR bit must also be clear (enabled).

• At most 1 bit can be unmasked at any given time.

DWord	Bit	Description					
0	31:0	Hardware Status Mask Register					
		Default Value:	FFFFFFFh				
		Format: Array of Masks					
		Refer to the table in the Interrupt Control F	Refer to the table in the Interrupt Control Register section for bit definitions.				



#### VCS\_PWRCTX\_MAXCNT - VCS IDLE Max Count

Register	Register Space: MMIO: 0/2/0									
Source:		VideoC	VideoCS							
Default V	/alue:	0x0000	0040							
Access:		R/W								
Size (in b	oits):	32								
Trusted 1	Гуре:	1								
Address:		12054h	1							
This regis IDLE	ster co	ntains the tim	e in 0.64us to wa	it before tel	lling power mana	gement hardware the rend	er pipe is			
DWord	Bit				Description					
0	31:20	Reserved								
		Format:				MBZ				
-	19:0	MFX IDLE W	ait Time				_			
		Format:			Max Count					
			long the comma gement hardware		should wait befo	re ensuring the pipe is IDLE	and to let			
		Value	Name		Descrip	otion				
	00040h [Default] 0x00040 * 0.64us ~ 41us wait t				ait time					
			Programming Notes							
	This is only useable if bit 0 of the PC_PSMI_CTRL is clear.									
		• The va	• The value in this field <i>must</i> be greater than 1.							



#### VCS\_INSTPM - VCS Instruction Parser Mode Register

Register										
Register	Space:	MMIO: 0/2/0								
Source:		VideoCS								
Default \	Value:	e: 0x0000000								
Access:	ess: R/W									
Size (in bits): 32										
Address:	:	120C0h-120C3h								
of instru "Synchr	uction: onizin	can be disabled (ignored)	trol the operation of the VCS Instruction Parser. Certain classes - often useful for detecting performance bottlenecks. Also, initiated - useful for ensuring the completion (vs. only ItValue=0000 0000h							
			Programming Notes							
All rese	rved bi	ts are implemented.								
DWord	Bit		Description							
0	31:16	Masks								
		Format: Mask[15:0]								
			ables for bits 15:0. If this register is written with any of these bits n the field 15:0 will not be modified. Reading these bits always							
-										
	15:11	Reserved								
	15:11	Reserved Format:	MBZ							
	15:11 10		MBZ							
		Format:	MBZ							
	10	Format: <b>Reserved</b> Format:								
		Format: Reserved Format: TLB Invalidate								
	10	Format: Reserved Format: TLB Invalidate Format: If set, this bit allows the co valid only with the Sync fluctures	MBZ U1 ommand stream engine to invalidate the MFX TLBs. This bit is							
	9	Format: Reserved Format: TLB Invalidate Format: If set, this bit allows the co valid only with the Sync fl Note: GFX soft resets do n TLBs post reset./	U1 U1 Dommand stream engine to invalidate the MFX TLBs. This bit is ush enable.							
	10	Format: Reserved Format: TLB Invalidate Format: If set, this bit allows the co valid only with the Sync fl Note: GFX soft resets do n	U1 U1 Dommand stream engine to invalidate the MFX TLBs. This bit is ush enable.							
	9	Format: Reserved Format: TLB Invalidate Format: If set, this bit allows the co valid only with the Sync fl Note: GFX soft resets do n TLBs post reset./ Reserved Format: Memory Sync Enable	U1 Dommand stream engine to invalidate the MFX TLBs. This bit is ush enable. ot invalidate TLBs, it is up to GFX driver to explicitly invalidate							
	10 9 8:7	Format: Reserved Format: TLB Invalidate Format: If set, this bit allows the co valid only with the Sync fl Note: GFX soft resets do n TLBs post reset./ Reserved Format: Memory Sync Enable If set, this bit allows the vide	U1 Dommand stream engine to invalidate the MFX TLBs. This bit is ush enable. ot invalidate TLBs, it is up to GFX driver to explicitly invalidate MBZ							



### VCS\_INSTPM - VCS Instruction Parser Mode Register

	This field is used to request a Sync Flush operation. This bit before completing the operation. See Sync Flush Setting the Sync Flush Enable will cause a config write address 0x4f100.	ush (Programming Environment).
	Programming Note	S
	The command parser must be stopped prior to issuing th bit in register <b>BCS_MI_MODE</b> . Only after observing <b>Ring</b> Flush be issued by setting this bit. Once this bit becomes the command parser is re-enabled by clearing <b>Stop Ring</b> .	Idle set in BCS_MI_MODE can a Sync clear again, indicating flush complete,
4:0	Reserved	
	Access:	R/W
	Format:	MBZ



#### VCS\_IMR - VCS Interrupt Mask Register

Register Spa	ice:	MMIO: 0/2/0							
Source:		VideoCS	VideoCS						
Default Valu	e:	0xFFFFFFF							
Access:		R/W							
Size (in bits)	:	32							
Address:		120A8h							
The IMR register is used by software to control which Interrupt Status Register bits are masked or unmasked Unmasked bits will be reported in the IIR, possibly triggering a CPU interrupt, and will persist in the IIR until cleared by software. Masked bits will not be reported in the IIR and therefore cannot generate CPU interrup				CPU interrupt, and will persist in the IIR until					
DWord	Bit		D	escription					
0	31:0	Interrupt Mask Bit	ts						
		-	Format: Array of interrupt mask bits Refer to the Interrupt Control Register section for bit definitions.						
		This field contains a bit mask which selects which interrupt bits (from the ISR) are reported in the IIR.							
		Value Name Description							
		FFFF FFFFh	[Default]						
		0h	Not Masked	Will be reported in the IIR					
		1h	Masked	Will not be reported in the IIR					



# VCS\_MI\_MODE - VCS Mode Register for Software Interface

Register Space:MMIO: 0/2/0									
Source:		V	'ideoCS						
Default \	Value:	0	x00000200						
Access:		R	/W						
Size (in b	oits):	3	2						
Address:		1	209Ch-1209	Fh					
The MI_I	MODE	register	contains info	ormat	ion that controls software interfac	ce aspects of the command parser.			
DWord	Bit				Description				
0	31:16	<b>Masks</b> A 1 in a	bit in this fi	eld al	lows the modification of the corre	esponding bit in Bits 15:0.			
	15	Suspen	d Flush						
		Mask:			MMIO(0x209c)#31				
		Value	Name			iption			
		0h	No Delay	HW well	HW will not delay flush, this bit will get cleared by MI_SUSPEND_FLUS well				
		1h							
1	14:12	Reserv	ed						
		Access: R/W			R/W				
	11	If bit se	<b>Invalidate UHPTR enable</b> If bit set H/W clears the valid bit of BCS_UHPTR (4134h, bit 0) when current active head pointer is equal to UHPTR.						
	10	Reserv	ed						
		Forma	t:			MBZ			
	9	Ring Id	lle (Read Or	nly St	atus bit)	-			
		Access				RO			
		Writes t	to this bit are	e not d	allowed.				
			Value			Name			
		0			Parser not idle				
		1			Parser idle [Default]				
	8	read a	re must set "1" in Ring	Idle	bit to force the Ring and Com bit after setting this bit to ensu is bit for Ring to resume normal				
		50,000	Value		s ou for hang to resume normal	Name			
		0			Normal Operation				



# VCS\_MI\_MODE - VCS Mode Register for Software Interface 1 Parser is turned off

7:0	Reserved	
	Access:	R/W



#### VCS\_PP\_DCLV - VCS PPGTT Directory Cacheline Valid Register

Register S	pace:	MMIO: 0/2/0							
Source:		VideoCS							
Default Va	alue:	0x0000000, 0x0000000							
Access:		R/W							
Size (in bi	ts):	64							
Address:		12220h							
This register controls update of the on-chip PPGTT Directory Cache during a context restore. Bits that are set will trigger the load of the corresponding 16 directory entry group. This register is restored with context (prior to restoring the on-chip directory cache itself). This register is also restored when switching to a context whose LRCA matches the current CCID if the <b>Force Pl Restore</b> bit is set in the context descriptor. The context image of this register must be updated and maintained by SW; SW should not normally need to read this register. This register can also effectively be used to limit the size of a processes' virtual address space. Any access by a process that requires a PD entry in a set that is not enabled in this register will cause a error, and no fetch of the PD entry will be attempted.									
DWord	Bit	Descriptio	on						
0	63:32	Reserved							
		Format:	MBZ						
	31:0 <b>PPGTT Directory Cache Restore [132] 16 entries</b>								
		Format: Enable[32	]						
		If set, the [1st32nd] 16 entries of the directory cache are considered valid and will be broug in on context restore. If clear, these entries are considered invalid and fetch of these entries not be attempted.							



VCS_TLBPEND_RDY0 - VCS Ready Bit Vector 0 for TLBPEND Registers							
Register Space:	MMIO: 0/	2/0					
Source:	VideoCS						
Default Value:	0x000000	00					
Access:	R/W						
Size (in bits):	32						
Trusted Type:	1						
Address:	14708h-1470Bh						
This register contains	This register contains the ready bits for entries 0-31 of TLBPEND structure (Cycles pending TLB translation).						
DWord Bit			Description				
0		31:0	Ready bits per entry				



Γ

VCS_TLBPEND_RDY1 - VCS Ready Bit Vector 1 for TLBPEND Registers					
Register Space:	MMIO: 0/	2/0			
Source:	VideoCS				
Default Value:	0x0000000				
Access:	R/W				
Size (in bits):	32				
Trusted Type:	1				
Address:	1470Ch-1470Fh				
This register contains the ready bits for entries 32-63 of TLBPEND structure (Cycles pending TLB translation).					
DWord	DWord Bit Description				
0	0 31:0 <b>Ready bits per entry</b>				



#### VCS\_TIMESTAMP - VCS Reported Timestamp Count

00 set by the context restore.				
set by the context restore.				
set by the context restore.				
lue that can be used as a timestamp. This register is not reset by a				
lue that can be used as a timestamp. This register is not reset by a				
lue that can be used as a timestamp. This register is not reset by a				
s a full chipset reset is performed. lue of the PCU TSC. The PCU TSC counts 10ns increments; this Ons granularity, rolling over every 1.5 hours).				
Description				
Reserved				
Format: MBZ				
Timestamp Value				
U36				
gles every 80 ns. The upper 28 bits are zero.				



### VCS\_RNCID - VCS Ring Buffer Next Context ID Register

Register Space: N	MMIO: 0/2/0				
Source: V	/ideoCS				
Default Value: 0	)x00000000, 0x0	0000000			
Access: F	R/W				
Size (in bits): 6	54				
Address: 1	.2198h-1219Fh				
This register contains t	he next ring co	ntext ID associated with the ring buffer.			
Programming Notes					
The current context (RCCID) register can be updated indirectly from this register on a context switch event. Note that this can only be triggered when arbitration is enabled or if the current context runs dry (head pointer becomes equal to tail pointer).					
DWord	Bit Description				
0	63:0 Context ID				
	See Context Descriptor for VCS.				



## VCS\_TLBPEND\_SEC0 - VCS Section 0 of TLBPEND Entry

Register	Space:	Dace: MMIO: 0/2/0					
Source:		VideoCS					
Default <b>`</b>	Value:	0x0000000					
Access:		R/W					
Size (in l	oits):	32					
Trusted	Type:	1					
Address	:	14400h-14403h					
This regi	ister is	lirectly mapped to the TLBPEND Array in the Graphic Arbiter.					
DWord	Bit	Description					
0	31	<b>tstatus</b> his bit will be used in conjunction with the ready bit to determine the stage of the translation. ee table below.					
	30:28	GTT bits Bits 3:1 of the GTT entry used to translate the Virtual Address. 000 if translation is pending.					
	27:0	<b>urrent address</b> he value of this field depends on the stage of the TLB translation for this entry: <b>A</b> - bits 27:20 = 00, bits 19:0 = Bits 31:12 of the Virtual Address of the cycle.					



# VCS\_TLBPEND\_SEC1 - VCS Section 1 of TLBPEND Entry

Register	Space:	MM	/IO: 0/2/0					
Source:		VideoCS						
Default \	/alue:	0x0000000						
Access:		R/W	N					
Size (in b	oits):	32						
Trusted <sup>-</sup>	Гуре:	1						
Address:		145	500h-14503h					
This regi	ster is	directly ma	apped to the current Virtual Addresses in the MTTLB (Texture and constant cache TLB).					
DWord	Bit		Description					
0	31:28	<b>Current a</b> Bits 9:6 of	address f the Virtual Address of the cycle.					
	27:24	Cacheabi	ility Control Bits					
		Bit 27 (bit 3 within the four-bit field) Reserved						
		GFDT bit effective field is i <u>c</u> Bits 25:2	bit 2 within the four-bit field) is the Graphics Data Type (GFDT) bit. It is the it for this surface when writes occur. GFDT can also be set by the GTT. The e GFDT is the logical OR of this field with the GFDT from the GTT entry. This gnored for reads. 24 (bits 1:0 within the four-bit field) contain the Cacheability Control field, ontrols cacheability as described in the following table:					
		Value	Description					
		00b L	Use cacheability control bits from GTT entry.					
		01b Data is not cached.						
		11b Data is cached.						
	23	<b>ZLR bit</b> Flag to indicate this is a zero length read (A read used to calculate a Physical Address for a write).						
	22:4	TAG						
		Cycle identification TAG.						



SRC ID	
Encoding of unit ge	nerating
Constant	Value
SRCID	
VCS_RD_SRCID	"00000"
VMC_RD_SRCID	"00001"
VMX_RARD_SRCID	"00010"
VMX_BSRD_SRCID	"00011"
VMX_RSRD_SRCID	"00100"
VIP_RD_SRCID	"00101"
VLF_RD_SRCID	"00110"
VDS_ZLRD_SRCID	"00111"
VCS_WR_SRCID	"01000"
VMX_BSWR_SRCID	"01001"
VDS_WR_SRCID	"01010"
VOP_WR_SRCID	"01011"
VLF_RSWR_SRCID	"01100"
VLF_FDWR_SRCID	"01101"
VMX_RSWR_SRCID	"01110"
BSP_WR_SRCID	"01111"
VCR_RD_SRCID	"10001"
VCR_WR_SRCID	"10010"
VCS_RD_PROBE	"10011"



# VCS\_TLBPEND\_SEC2 - VCS Section 2 of TLBPEND Entry

Register Space:	MMI	O: 0/2/0				
Source:	Vide	VideoCS				
Default Value:	0x00	0x0000000				
Access:	R/W					
Size (in bits):	32					
Trusted Type:	1					
Address:	1460	14600h-14603h				
This register is	directly map	oped to the current Virtual Addresses in the MTTLB (Texture and constant cache TLB).				
DWord	Bit	Description				
0	31:11	Reserved				
	10:8	Current address				
		Bits 11:9 of the Virtual Address of the cycle.				
	7:0	0 PAT entry				
		Location of Physical Address in Physical Address Table.				



VCS	<b>_TI</b>		shold for the counter of bit stream ecode engine				
Register	Space	e: MMIO: 0/2/0					
Source:		VideoCS					
Default <b>'</b>	Value:	0x00150000					
Access:		R/W					
Size (in l	oits):	: 32					
Address	:	1217Ch-1217Fh					
DWord	Bit		Description				
0	31:0	Threshold Value					
	Default Value: 00150000h						
	The value in this register reflects the number of clocks the bit stream decode engine is expected to run. If the value is exceeded the counter is reset and an interrupt may be enabled in the device.						



VCS_TLBPEND_VLD0 - VCS Valid Bit Vector 0 for TLBPEND Registers						
Register Space:	MMIO: 0/2	2/0				
Source:	VideoCS					
Default Value:	0x0000000					
Access:	R/W					
Size (in bits):	32					
Trusted Type:	1					
Address:	14700h-14703h					
This register contains the valid bits for entries 0-31 of TLBPEND structure (Cycles pending TLB translation).						
DWord	DWord Bit Description					
0	0 31:0 Valid bits per entry					



VCS_TLBPEND_VLD1 - VCS Valid Bit Vector 1 for TLBPEND Registers						
Register Space:	MMIO: 0/2	2/0				
Source:	VideoCS					
Default Value:	0x0000000	0x0000000				
Access:	R/W	R/W				
Size (in bits):	32					
Trusted Type:	1					
Address:	ddress: 14704h-14707h					
This register contair	This register contains the valid bits for entries 32-63 of TLBPEND structure (Cycles pending TLB translation).					
DWord	DWord Bit Description					
0	0 31:0 Valid bits per entry					



#### **VFSKPD - VF Scratch Pad**

Register									
Source:		RenderCS							
Default \	/alue:	0x0000000							
Access:		R/W							
Size (in b	oits):	3	2						
Address: 02470h									
Address: 02740h-02743h									
DWord	Bit			Descrip	otion				
0	31:16	Mask B	its						
		Format	t:	Mask[15	5:0]				
		Must b	e set to mo	dify corresponding bit in Bits 1	5:0. (All bi	its implemented)			
	15	Reserve	ed						
		Format	t:		I	MBZ			
	14:7	Reserve	Reserved						
		Format	t:		I	MBZ			
	6	Reserved							
		Format	t:	MBZ					
	5	Reserve	Reserved						
		Format: MBZ							
	4:3	Reserved							
		Format: MBZ							
	2	Vertex Cache Implicit Disable Inhibit							
		Format: U1							
		Value	Name		Descrip	otion			
		0h		Allow VF to disable VS0 when S	Sequentia	al index or Prim ID is a valid Element.			
			[Default]		1				
		1h		VF never implicitly disables the Cache when required.	e vertex ca	ache. Software must disable the VS0			
	1	Disable	over Fetc	n Cache					
		Format: MBZ This bit must be '0' always.							
	0	Reserve	ed						
		Format	::		1	MBZ			



#### VICTLB\_VA - VIC Virtual page Address Registers

Register Space:	MMIO: 0/2/0				
Source:	RenderCS				
Default Value:	0x0000000				
Access:	RO				
Size (in bits):	32				
Trusted Type:	1				
Address:	04900h-0490	)3h			
J	These registers are directly mapped to the current Virtual Addresses in the VICTLB (Vertex Fetch, Instruction Cache, and Command Streamer TLB.)				
DWord	Bit		Description		
0	31:12	Address			
		Format:	GraphicsAddress[31:12]		
		Page virtual address.			
	11:0	Reserved			
		Format: MBZ			



#### VBSYNC - Video/Blitter Semaphore Sync Register

Register Space:		MMIO: 0/2/0				
Source:		VideoCS				
Default Value:		0x0000000				
Access:		R/W				
Size (in bits):		32				
Trusted Type:		1				
Address:		12040h				
This register is written by BCS, read by VCS.						
DWord	Bit	Description				
0	31:0	<b>Semaphore Data</b> Semaphore data for synchronization between video codec engine and blitter engine.				
		semaphore data for synchronization between video codec engine and bitter engine.				



#### MFX\_MODE - Video Mode Register

Register Space:	MMIO: 0/2/0
Source:	VideoCS, VideoCS2
Default Value:	0x0000000
Access:	R/W
Size (in bits):	32
Trusted Type:	1
Address:	1229Ch

DWord	Bit		Description								
0	31:16	Mask Bits									
		Format:				Mask[15:0]					
		Must be set to modify corresponding bit in Bits 15:0. (All implemented bits)									
	15:14	Reserved									
		Forma	t:				MBZ				
	13:10	Reserved									
		Forma	t:				MBZ				
	9	Per-Process GTT Enable									
		Forma	t:	Enable	able Per-Process GTT BS Mode Enable						
		Value	Name		Description						
		0h	PPGTT Disable [ <b>Default]</b>		When clear, the Global GTT will be used to translate memory access from designated commands and for commands that select the PPGTT as their translation space.						
		1h	PPGTT Enable		When set, the PPGTT will be used to translate memory access from designated commands and for commands that select the PPGTT as their translation space.						
	8	Reserved									
	7	Reserved									
		Format: MBZ									
	6:5	Reserved									
		Format: MBZ									
	4:0	Reserved									
		Forma	t:				MBZ				



#### VRSYNC - Video/Render Semaphore Sync Register

Register Space:		MMIO: 0/2/0		
Source:		VideoCS		
Default Value:		0x0000000		
Access:		R/W		
Size (in bits):		32		
Trusted Type:		1		
Address:		12044h		
This register is written by CS, read by VCS.				
DWord	Bit	Description		
0	31:0	Semaphore Data		
		Semaphore data for synchronization between video codec engine and render engine.		



#### **VS\_INVOCATION\_COUNT** - **VS Invocation Counter**

Register Space:		MMIO: 0/2/0					
Source:		RenderCS					
Default Value:		0x0000000, 0x0000000					
Access:		R/W					
Size (in bits):		64					
Trusted Type:		1					
Address:		02320h					
This reg restore.	ister	tores the value of the vertex count shaded by VS. This register is part of the context save and					
DWord	Bit	Description					
0	63:0	'S Invocation Count Report					
		Number of vertices that are dispatched as threads by the VS stage. Updated only when Statistics					
		<b>Enable</b> is set in VS_STATE (see the Vertex Shader Chapter in the 3D Volume.)					



SYNC_FLIP_STATUS - Wait For Event and Display Flip Flags
Register

Register Space:		ce: MMIO: 0/2/0				
Source:		BSpec				
Default Value:		e: 0x0000000				
Access:		R/W				
Size (in b	bits):	32				
Address	:	022D0h				
lame:		RCS Wait For Event and Display Flip Flags Register				
hortNa	me:	RCS_SYNC_FLIP_STATUS				
		is the saved value of what wait for events are still valid. This register is part of context save and C6 feature.				
		Programming Notes				
_		<b>ng Restriction:</b> r should NEVER be programmed by SW, this is for HW internal use only.				
Word	Bit	Description				
0	31	Reserved				
		Format: MBZ				
	30	Display Plane A Asynchronous Display Flip Pending				
		Format: Enable				
		This field enables a wait for the duration of a Display Plane A "Flip Pending" condition. If a flip equest is pending, the parser will wait until the flip operation has completed (i.e., the new front buffer address has now been loaded into the active front buffer registers). See Display Flip Pending Condition (in the Device Programming Interface chapter of MI Functions.				
	29	Display Plane A Synchronous Flip Display Pending				
		Format: Enable				
		This field enables a wait for the duration of a Display Plane A "Flip Pending" condition. If a flip request is pending, the parser will wait until the flip operation has completed (i.e., the new front buffer address has now been loaded into the active front buffer registers). See Display Flip Pending Condition (in the Device Programming Interface chapter of MI Functions.				
	28	Display Sprite A Synchronous Flip Display Pending				
		Format: Enable				
		This field enables a wait for the duration of a Display Sprite A "Flip Pending" condition. If a flip request is pending, the parser will wait until the flip operation has completed (i.e., the new front buffer address has now been loaded into the active front buffer registers). See Display Flip Pending				

27 Reserved



		Register			
	Format:	MBZ			
26	Display Plane B Asynchro	nous Display Flip Pending			
	Format:	Enable			
	request is pending, the par buffer address has now be	r the duration of a Display Plane B "Flip Pending" condition. If a flip rser will wait until the flip operation has completed (i.e., the new fro en loaded into the active front buffer registers). See Display Flip Pe rogramming Interface chapter of MI Functions.			
25	Display Plane B Synchror	ious Flip Display Pending			
	Format:	Enable			
	request is pending, the par buffer address has now be	r the duration of a Display Plane B Flip Pending condition. If a flip rser will wait until the flip operation has completed (i.e., the new fro en loaded into the active front buffer registers). See Display Flip Pe rogramming Interface chapter of MI Functions.			
24	Display Sprite B Synchron	nous Flip Display Pending			
	Format:	Enable			
	Condition in the Device Pro	en loaded into the active front buffer registers). See Display Flip Pe ogramming Interface chapter of MI Functions.			
23		onous Performance Flip Pending Wait Enable			
	Source:	RenderCS			
	Format:EnableThis field enables a wait for the duration of a Display Plane A Flip Pending condition. If a flip request is pending, the parser will wait until the flip operation has completed (i.e., the new front buffer address has now been loaded into the active front buffer registers). See Display Flip Pendin Condition (in the Device Programming Interface chapter of MI Functions.				
23	Reserved				
	Source:	BlitterCS			
	Format:	MBZ			
22	Display Plane A Asynchronous Flip Pending Wait Enable				
	Format:	Enable			
	This field enables a wait for the duration of a Display Plane A Flip Pending condition. If a flip request is pending, the parser will wait until the flip operation has completed (i.e., the new front buffer address has now been loaded into the active front buffer registers). See Display Flip Pend Condition (in the Device Programming Interface chapter of MI Functions.				



	R					
21	Display Plane A Synchronous Flip P	ending Wait Enable				
	Format:	Enable				
	This field enables a wait for the durati	on of a Display Plane A Flip Pending condition. If a flip				
	1 1 3	it until the flip operation has completed (i.e., the new fro				
		nto the active front buffer registers). See Display Flip Pe				
	Condition (in the Device Programming	g Interface chapter of MI Functions.				
20	Display Sprite A Synchronous Flip P	ending Wait Enable				
	Format:	Enable				
		on of a Display Sprite A Flip Pending condition. If a flip				
		it until the flip operation has completed (i.e., the new fro				
	buffer address has now been loaded into the active front buffer registers). See Display Flip Pend					
	Condition in the Device Programming	) Interface chapter of MI Functions.				
19	Reserved					
	Format:	MBZ				
18						
	Format:	Enable				
	This field enables a wait while a Display Pipe A Scan Line condition exists. This condition is define					
	-	ied in the Pipe A Display Scan Line Count Range Compa evice Programming Interface chapter of MI Functions.				
17	Display Pipe A Vertical Blank Wait I	Enable				
	Format:	Enable				
	This field enables a wait until the next Display Pipe A Vertical Blank event occurs. This event is					
	defined as the start of the next Display Pipe A vertical blank period. Note that this can cause a wa					
	for up to an entire refresh period. See Vertical Blank Event (See Programming Interface).					
16	Display Pipe A H Blank Wait Enable					
16	Display Pipe A H Blank Wait Enable Format:	Enable				
16	Format:	Enable				
16	Format: This field enables a wait until the start event is defined as the start of the nex	Enable t of next Display Pipe A Horizontal Blank event occurs. T xt Display A Horizontal blank period. Note that this can				
16	Format: This field enables a wait until the start event is defined as the start of the nex a wait for up to a line. See Horizontal	Enable t of next Display Pipe A Horizontal Blank event occurs. T xt Display A Horizontal blank period. Note that this can				
16	Format: This field enables a wait until the start event is defined as the start of the nex	Enable t of next Display Pipe A Horizontal Blank event occurs. T xt Display A Horizontal blank period. Note that this can				
16	Format: This field enables a wait until the start event is defined as the start of the nex a wait for up to a line. See Horizontal	Enable t of next Display Pipe A Horizontal Blank event occurs. T xt Display A Horizontal blank period. Note that this can Blank Event in the Device Programming Interface chapte				
	Format: This field enables a wait until the start event is defined as the start of the nex a wait for up to a line. See Horizontal MI Functions.	Enable t of next Display Pipe A Horizontal Blank event occurs. Th xt Display A Horizontal blank period. Note that this can Blank Event in the Device Programming Interface chapte				
	Format: This field enables a wait until the start event is defined as the start of the nex a wait for up to a line. See Horizontal MI Functions. Display Plane B Asynchronous Perfe	Enable t of next Display Pipe A Horizontal Blank event occurs. Th xt Display A Horizontal blank period. Note that this can o Blank Event in the Device Programming Interface chapte ormance Flip Pending Wait Enable				
	Format: This field enables a wait until the start event is defined as the start of the nex a wait for up to a line. See Horizontal MI Functions. Display Plane B Asynchronous Perfer Source: Format:	Enable t of next Display Pipe A Horizontal Blank event occurs. Th xt Display A Horizontal blank period. Note that this can o Blank Event in the Device Programming Interface chapte ormance Flip Pending Wait Enable RenderCS				



SYNC_FLIP_STATUS	- Wait For Event and	<b>Display Flip Flags</b>
	Register	

buffer address has now been loaded into the active front buffer registers). See Display Flip Pending Condition (in the Device Programming Interface chapter of MI Functions.

15	Reserved					
	Source:	BlitterCS				
	Format:	MBZ				
14	Display Plane B Asynchronous Flip Pending V	Wait I	Enable			
	Format:		Enable			
	This field enables a wait for the duration of a Di request is pending, the parser will wait until the buffer address has now been loaded into the ac Condition (in the Device Programming Interface	flip o tive f	peratic ront bu	on has completed (i.e., the new front iffer registers). See Display Flip Pending		
13	Display Plane B Synchronous Flip Pending W	/ait Er	nable			
	Format:		Enable			
	This field enables a wait for the duration of a Display Plane B Flip Pending condition. If a flip request is pending, the parser will wait until the flip operation has completed (i.e., the new front buffer address has now been loaded into the active front buffer registers). See Display Flip Pending Condition (in the Device Programming Interface chapter of MI Functions.					
12	Display Sprite B Synchronous Flip Pending W	Vait E	nable			
	Format:		Enable			
	This field enables a wait for the duration of a Display Sprite B Flip Pending condition. If a flip request is pending, the parser will wait until the flip operation has completed (i.e., the new front buffer address has now been loaded into the active front buffer registers). See Display Flip Pending Condition in the Device Programming Interface chapter of MI Functions.					
11	Reserved					
	Format:			MBZ		
10	Display Pipe B Scan Line Wait Enable					
	Format:		Enable			
	This field enables a wait while a Display Pipe B Scan Line condition exists. This condition is defined as the the start of the scan line specified in the Pipe B Display Scan Line Count Range Compare Register. See Scan Line Event in the Device Programming Interface chapter of MI Functions.					
9	Display Pipe B Vertical Blank Wait Enable					
	Format:		Enable	,		
	This field enables a wait until the next Display Pipe B Vertical Blank event occurs. This event is defined as the start of the next Display Pipe B vertical blank period. Note that this can cause a wait for up to an entire refresh period. See Vertical Blank Event (See Programming Interface).					



SYNC	_FLIP_	STATUS	- Wait For Event and Display Flip Flags				
			Register				
	B Display I	Pipe B H Blank V	Vait Enable				
	Format:	•	Enable				
	event is c a wait for	This field enables a wait until the start of next Display Pipe B Horizontal Blank event occurs. This event is defined as the start of the next Display B Horizontal blank period. Note that this can cause a wait for up to a line. See Horizontal Blank Event in the Device Programming Interface chapter of MI Functions.					
7	5 Reserved	1					
	Format:		MBZ				
4	4:0 <b>Condition Code Wait Select</b> This field enables a wait for the duration that the corresponding condition code is active. These enable select one of 15 condition codes in the EXCC register, that cause the parser to wait until to condition-code in the EXCC is cleared.						
	Value	Name	Description				
	0h	Not Enabled	Condition Code Wait not enabled				
	1h-5h	Enabled	Condition Code select enabled; selects one of 5 codes, 0 - 4				
	6h-15h						
		Programming Notes					
	unimple	mented conditio	n codes are implemented. The parser operation is UNDEFINED if an n code is selected by this field. The description of the EXCC register ers) lists the codes that are implemented.				



SYNC_FLIP_STATUS_1 - Wait For Event and Display Flip
Flags Register 1

Register Space:	MMIO: 0/2/0
Source:	BSpec
Default Value:	0x0000000
Access:	R/W
Size (in bits):	32
Address:	022D4h
Name:	RCS Wait For Event and Display Flip Flags Register 1
ShortName:	RCS_SYNC_FLIP_STATUS_1

This register is the saved value of what wait for events are still valid. This register is part of context save and restore for RC6 feature.

DWord	Bit	Description				
0	31:27	Reserved				
		Format:		MBZ		
	26:15	Reserved				
		Format:		MBZ		
	14:12	Reserved				
		Format:		MBZ		
	11	SyncFlush Status				
		Format:	Enable	2		
		This field toggles on completion of sync flush. This bit toggle generates Interrupt and also reports interrupt status to HWSP on sync flush done.				
	10	Display Plane C Asynchronous Display Flip Pending				
		Format:	Enable			
		This field enables a wait for the duration of a Display Plane C Flip Pending condition. If a flip request is pending, the parser will wait until the flip operation has completed (i.e., the new front buffer address has now been loaded into the active front buffer registers). See Display Flip Pending Condition (in the Device Programming Interface chapter of MI Functions.				
	9	Display Plane C Synchronous Flip Display Pending				
		Format:	Enable			
		This field enables a wait for the duration of a Display Plane C Flip Pending condition. If a flip request is pending, the parser will wait until the flip operation has completed (i.e., the new front buffer address has now been loaded into the active front buffer registers). See Display Flip Pending Condition (in the Device Programming Interface chapter of MI Functions.				
	8	Display Sprite C Synchronous Flip Display Pending				
		Format:	Enable			



SYNC_FLIP_	<b>STATUS</b>	<b>1</b> - Wait	<b>For Event</b>	and Display	Flip
	I	Flags Reg	jister 1		

This field enables a wait for the duration of a Display Sprite C Flip Pending condition. If a flip
request is pending, the parser will wait until the flip operation has completed (i.e., the new front
buffer address has now been loaded into the active front buffer registers). See Display Flip
Pending Condition in the Device Programming Interface chapter of MI Functions.

7	Display Plane C Asynchronous Performance Flip Pending Wait Enable				
	Source:	RenderCS			
	Format:	Enable			
	request is pending, the parser w buffer address has now been loa	This field enables a wait for the duration of a Display Plane C Flip Pending condition. If a flip request is pending, the parser will wait until the flip operation has completed (i.e., the new front buffer address has now been loaded into the active front buffer registers). See Display Flip Pending Condition (in the Device Programming Interface chapter of MI Functions.			
7	Reserved				
	Source:	BlitterCS			
	Format:	MBZ			
6	Display Plane C Asynchronous	Flip Pending Wait Enable			
	Format:	Enable			
	buffer address has now been loa	ill wait until the flip operation has completed (i.e., the new front ded into the active front buffer registers). See Display Flip e Programming Interface chapter of MI Functions.			
5	Display Plane C Synchronous F	lip Pending Wait Enable			
	Format:	Enable			
	This field enables a wait for the duration of a Display Plane C Flip Pending condition. If a flip request is pending, the parser will wait until the flip operation has completed (i.e., the new front buffer address has now been loaded into the active front buffer registers). See Display Flip Pending Condition (in the Device Programming Interface chapter of MI Functions.				
4	Display Sprite C Synchronous	Flip Pending Wait Enable			
	Format:	Enable			
	request is pending, the parser w buffer address has now been loa	duration of a Display Sprite C Flip Pending condition. If a flip ill wait until the flip operation has completed (i.e., the new front ided into the active front buffer registers). See Display Flip Programming Interface chapter of MI Functions.			
3	Reserved				
	Format:	MBZ			



SYNC_FLIP_STATUS_1 - W	ait For Event and Display Flip			
Flags Register 1				

		Format:	Enable
		This field enables a wait while a Display Pipe C Sca defined as the the start of the scan line specified ir Compare Register. See Scan Line Event in the Devic Functions.	the Pipe C Display Scan Line Count Range
	1	Display Pipe C Vertical Blank Wait Enable	
		Format:	Enable
		This field enables a wait until the next Display Pipe defined as the start of the next Display Pipe C verti wait for up to an entire refresh period. See Vertical	cal blank period. Note that this can cause a
	0	Display Pipe C H Blank Wait Enable	
		Format:	Enable
		This field enables a wait until the start of next Disp event is defined as the start of the next Display C H cause a wait for up to a line. See Horizontal Blank H chapter of MI Functions.	lorizontal blank period. Note that this can



# **PR\_CTR\_CTL** - Watchdog Counter Control

Register Space: MMIO: 0/2/0								
Source:		BS	Spec					
Default Value: 0x0000001								
Access: R/W			W					
Size (in bits): 32								
Address: 02178h								
Name:		R	CS Watchdog	g Counter Control				
ShortNam	ne:	PF	R_CTR_CTL					
DWord	Bit	Description						
0	31	Count S	ount Select					
		Format	ormat: U1					
			-					
		Value	Name	Description				
		0h	[Default]	Use the timestamp to increment the watchdog count (every 640ns)				
		1h		Use the fixed function clock (csclk) to increment the watchdog count				
	30:0	Counter	r Logic Op					
		Default Value: 1h						
			This field specifies the action to be taken by the clock counter to generate interrupts.					
		Writing a Zero value to this register starts the counting.						
		Writing	a Value of 0	000_0001 to this counter stops the counter.				



# PR\_CTR\_THRSH - Watchdog Counter Threshold

Register Space:		e: MMIO: 0/2/0					
Source:		BSpec					
Default \	Value:	0x00145855					
Access:		R/W					
Size (in b	oits):	32					
Address	:	0217Ch					
Name:		RCS Watchdog Counter Thresh	hold				
ShortName: PR_CTR_THRSH							
DWord	Bit		Description				
0	31:0	0 Counter Logic Threshold					
		Default Value:	00145855h				
		Format:	U32				
		This field specifies the threshold that the hardware checks against for the value of the render of counter before generating an interrupt. The counter in hardware generates an interrupt when threshold is reached, rolls over and starts counting again. The interrupt generated is the "Med Hang Notify" interrupt since this watchdog timer is intended primarily to remedy VLD hangs of the main pipeline.					

### **WR\_WATERMARK** - Write Watermark

	/								
Register Space:		: MMIO: 0/2/0							
Default \	√alue:	0x000FFEA4							
Size (in b	oits):	32							
Address:	:	04028h-0402Bh							
Write W	/aterm	nark							
DWord	Bit	Bit Description							
0	31:20	Counter Extra Bits	Description						
0	51.20	Default Value:	000000000000b						
		Access:	R/W						
		Counter Extra Bits	.,,,,,						
	19	Watermark Timeout Enabl							
	15	Default Value:		1b					
		Access:	R/W						
		Watermark timeout enable.							
	18:8	Watermark Timeout							
		Default Value:	1111111110b						
		Access:	R/W						
		Number of clocks that the write pipe queue is allowed to keep a ready write cycle,							
		without reads or writes to the queue. Once this value is met, and if the feature is							
		enabled, the watermark will be considered reach, and all pending write requests will be issued.							
	7	Watermark En							
		Default Value:		1b					
		Access:		R/W					
		Enable write request grou	ıping						
	6:0	High Watermark							



### **WR\_WATERMARK** - Write Watermark

Default Value:	0100100b		
Access: R/W			
•	to be collected before initiating a write burst. Once ntil all the available writes are requested.		



### ZTLB\_LRA\_0 - ZTLB LRA 0

1F107F00 050h-04053h Reserved Default Value: Access: Reserved Bits 4 ZTLB LRA1 Max Default Value: Access: ZTLB LRA1 Max Format: U6	Description           0b           RO           0011111b           R/W		
050h-04053h          Reserved         Default Value:         Access:         Reserved Bits         4         ZTLB LRA1 Max         Default Value:         Access:         ZTLB LRA1 Max         Default Value:         Access:         ZTLB LRA1 Max	0b RO 0011111b		
Reserved         Default Value:         Access:         Reserved Bits         4         ZTLB LRA1 Max         Default Value:         Access:         ZTLB LRA1 Max         Default Value:         Access:         ZTLB LRA1 Max	0b RO 0011111b		
ReservedDefault Value:Access:Reserved Bits4ZTLB LRA1 MaxDefault Value:Access:ZTLB LRA1 Max	0b RO 0011111b		
ReservedDefault Value:Access:Reserved Bits4ZTLB LRA1 MaxDefault Value:Access:ZTLB LRA1 Max	0b RO 0011111b		
Default Value: Access: Reserved Bits 4 <b>ZTLB LRA1 Max</b> Default Value: Access: ZTLB LRA1 Max	0b RO 0011111b		
Access: Reserved Bits <b>ZTLB LRA1 Max</b> Default Value: Access: ZTLB LRA1 Max	0011111b		
Reserved Bits          A       ZTLB LRA1 Max         Default Value:       Access:         ZTLB LRA1 Max       ZTLB LRA1 Max	0011111b		
4 <b>ZTLB LRA1 Max</b> Default Value: Access: ZTLB LRA1 Max			
Default Value: Access: ZTLB LRA1 Max			
Access: ZTLB LRA1 Max			
ZTLB LRA1 Max	R/W		
Format: U6			
Maximum value c	of programmable LRA1		
Reserved			
Default Value:	Ob		
Access:	RO		
Reserved			
Format: U1			
6 ZTLB LRA1 Min			
Default Value:	0010000b		
Access:	R/W		
ZTLB LRA1 Min			
Format: M	IBZ		
Minimum value o	Minimum value of programmable LRA1		
Reserved			
Default Value:	Ob		
Access:	RO		
Reserved Bits			
5	ZTLB LRA1 Min Format: M Minimum value c 5 <b>Reserved</b> Default Value: Access:		



## ZTLB\_LRA\_0 - ZTLB LRA 0

14:8	ZTLB LRA0 Max			
	Default Value: 1111111b			
	Access:	R/W		
	ZTLB LRA0 Max			
	Format: U1			
	Maximum value of programmable LRA0			
7	Reserved			
	Default Value:		0b	
	Access:		RO	
	Reserved			
	Format: U1			
6:0	ZTLB LRA0 Min			
	Default Value:	000000b		
	Access:	R/W		
	ZTLB LRA0 Min			
	Format: U6			
	Minimum value of programmable LRA0			



## ZTLB\_LRA\_1 - ZTLB LRA 1

Register Space:	MMIC	D: 0/2/0	
Default Value:		02F20	
Size (in bits):	32		
Address:		1h-04057h	
ZTLB TLB LRA 1			
DWord	Bit		Description
0	31:22	Reserved	
		Default Value:	000000000b
		Access:	RO
		Reserved	
		Format: MBZ	
	21:20	STC LRA	
		Default Value:	00ь
		Access:	R/W
		STC LRA	I
		Format: U6	
		Which LRA should STC use	
	19:18	HIZ LRA	
		Default Value:	00b
		Access:	R/W
		HIZ LRA	
		Format: U1	
		Which LRA should HIZ use	
	17:16	RCZ LRA	
		Default Value:	00b
		Access:	R/W
		RCZ LRA	
		Format: MBZ	
		Which LRA should RCZ use	
-	15	Reserved	
		Default Value:	0b



## ZTLB\_LRA\_1 - ZTLB LRA 1

-			
	Access:		RO
	Reserved Bits		
14:8	ZTLB LRA2 Max		
	Default Value:	0101111b	
	Access:	R/W	
	ZTLB LRA2 Max		
	Format: U1		
	Maximum value of programmable LRA2		
7	Reserved		
	Default Value:		0b
	Access:		RO
	Reserved		
	Format: MBZ		
6:0	ZTLB LRA2 Min	-	
	Default Value:	0100000b	
	Access: R/W		
	ZTLB LRA2 Min		
	Format: U6		
	Minimum value of programmable LRA2		