Intel® OpenSource HD Graphics Programmer's Reference Manual (PRM)
Volume 3 Part 3: North Display Engine Registers
(Ivy Bridge)

For the 2012 Intel® Core™ Processor Family

May 2012

Revision 1.0

NOTICE:
This document contains information on products in the design phase of development, and Intel reserves the right to add or remove product features at any time, with or without changes to this open source documentation.

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.

Contents

1. Introduction

This chapter contains the register descriptions for the display portion of a family of graphics devices.

These registers vary by devices within the family of devices, so special attention needs to be paid to which devices use which registers and register fields.

Different devices within the family may add, modify, or delete registers or register fields relative to another device in the same family based on the supported functions of that device.

Unless specifically indicated, all references to Cougarpoint (CPT) will apply to both Cougarpoint (CPT) and Pantherpoint (PPT).

1.1 Terminology

Description	Software Use	Should be implemented as
Read/Write, R/W	This bit can be read or written.	
Reserved	Don't assume a value for these bits. Writes have no effect.	Writes are ignored. Reads return zero.
Reserved: must be zero, MBZ	Software must always write a zero to these bits. This allows new features to be added using these bits that will be disabled when using old software and as the default case.	Writes are ignored. Reads return zero. Maybe be connected as Read/Write in future projects.
Reserved: PBC, software must preserve contents	Software must write the original value back to this bit. This allows new features to be added using these bits.	Read only.
Read Only	This bit is read only. The read value is determined by hardware. Writes to this bit have no effect.	According to each specific bit. The bit value is determined by hardware and not affected by register writes to the actual bit.
Read/Clear, Read/Write Clear	This bit can be read. Writes to it with a one cause the bit to clear.	Hardware events cause the bit to be set and the bit will be cleared on a write operation where the corresponding bit has a one for a value.
Double Buffered	Write when desired. Read gives the unbuffered value (written value) unless specified otherwise. Written values will update to take effect after a certain point. Some have a specific arming sequence where a write to another register is required before the update can take place. This is used to ensure atomic updates of several registers.	Two stages of registers used. First stage is written into and used for readback (unless specified otherwise). First stage value is transferred into second stage at the update point. Second stage value is used to control hardware. Arm/disarm flag for specific arming sequences.

1.2 Display Pixel Rate Limitations

The maximum display pixel rate is limited by factors including the memory bandwidth and latency available to display, maximum watermark values, maximum display PLL frequencies, maximum bandwidth supported by the port technology, and restrictions within the display pipes.

DisplayPort Restrictions (maximum may be limited by other factors):

Port	Configuration	Maximum Pixel Rate
DisplayPort A (eDP)	6 bits per color	388 MHz
DisplayPort A (eDP)	8 bits per color	348 MHz
DisplayPort B, C, D	6 bits per color with FDI using 4 lanes	357 MHz
DisplayPort B, C, D	6 bits per color with FDI using 3 lanes	315 MHz
DisplayPort B, C, D	8 bits per color	348 MHz

Display Pipe Restrictions (maximum may be limited by other factors):

Within a display pipe the restrictions on the maximum pixel rate are based on the planes enabled, the pixel format of those planes, panel fitting, and sprite scaling.

The restriction is found with the following formula:

```
// Find the ratio for each plane
For each enabled plane (primary and sprite) {
    If sprite scaling is enabled {
        If pixel format is 16bpp {Plane Ratio = 32/33}
        Else If pixel format is 32bpp {Plane Ratio = 16/19}
        Else If pixel format is 64bpp {Plane Ratio = 8/12}
        Else {Plane Ratio = 1/1}
    }
    Else If both sprite and primary planes are enabled {
        If pixel format is 32bpp {Plane Ratio = 16/17}
        Else If pixel format is 64bpp {Plane Ratio = 8/10}
        Else {Plane Ratio = 1/1}
    }
    Else If only a single plane is enabled {
        If pixel format is 64bpp {Plane Ratio = 8/9}
        Else {Plane Ratio = 1/1}
    }
    Else {Plane Ratio = 1/1}
}

// Adjust for sprite down-scaling
If sprite scaling is enabled and sprite destination width < sprite source width {
```

Sprite Plane Ratio = Sprite Plane Ratio * (sprite destination width / sprite source width)

}

// Select the worst ratio, Cursor does not contribute to any restrictions on pipe ratio

Pipe Ratio = Minimum(Sprite Plane Ratio, Primary Plane Ratio)

// Adjust for panel fitter horizontal down-scaling

If panel fitting is enabled and panel fitter window horizontal size < pipe horizontal source size {

Pipe Ratio = Pipe Ratio * (panel fitter window horizontal size / pipe horizontal source size)

}

// Adjust for 90% rule

Pipe Ratio = Minimum(Pipe Ratio, 9/10)

The resulting Pipe Ratio gives the ratio of the maximum allowed pixel rate for this display pipe divided by the core display clock frequency (CDCLK).

On Ivybridge the core display clock frequency is 400 MHz.

Example Primary plane 32bpp, sprite plane 16bpp, sprite up-scaling, panel fitting down-scaling 1/1.12:

Primary ratio = 16/19

Sprite ratio = 32/33

Pipe ratio = 16/19 * 1/1.12

Maximum pipe pixel rate = 16/19 * 1/1.12 * 400 MHz = 300 MHz

Example Primary plane 64bpp, sprite plane 32bpp, sprite up-scaling, no panel fitting:

Primary ratio = 8/12

Sprite ratio = 16/19

Pipe ratio = 8/12

Maximum pipe pixel rate = 8/12 * 400 MHz = 266 MHz

2. Display Mode Set Sequence

Wait Values
PCH clock reference source and PCH SSC modulator warmup = 1uS
PCH FDI receiver PLL warmup = 25us
PCH DPLL warmup = 50uS
CPU DisplayPort PLL warmup = 20uS
CPU FDI transmitter PLL warmup = 10us
DMI latency = 20uS
FDI training pattern 1 time = 0.5uS
FDI training pattern 2 time = 1.5uS
FDI idle pattern time = 31uS
FDI auto training time = 5uS

Enable Sequence
1. Enable panel power as needed to retrieve panel configuration a. Enable panel power override using AUX VDD enable override bit b. Wait for delay given in panel requirements c. Leave panel power override enabled until later step
2. Enable PCH clock reference source and PCH SSC modulator, wait for warmup (Can be done anytime before enabling port)
3. If enabling CPU embedded DisplayPort A: (Can be done anytime before enabling CPU pipe or port) a. Enable PCH 120MHz clock source output to CPU, wait for DMI latency b. Configure and enable CPU DisplayPort PLL in the DisplayPort A control register, wait for warmup
4. If enabling port on PCH: (Must be done before enabling CPU pipe or FDI) a. Enable PCH FDI Receiver PLL, wait for warmup plus DMI latency b. Switch from Rawclk to PCDclk in FDI Receiver c. Enable CPU FDI Transmitter PLL, wait for warmup
5. Enable CPU panel fitter if needed (Can be done anytime before enabling CPU pipe)
6. Configure CPU pipe timings, M/N/TU, and other pipe settings (Can be done anytime before enabling CPU pipe)
7. Enable CPU pipe
8. Configure and enable CPU planes (VGA or hires)
9. If enabling port on PCH: a. Program PCH FDI Receiver TU size same as Transmitter TU size for TU error checking b. If Auto Train FDI i. Set pre-emphasis and voltage (iterate if training steps fail) ii. Enable CPU FDI Transmitter and PCH FDI Receiver with auto training enabled Do not change auto-traing setting while FDI is enabled iii. Wait for FDI auto training time iv. Read CPU FDI Transmitter register for auto train done • If not done, see note on FDI training failure handling v. Enable PCH FDI Receiver Fill Start and Fill End Error Correction

Wait Values

c. Configure and enable PCH DPLL, wait for PCH DPLL warmup (Can be done anytime before enabling PCH transcoder)
d. Configure DPLL_SEL to set the DPLL to transcoder mapping and enable DPLL to the transcoder
e. Configure DPLL_CTL DPLL HDMI multiplier
f. Configure PCH transcoder timings, M/N/TU, and other transcoder settings (should match CPU settings)
g. Configure and enable Transcoder DisplayPort Control if DisplayPort will be used
Workaround: Set timing override (transcoder A 0xF0064, transcoder B 0xF1064, transcoder C 0xF2064) bit 31 = 1
h. Enable PCH transcoder

10. Enable ports
DisplayPort must enable in training pattern 1
11. Enable panel power through panel power sequencing
12. Wait for panel power sequencing to reach enabled steady state
13. Disable panel power override
14. If DisplayPort, complete link training
15. Enable panel backlight
16. If Audio will be used, follow audio enable sequence documented in the audio registers section

Disable Sequence

1. If Audio is used, follow audio disable sequence documented in the audio registers section.
2. Disable panel backlight
3. Disable panel power through panel power sequencing
4. Disable CPU planes (VGA or hires)
5. Disable CPU pipe
6. Wait for CPU pipe off status (CPU pipe config register pipe state)
7. Disable CPU panel fitter (Can be done anytime after CPU pipe is off)
8. If disabling DisplayPort on PCH, write the DisplayPort control register bit 31 to 0b.
9. If disabling CPU embedded DisplayPort A
a. Disable port
b. Disable CPU DisplayPort PLL in the DisplayPort A control register
c. Disable PCH 120MHz clock source output to CPU
10. If disabling any port on PCH:
a. Disable CPU FDI Transmitter and PCH FDI Receiver
Workaround: If Auto Train FDI, clear the transmitter auto training enable bit in the same write as the transmitter is disabled and clear the receiver auto training enable bit in the same write as the receiver is disabled.
b. Disable port
c. Disable PCH transcoder
d. Wait for PCH transcoder off status (PCH transcoder config register transcoder state)
Workaround: Clear timing override (transcoder A 0xF0064, transcoder B 0xF1064, transcoder C 0xF2064) bit 31 = 0.
e. Disable Transcoder DisplayPort Control if DisplayPort was used
f. Disable Transcoder DPLL Enable bit in DPLL_SEL
g. Disable PCH DPLL (Can be done anytime after PCH ports and transcoder are off)
h. If no other PCH transcoder is enabled
i. Switch from PCDclk to Rawclk in PCH FDI Receiver
ii. Disable CPU FDI Transmitter PLL

Wait Values
iii. Disable PCH FDI Receiver PLL 11. If SSC is no longer needed, disable PCH SSC modulator 12. If clock reference no longer needed, disable PCH clock reference source
Pipe timings change
Use complete disable sequence followed by complete enable sequence with new mode programming. Pipe source size can be changed on the fly when panel fitting is enabled.
Notes
CPU FDI Transmitter should not be set to idle while PCH transcoder is enabled. **FDI training failure handling:** When a failure is detected, reread the failing register bit at least once to confirm failure, then disable CPU FDI Transmitter and PCH FDI Receiver and return to the start of FDI training sequence to retry training. Retraining should iterate through the available pre-emphasis and voltage settings. Each setting should be tried at least twice before failing mode set.

3. North Display Engine Shared Functions

3.1 VGA

The VGA Control register is located here. The VGA I/O registers are located in the VGA Registers document.

3.1.1 VGA_CONTROL-VGA Control

VGA_CONTROL		
Register Space:		MMIO: 0/2/0
Default Value:		0x00000000
Access:		R/W
Size (in bits):		32
Address:	41000h-41003h	
Name:	VGA Control	
ShortName:	VGA_CONTROL	

Note: VGA requires panel fitting to be enabled.
Note: VGA is always connected to pipe A.

DWord	Bit	Description
0	31	**VGA Display Disable** This bit will disable the VGA compatible display mode. It has no effect on VGA register or A0000-BFFFF memory aperture accesses which are controlled by the PCI configuration and VGA register settings. VGA display should only be enabled if all display planes other than VGA are disabled.

Value	Name	Description
0b	Enable **[Default]**	VGA Display Enabled
1b	Disable	VGA Display Disabled

Programming Notes
The VGA SR01 screen off bit must be programmed when enabling and disabling VGA. See the VGA Registers document.
KVMR sprite can temporarily override VGA display to be disabled, causing this bit to become 1b (Disable). In order to properly disable VGA this bit should be programmed to 1b even if it already reads as 1b.
Workaround : Program registers 42000h bits 31:29 = 101b and 42004h bit 25 = 0b before enabling VGA display and keep them at those values while VGA display is enabled. It is safe to have those values even when VGA display is disabled.

	30:27	**Reserved**
		Format: PBC
	26	**VGA Border Enable**

VGA_CONTROL

This bit determines if the VGA border areas are included in the active display area. The border will be scaled along with the pixel data.

Value	Name	Description
0b	Disable **[Default]**	VGA border areas are not displayed
1b	Enable	VGA border areas are displayed

25 **Reserved**

Format:	PBC

24 **Pipe CSC Enable**

This bit enables pipe color space conversion for the VGA pixel data. CSC mode in the pipe CSC registers must be set to match the format of the VGA pixel data.

Value	Name	Description
0b	Bypass **[Default]**	VGA pixel data bypasses the pipe color space conversion logic
1b	Pass	VGA pixel data passes through the pipe color space conversion logic

23:21 **Reserved**

Format:	PBC

20 **Legacy 8Bit Palette En**

This bit only affects reads and writes to the palette through VGA I/O addresses. In the 6-bit mode, the 8-bits of data are shifted up two bits on the write (upper two bits are lost) and shifted two bits down on the read. It provides backward compatibility for original VGA programs (in its default state) as well as VESA VBE support for 8-bit palette. It does not affect palette accesses through the palette register MMIO path.

Value	Name	Description
0b	6 bit DAC **[Default]**	6-bit DAC
1b	8 bit DAC	8-bit DAC

17:16 **Reserved**

Format:	PBC

7:6 **Blink Duty Cycle**

Controls the VGA text mode blink duty cycle relative to the VGA cursor blink duty cycle.

Value	Name	Description
00b	100% **[Default]**	100% Duty Cycle,% Full Cursor Rate
01b	25%	25% Duty Cycle, 1/2 Cursor Rate
10b	50%	50% Duty Cycle, 1/2 Cursor Rate
11b	75%	75% Duty Cycle, 1/2 Cursor Rate

5:0 **VSYNC Blink Rate**

Controls the VGA blink rate in terms of the number of VSYNCs per on/off cycle. These bits are programmed with the (VSYNCs/cycle)/2-1. The proper programming of this register is determined by the VSYNC rate that the display requires when in a VGA display mode.

3.2 Frame Buffer Compression

3.2.1 FBC_CFB_BASE-FBC Compressed Buffer Address

FBC_CFB_BASE		
Register Space:		MMIO: 0/2/0
Default Value:		0x00000000
Access:		R/W
Size (in bits):		32
Address:	43200h-43203h	
Name:	FBC Compressed Buffer Address	
ShortName:	FBC_CFB_BASE	

The contents of this register can not be changed while compression is enabled.

DWord	Bit	Description
0	31:28	**Reserved** Format: MBZ
	27:12	**CFB Offset Address** This register specifies offset of the Compressed Frame Buffer from the base of stolen memory. The buffer must be 4K byte aligned.
	11:0	**Reserved** Format: MBZ

3.2.2 FBC_CTL-FBC Control

FBC_CTL		
Register Space:		MMIO: 0/2/0
Default Value:		0x00000000
Access:		R/W
Size (in bits):		32
Address:	43208h-4320Bh	
Name:	FBC Control	
ShortName:	FBC_CTL	

The contents of this register can not be changed, except bit 31, while compression is enabled.
Frame Buffer Compression is only supported with 16bpp and 32bpp 8:8:8 primary plane source pixel formats. It is not supported with any 10:10:10 or 64bpp format.
Frame Buffer Compression is only supported with memory surfaces of 4096 lines or less and pipe source sizes of 4096 pixels by 2048 lines or less.

DWord	Bit	Description
0	31	**Enable FBC** This bit is used to globally enable FBC function at the next Vertical Blank start. Frame buffer compression can only be enabled after the selected primary plane has been enabled for

one or more vertical blanks and must be disabled before disabling the primary plane.

Value	Name	Description
0b	Disable **[Default]**	Disable frame buffer compression
1b	Enable	Enable frame buffer compression

Programming Notes	Project
Workaround (WaFbcAsynchFlipDisableFbcQueue) : Display register 42000h bit 22 must be set to 1b for the entire time that Frame Buffer Compression is enabled.	
Workaround (WaFbcDisableDpfcClockGating) : Display register 42020h bit 9 must be set to 1b for the entire time that Frame Buffer Compression is enabled.	

30:29 **Plane Select**

Value	Name	Description
00b	Primary Plane A **[Default]**	Primary Plane A
01b	Primary Plane B	Primary Plane B
10b	Primary Plane C	Primary Plane C
11b	Reserved	Reserved

28 **CPU Fence Enable**

Value	Name	Description
0b	No CPU Disp Buf **[Default]**	Display Buffer is not in a CPU fence. No modifications are allowed from CPU to the Display Buffer
1b	CPU Disp Buf	Display Buffer exists in a CPU fence

27:25 **Reserved**

Format:	MBZ

14:11 **Reserved**

Format:	MBZ

7:6 **Compression Limit**

This register sets a minimum limit on compression. This determines the maximum size of the compressed frame buffer. Display lines that do not meet the compression limit will not be compressed, so the best compression will be achieved with a 1:1 ratio.

Compression Ratio 1, Pixel Format 16 bpp - Not Supported

Compression Ratio 1, Pixel Format 32 bpp - Supported (CFB=FB)

Compression Ratio 1/2, Pixel Format 16 bpp - Supported (CFB=FB)

Compression Ratio 1/2, Pixel Format 32 bpp - Supported (CFB=1/2 FB)

Compression Ratio 1/4, Pixel Format 16 bpp - Supported (CFB=1/2FB)

Compression Ratio 1/4, Pixel Format 32 bpp - Supported (CFB=1/4 FB)

FB = Frame Buffer Size

CFB = Compressed Frame Buffer Size

Value	Name	Description
00b	1:1 **[Default]**	1:1 compression, compressed buffer is the same size as the uncompressed buffer
01b	2:1	2:1 compression, compressed buffer is one half the size of the uncompressed buffer.

FBC_CTL				
		10b	4:1	4:1 compression, compressed buffer is one quarter the size of the uncompressed buffer.
		11b	Reserved	Reserved
	5:4	**Write Back Watermark** The compressed data write back engine waits for this number of entries to be ready before writing the data out to memory.		

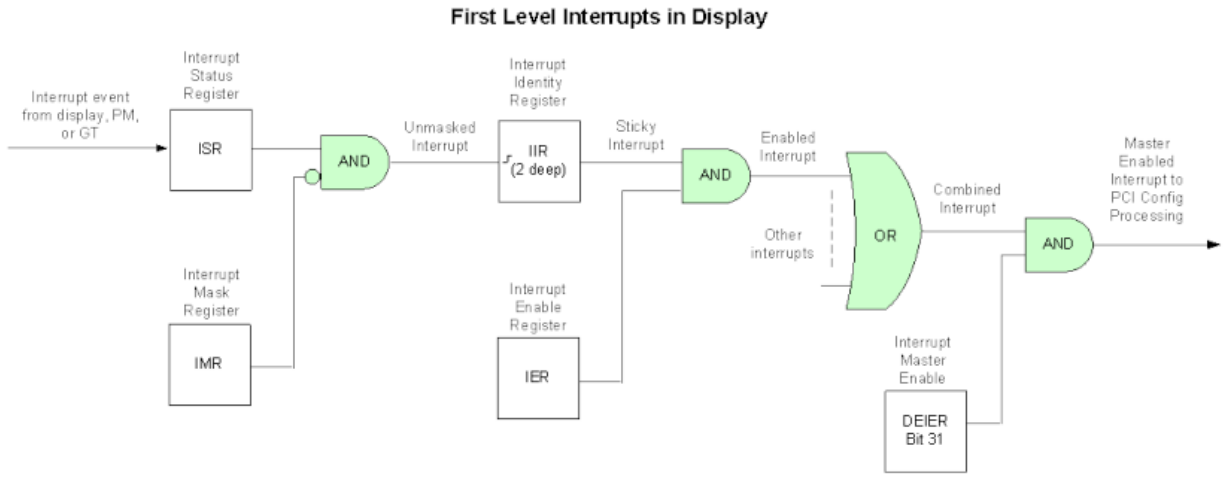
Value	Name	Description
00b	4 **[Default]**	4 entries
01b	8	8 entries
10b	16	16 entries
11b	32	32 entries

	3:0	**CPU Fence Number**

Default Value:	0000b Fence 0

Programming Notes
Restriction : This field must be programmed to 0000b.

3.3 Interrupts



For every first level interrupt bit:

The interrupt event comes in.

There may be more levels of interrupt handling behind each event. For example the PCH Display interrupt event is the result of the SDE interrupt registers.

The interrupt event goes to the Interrupt Status Register (ISR) where live status can be read back.

The live status is not useful for pulse interrupt events due to the short period that the status will be present.

The interrupt event is ANDed with the inverted Interrupt Mask Register (IMR) to create the unmasked interrupt.

Only unmasked interrupts will proceed.

The unmasked interrupt rising edge sets the sticky bit in the Interrupt Indentity Register (IIR).

The IIR can be cleared by writing a 1 to it.

The IIR can queue up to two interrupt events. When the IIR is cleared, it will set itself again if a second event was stored.

The sticky interrupt is ANDed with the Interrupt Enable Register (IER) to create the enabled interrupt.

Only enabled interrupts will proceed.

All enabled interrupts are then ORed to create the combined interrupt.

The combined interrupt is ANDed with the Master Interrupt Enable (DEIER Bit 31) to create the master enabled interrupt.

Only a master enabled interrupt will proceed.

The master enabled interrupt then goes to PCI device 2 configuration registers PCISTS2, PCICMD2, and MC which control the MSI and line interrupt.

A Function Level Reset (FLR) or Reset Warn will reset all graphics interrupt logic, causing the master enabled interrupt to de-assert which can cause the MSI or line interrupt to de-assert.

3.3.1 Display Engine Interrupt Bit Definition

Display Engine Interrupt Bit Definition		
Register Space:	MMIO: 0/2/0	
Project:		
Default Value:	0x00000000	
Size (in bits):	32	
The DE_IIR and GT_IIR and PM_IIR are ORed together to generate the CPU interrupt. The Display Engine Interrupt Control Registers all share the same bit definitions from this table.		
DWord	**Bit**	**Description**
0	31	**Master Interrupt Control** This bit exists only in the DEIER Display Engine Interrupt Enable Register. This is the master control for the Display to CPU interrupt. This bit must be set to 1 for any interrupts to propagate to the system.
	30	**Error Interrupts Combined** This is an active high level while any of the Error Interrupt bits are set.
	29	**GSE** This is an active high pulse on the GSE system level event.
	28	**PCH Display interrupt event** This is an active high level while there is an interrupt being generated by the PCH Display. It will stay asserted until the interrupts in the PCH Display are all cleared.
	27	**DisplayPort A Hotplug**

Display Engine Interrupt Bit Definition		
		This is an active high level while either of the Digital Port A Hot Plug Interrupt Detect Status register bits are set.
	26	**AUX Channel A** This is an active high pulse on the AUX A done event.
	25	**DPST histogram event** This is an active high pulse on the DPST histogram event.
	24	**DPST phase in event** This is an active high pulse on the DPST phase in event.
	23:15	**Reserved**
	14	**Sprite Plane Flip Done C** This is an active high pulse when a sprite plane flip is done.
	13	**Primary Plane Flip Done C** This is an active high pulse when a primary plane flip is done.
	12	**Line Compare Pipe C** This is an active high level for the duration of the selected pipe scan line.
	11	**Vsync Pipe C** This is an active high level for the duration of the pipe vertical sync.
	10	**Vblank Pipe C** This is an active high level for the duration of the pipe vertical blank.
	9	**Sprite Plane Flip Done B** This is an active high pulse when a sprite plane flip is done.
	8	**Primary Plane Flip Done B** This is an active high pulse when a primary plane flip is done.
	7	**Line Compare Pipe B** This is an active high level for the duration of the selected pipe scan line.
	6	**Vsync Pipe B** This is an active high level for the duration of the pipe vertical sync.
	5	**Vblank Pipe B** This is an active high level for the duration of the pipe vertical blank.
	4	**Sprite Plane Flip Done A** This is an active high pulse when a sprite plane flip is done.
	3	**Primary Plane Flip Done A** This is an active high pulse when a primary plane flip is done.
	2	**Line Compare Pipe A** This is an active high level for the duration of the selected pipe scan line.
	1	**Vsync Pipe A** This is an active high level for the duration of the pipe vertical sync.
	0	**Vblank Pipe A** This is an active high level for the duration of the pipe vertical blank.

3.3.2 GT Interrupt Bit Definition

GT Interrupt Bit Definition		
Register Space:		MMIO: 0/2/0
Project:		
Default Value:		0x00000000
Size (in bits):		32
The DE_IIR and GT_IIR and PM_IIR are ORed together to generate the CPU interrupt. The GT Interrupt Control Registers all share the same bit definitions from this table.		
DWord	**Bit**	**Description**
0	31	**Reserved**
	30	**Blitter AS Context Switch Interrupt**
	29	**Blitter page directory faults**
	28:27	**Reserved**
	26	**Blitter MI FLUSH DW notify**
	25	**Blitter Command Streamer error interrupt**
	24	**Billter MMIO sync flush status**
	23	**Reserved**
	22	**Blitter Command Streamer MI USER INTERRUPT**
	21	**Reserved**
	20	**Video AS Context Switch Interrupt**
	19	**Video page directory faults**
	18	**Video Command Streamer Watchdog counter exceeded**
	17	**Reserved**
	16	**Video MI FLUSH DW notify**
	15	**Video Command Streamer error interrupt**
	14	**Video MMIO sync flush status**
	13	**Reserved**
	12	**Video Command Streamer MI USER INTERRUPT**
	11:10	**Reserved**
	9	**Render Monitor Buffer Half Full**
	8	**Render AS Context Switch Interrupt**
	7	**Render page directory faults**
	6	**Render Command Streamer Watchdog counter exceeded**
	5	**L3 Parity Error**
	4	**Render PIPE CONTROL notify**
	3	**Render Command Streamer error interrupt**
	2	**Render MMIO sync flush status**
	0	**Render Command Streamer MI USER INTERRUPT**

3.3.3 Power Management Interrupt Bit Definition

Power Management Interrupt Bit Definition		
Register Space:	MMIO: 0/2/0	
Project:		
Default Value:	0x00000000	
Size (in bits):	32	
The DE_IIR and GT_IIR and PM_IIR are ORed together to generate the CPU interrupt. The Power Management Interrupt Control Registers all share the same bit definitions from this table.		
DWord	**Bit**	**Description**
0	31:26	**Reserved** Format: MBZ
	25	**PCU pcode2driver mailbox event**
	24	**PCU Thermal Event**
	23:7	**Reserved** Format: MBZ
	6	**Render Frequency Downward Timeout During RC6 interrupt**
	5	**RP UP threshold interrupt**
	4	**RP DOWN threshold interrupt**
	3	**Reserved** Format: MBZ
	2	**Render geyserville UP evaluation interval interrupt**
	1	**Render geyserville Down evaluation interval interrupt**
	0	**Reserved** Format: MBZ

3.3.4 ISR-Interrupt Status

ISR	
Register Space:	MMIO: 0/2/0
Project:	
Default Value:	0x00000000
Access:	RO
Size (in bits):	32
Address:	44000h-44003h
Name:	DE Interrupt Status
ShortName:	DE_ISR
Address:	44010h-44013h
Name:	GT Interrupt Status
ShortName:	GT_ISR
Address:	44020h-44023h
Name:	PM Interrupt Status

ISR		
ShortName:	PM_ISR	
See the interrupt bit definition tables to find the source event for each interrupt bit.		
DWord	**Bit**	**Description**
0	31:0	**Interrupt Status Bits** This field contains the non-persistent values of all interrupt status bits. The IMR register selects which of these interrupt conditions are reported in the persistent IIR.

Value	Name	Description
0b	Condition Doesn't exist	Interrupt Condition currently does not exist
1b	Condition Exists	Interrupt Condition currently exists

Programming Notes
Restriction : Some inputs to this register are short pulses; therefore software should not expect to use this register to sample these conditions.

3.4 IMR-Interrupt Mask

IMR	
Register Space:	MMIO: 0/2/0
Default Value:	0x00000000
Access:	R/W
Size (in bits):	32
Address:	44004h-44007h
Name:	DE Interrupt Mask
ShortName:	DE_IMR
Address:	44014h-44017h
Name:	GT Interrupt Mask
ShortName:	GT_IMR
Address:	44024h-44027h
Name:	PM Interrupt Mask
ShortName:	PM_IMR

For GT command streamer interrupts DO NOT use this register to mask interrupt events. Instead use the individual GT command streamer MASK bits.
For PM interrupts DO NOT use this register to mask interrupt events. Instead use the individual PM MASK bits in the corresponding PMunit register space.
See the interrupt bit definition tables to find the source event for each interrupt bit.

DWord	Bit	Description
0	31:0	**Interrupt Mask Bits** This field contains a bit mask which selects which interrupt bits from the ISR are reported in the IIR.

Value	Name	Description
0b	Not Masked	Not Masked - will be reported in the IIR
1b	Masked	Masked - will not be reported in the IIR

3.4.1 IIR-Interrupt Identity

IIR		
Register Space:		MMIO: 0/2/0
Default Value:		0x00000000
Access:		R/WC
Size (in bits):		32
Address:	44008h-4400Bh	
Name:	DE Interrupt Identity	
ShortName:	DE_IIR	
Address:	44018h-4401Bh	
Name:	GT Interrupt Identity	
ShortName:	GT_IIR	
Address:	44028h-4402Bh	
Name:	PM Interrupt Identity	
ShortName:	PM_IIR	

See the interrupt bit definition tables to find the source event for each interrupt bit.

DWord	Bit	Description
0	31:0	**Interrupt Identity Bits** This field holds the persistent values of the interrupt bits from the ISR which are unmasked by the IMR. If enabled by the IER, bits set in this register will generate a CPU interrupt. Bits set in this register will remain set (persist) until the interrupt condition is cleared by writing a '1' to the appropriate bits. For each bit, the IIR can store a second pending interrupt if two or more of the same interrupt conditions occur before the first condition is cleared, and upon clearing the interrupt, the IIR bit will momentarily go low, then return high to indicate there is another interrupt pending

Value	Name	Description
0b	Condition Not Detected	Interrupt Condition Not Detected
1b	Condition Detected	Interrupt Condition Detected (may or may not have generated a CPU interrupt)

3.4.2 IER-Interrupt Enable

IER		
Register Space:		MMIO: 0/2/0
Project:		
Default Value:		0x00000000
Access:		R/W
Size (in bits):		32
Address:	4400Ch-4400Fh	
Name:	DE Interrupt Enable	
ShortName:	DE_IER	
Address:	4401Ch-4401Fh	
Name:	GT Interrupt Enable	
ShortName:	GT_IER	
Address:	4402Ch-4402Fh	
Name:	PM Interrupt Enable	
ShortName:	PM_IER	

See the interrupt bit definition tables to find the source event for each interrupt bit.

DWord	Bit	Description
0	31:0	**Interrupt Enable Bits** The bits in this register enable a CPU interrupt to be generated whenever the corresponding bit in the IIR becomes set. A disabled interrupt will still appear in the IIR register to allow polling of interrupt sources. The DE_IER master interrupt control bit must be set to 1 for any interrupts to propagate to the system.

Value	Name	Description
0b	Disable	Disable
1b	Enable	Enable

3.4.3 HOTPLUG_CTL-Hot Plug Control

HOTPLUG_CTL		
Register Space:		MMIO: 0/2/0
Project:		
Default Value:		0x00000000
Access:		R/W
Size (in bits):		32
Address:	44030h-44033h	
Name:	Hot Plug Control	
ShortName:	HOTPLUG_CTL	

DWord	Bit	Description
0	31:5	**Reserved**
	4	**DP A HPD Input Enable**

HOTPLUG_CTL		
		Controls the state of the hot plug detect buffer for the digital port A. The buffer state is independent of whether the port is enabled or not. **Value** / **Name** / **Description** 0b / Disable **[Default]** / Buffer disabled 1b / Enable / Buffer enabled. Hot plugs bit reflect the electrical state of the HPD pin
	3:2	**DP A HPD Short Pulse Duration** These bits define the duration of the pulse defined as a short pulse for digital port A. **Value** / **Name** / **Description** 00b / 2ms / 2 ms 01b / 4.5ms / 4.5 ms 10b / 6ms / 6 ms 11b / 100ms / 100 ms
	1:0	**DP A HPD Status** Access: R/WC This reflects hot plug detect status on the digital port A. This bit is used for either monitor hotplug/unplug or for notification of a sink event. When either a long or short pulse is detected, one of these bits will set. These bits are ORed together to go to the main ISR hotplug register bit. These are sticky bits, cleared by writing 1s to them. **Value** / **Name** / **Description** 00b / Not Detected / Digital port hot plug event not detected 1Xb / Long Pulse / Digital port long pulse hot plug event detected X1b / Short Pulse / Digital port short pulse hot plug event detected

3.4.4 ERR_INT-Error Interrupts

ERR_INT		
Register Space:	MMIO: 0/2/0	
Project:		
Default Value:	0x00000000	
Access:	R/WC	
Size (in bits):	32	
Address:	44040h-44043h	
Name:	Error Interrupts	
ShortName:	ERR_INT	

These are sticky bits, cleared by writing 1 to them. All the Error Interrupt bits are ORed together to go to the Display Engine ISR Error Interrupts Combined bit.

DWord	Bit	Description
0	31	**Poison Status** This bit is set upon receiving the poison message. **Value** / **Name** / **Description** 0b / Not Detected **[Default]** / Event not detected 1b / Detected / Event detected
	30	**Reserved** Format: MBZ

ERR_INT

29	**Invalid GTT page table entry** This bit is set upon receiving the iMPH Invalid GTT page table entry notification.	
	Value	**Name**	**Description**
	0b	Not Detected **[Default]**	Event not detected
	1b	Detected	Event detected
28	**Invalid page table entry data** This bit is set upon receiving the iMPH Invalid page table entry data notification.	
	Value	**Name**	**Description**
	0b	Not Detected **[Default]**	Event not detected
	1b	Detected	Event detected
27:24	**Reserved**	
	Format:	MBZ
23	**Sprite GTT Fault Status C** This bit is set when a GTT fault is detected for this sprite plane.	
	Value	**Name**	**Description**
	0b	Not Detected **[Default]**	Event not detected
	1b	Detected	Event detected
22	**Primary GTT Fault Status C** This bit is set when a GTT fault is detected for this primary plane.	
	Value	**Name**	**Description**
	0b	Not Detected **[Default]**	Event not detected
	1b	Detected	Event detected
21	**Cursor GTT Fault Status C** This bit is set when a GTT fault is detected for this cursor plane.	
	Value	**Name**	**Description**
	0b	Not Detected **[Default]**	Event not detected
	1b	Detected	Event detected
20	**Sprite GTT Fault Status B** This bit is set when a GTT fault is detected for this sprite plane.	
	Value	**Name**	**Description**
	0b	Not Detected **[Default]**	Event not detected
	1b	Detected	Event detected
19	**Primary GTT Fault Status B** This bit is set when a GTT fault is detected for this primary plane.	
	Value	**Name**	**Description**
	0b	Not Detected **[Default]**	Event not detected
	1b	Detected	Event detected
18	**Cursor GTT Fault Status B** This bit is set when a GTT fault is detected for this cursor plane.	
	Value	**Name**	**Description**
	0b	Not Detected **[Default]**	Event not detected
	1b	Detected	Event detected
17	**Sprite GTT Fault Status A** This bit is set when a GTT fault is detected for this sprite plane.	
	Value	**Name**	**Description**
	0b	Not Detected **[Default]**	Event not detected
	1b	Detected	Event detected
16	**Primary GTT Fault Status A**	

ERR_INT

This bit is set when a GTT fault is detected for this primary plane.

Value	Name	Description
0b	Not Detected **[Default]**	Event not detected
1b	Detected	Event detected

15 — Cursor GTT Fault Status A

This bit is set when a GTT fault is detected for this cursor plane.

Value	Name	Description
0b	Not Detected **[Default]**	Event not detected
1b	Detected	Event detected

14:12 — Reserved

Format:	MBZ

11:9 — Reserved

Format:	MBZ

6 — Pipe FIFO Underrun C

This bit is set when the pipe FIFO underrun signal is high.

Value	Name	Description
0b	Not Detected **[Default]**	Event not detected
1b	Detected	Event detected

3 — Pipe FIFO Underrun B

This bit is set when the pipe FIFO underrun signal is high.

Value	Name	Description
0b	Not Detected **[Default]**	Event not detected
1b	Detected	Event detected

0 — Pipe FIFO Underrun A

This bit is set when the pipe FIFO underrun signal is high.

Value	Name	Description
0b	Not Detected **[Default]**	Event not detected
1b	Detected	Event detected

3.5 Display Engine Render Response

3.5.1 Display Engine Render Response Message Bit Definition

Display Engine Render Response Message Bit Definition		
Project:		
Size (in bits):		32
Default Value:		0x00000000
Display Engine (DE) render response message bits come from events within the display engine. The Display Engine Render Response Message Registers all share the same bit definitions from this table.		
DWord	**Bit**	**Description**
0	31:23	**Reserved**
	22	**Pipe C Start of Horizontal Blank Event** This event will be reported on the start of the Pipe C Horizontal Blank.
	21	**Pipe C Start of Vertical Blank Event** This event will be reported on the start of the Pipe C Vertical Blank.
	20	**Pipe C Sprite Plane Flip Done Event** This event will be reported on the completion of a flip for the Pipe C Sprite Plane.
	19:16	**Reserved**
	15	**Pipe C Primary Plane Flip Done Event** This event will be reported on the completion of a flip for the Pipe C Primary Plane.
	14	**Pipe C Scanline Event** This event will be reported on the start of the scan line specified in the Pipe C Display Scan Line Count Range Compare Register.
	13	**Pipe B Start of Horizontal Blank Event** This even will be reported on the start of the Pipe B Horizontal Blank.
	12	**Reserved**
	11	**Pipe B Start of Vertical Blank Event** This even will be reported on the start of the Pipe B Vertical Blank.
	10	**Pipe B Sprite Plane Flip Done Event** This even will be reported on the completion of a flip for the Pipe B Sprite Plane.
	9	**Pipe B Primary Plane Flip Done Event** This even will be reported on the completion of a flip for the Pipe B Primary Plane.
	8	**Pipe B Scanline Event** This even will be reported on the start of the scan line specified in the Pipe B Display Scan Line Count Range Compare Register.
	7:6	**Reserved**
	5	**Pipe A Start of Horizontal Blank Event** This even will be reported on the start of the Pipe A Horizontal Blank.
	4	**Reserved**
	3	**Pipe A Start of Vertical Blank Event** This even will be reported on the start of the Pipe A Vertical Blank.
	2	**Pipe A Sprite Plane Flip Done Event** This even will be reported on the completion of a flip for the Pipe A Sprite Plane.
	1	**Pipe A Primary Plane Flip Done Event** This even will be reported on the completion of a flip for the Pipe A Primary Plane.

Display Engine Render Response Message Bit Definition		
	0	**Pipe A Scanline Event** This even will be reported on the start of the scan line specified in the Pipe A Display Scan Line Count Range Compare Register.

3.5.2 DE_RRMR - Display Engine Render Response Mask

DE_RRMR		
Register Space:		MMIO: 0/2/0
Project:		
Default Value:		0x00000000
Access:		R/W
Size (in bits):		32
Address:	44050h-44053h	
Name:	Render Response Mask	
ShortName:	DE_RRMR	

See the render response message bit definition table to find the source event for each bit.

This register is used by software to control which render response message bits are masked or unmasked.
Unmasked bits will cause a render response message to be sent and will be reported in that message.
Masked bits will not be reported and will not cause a render response message to be sent.

Events must be unmasked prior to waiting for them with a MI_WAIT_FOR_EVENT ring command, or in the case of flips or scanlines, prior to starting the flip or loading the scanline.

Unmasked events will wake render (command streamer) as they occur, so for improved power savings it is recommended to only unmask events that are required.

Programming this register can be done through MMIO or a command streamer LOAD_REGISTER_IMMEDIATE (LRI) command.
When using LRI care must be taken to follow all the programming rules for LRI targetting the display engine.

Programming Notes

The render response message is sent from the display engine to the render command streamer (CS) or blitter command streamer (BCS). The message is used to inform CS and BCS of certain display events.

Vertical and horizontal blank events occur periodically while the associated display pipe timing generator is running and will be reported in a render response to CS if un-masked here.

Scanline events occur after they have been initiated through MMIO writes or LOAD_REGISTER_IMMEDIATE to the Display Load Scan Lines register. A scanline event will be reported in a render response to CS if un-masked here.

Flip done events occur after they have been initiated through MI_DISPLAY_FLIP or MMIO write to the primary or sprite plane surface address registers.
A flip event will be reported in a render response to CS if un-masked here and the flip source is CS.
A flip event will be reported in a render response to BCS if un-masked here and the flip source is BCS.

Workaround : Always program MI_MODE 0x209C bit 14 to 1b.

Workaround (WaDisplayFlipAWaitAFlipBWaitB) : Do not cause more than one display event to be reported in a

DE_RRMR		
single render response. The following restrictions are necessary: * Do not un-mask more than one blank event at a time. Do not mix un-masked blanks with waits for scanlines or flips to CS. * Do not initiate more than one un-masked scanline event at a time. After each scanline load always wait for the scanline done before changing this mask or initiating a new scanline or CS flip. * Do not initiate more than one un-masked flip event to CS at a time. After each CS flip always wait for the CS flip done before changing this mask or initiating a new CS flip or scanline. * Do not initiate more than one un-masked flip event to BCS at a time. After each BCS flip always wait for the BCS flip done before changing this mask or initiating a new BCS flip. Flips to BCS can be mixed with flips to CS, blanks, or scanlines.		
DWord	**Bit**	**Description**
0	31:0	**DE RRMR** Format: Display Engine Render Response Message Bit Definition This field contains a bit mask which selects which events cause and are reported in the render response message.

Value	Name	Description
0b	Not Masked	Not Masked - will cause and be reported in the message
1b	Masked	Masked - will not cause or be reported in the message

3.6 Display Timestamp

This register provides an elapsed real-time value that can be used as a timestamp for GPU events over short periods of time.

The register counts at a constant frequency by adjusting the increment amount according to the actual core display clock frequency. SW therefore does not need to know the reference clock frequency.

3.6.1 TIMESTAMP_CTR-Time Stamp Counter Value

TIMESTAMP_CTR	
Register Space:	MMIO: 0/2/0
Project:	
Default Value:	0x00000000
Access:	R/WC
Size (in bits):	32
Address:	44070h-44073h
Name:	Time Stamp Counter
ShortName:	TIMESTAMP_CTR

DWord	Bit	Description
0	31:0	**TIMESTAMP Counter** This field increments every microsecond. The value in this field is latched in the Pipe Flip TIMESTAMP registers when flips occur, and in the Pipe Frame TIMESTAMP registers at start of vertical blank. The register value will reset if any value is written to it. The register is not reset by a graphics software reset.

3.7 Display Arbitration Control

3.7.1 ARB_CTL-Display Arbitration Control 1

ARB_CTL

Register Space:	MMIO: 0/2/0
Project:	
Default Value:	0x16661056
Access:	R/W
Size (in bits):	32

Address:	45000h-45003h
Name:	Display Arbitration Control 1
ShortName:	ARB_CTL

DWord	Bit	Description
0	28:26	**HP Queue Watermark** Default Value: 101b 6 The value in this register indicates the number of entries the high priority queue should have before it can be read. The value is zero based.
	25:24	**LP Write Request Limit** The value in this register indicates the maximum number of back to back LP write requests that will be accepted from a single client before re-arbitrating. Value / Name / Description: 00b / 1 / 1 01b / 2 / 2 10b / 4 **[Default]** / 4 (default) 11b / 8 / 8
	23:20	**TLB Request Limit** The value in this register indicates the maximum number of TLB requests that can be made in an arbitration loop. Zero is not a valid programming. Value / Name: [1,15] / 0110b / 6 **[Default]**
	19:16	**TLB Request InFlight Limit** The value in this register indicates the maximum number of TLB (or VTd) requests that can be in flight at any given time. Zero is not a valid programming. Value / Name: [1,15] / 0110b / 6 **[Default]**
	14:13	**Tiled Address Swizzling** DRAM configuration registers show if memory address swizzling is needed. Value / Name / Description: 00b / No Display / No display request address swizzling 01b / Enable / Enable display request address bit[6] swizzling for tiled surfaces 10b / Reserved / Reserved 11b / Reserved / Reserved
	12:8	**HP Page Break Limit**

ARB_CTL		
		The value in this register represents the maximum number of page breaks allowed in a HP request chain. Zero is not a valid programming. Value: [1,31] Name: 10000b: 16 **[Default]**
	7	**Reserved**
	6:0	**HP Data Request Limit** The value in this register represents the maximum number of cachelines allowed in a HP request chain. Zero is not a valid programming. Value: [1,127] Name: 1010110b: 86 **[Default]**

3.7.2 ARB_CTL2-Display Arbitration Control 2

ARB_CTL2		
Register Space:		MMIO: 0/2/0
Default Value:		0x00000000
Access:		R/W
Size (in bits):		32
Address:	45004h-45007h	
Name:	Display Arbitration Control 2	
ShortName:	ARB_CTL2	

DWord	Bit	Description
0	30:9	**Reserved**
	8	**Fetch Timing** The value in this register is valid only when Opportunistic Fetches are enabled. The value in this register is used to specify when an opportunistic fetch can happen. For any opportunistic fetch to happen, display should not be in the process of waking the system. Value: 0b — Name: FE inSR — Description: Fetch on falling edge of inSR Value: 1b — Name: Not inSR — Description: Fetch when not inSR
	7	**Opportunistic Fetch Behavior** The value in this register is valid only when Opportunistic Fetches are enabled. The value in this register represents the fetch behavior when an opportunistic fetch is triggered. For any opportunistic fetch to happen, display should not be in the process of waking the system. Value: 0b — Name: One Burst — Description: One Burst Only Value: 1b — Name: Fill FIFO — Description: Fill FIFO to Top
	6	**Data Buffer Partitioning** Double Buffer Update Point: Start of vertical blank on the low power pipe or not in low power mode or all pipes disabled This bit controls the data buffer partitioning when sprite LP states are used.

ARB_CTL2

		Value	Name	Description
		0b	1/2	Sprite has 1/2 and primary has 1/2 of the buffer
		1b	5/6	Sprite has 5/6 and primary has 1/6 of the buffer
	5:4	**Inflight HP Read Request Limit** The value in this register represents the maximum number of HP read request transactions that can inflight at any given time.		
		Value	Name	Description
		00b	128 HP	128 HP inflight transactions limit
		01b	64 HP	64 HP inflight transactions limit
		10b	32 HP	32 HP inflight transactions limit
		11b	16 HP	16 HP inflight transactions limit
	3:2	**Reserved**		
	1:0	**RTID FIFO Watermark** The value in this register represents the watermark value for the RTID FIFO. HP transactions will start only when the FIFO level is above or equal the watermark		
		Value	Name	Description
		00b	8 RTIDs	8 RTIDs available in FIFO
		01b	16 RTIDs	16 RTIDs available in FIFO
		10b	32 RTIDs	32 RTIDs available in FIFO
		11b	Reserved	Reserved

3.7.3 MSG_CTL-Display Message Control

MSG_CTL

Register Space:	MMIO: 0/2/0
Default Value:	0x00000000
Access:	R/W
Size (in bits):	32

Address:	45010h-45013h
Name:	Display Message Control
ShortName:	MSG_CTL

DWord	Bit	Description
0	31:2	**Reserved** Format: PBC
	1	**Wait for PCH ResetWarn Ack** Set to 1b to make north display wait for south display to acknowledge a Reset Warn. By default north display will not wait.

Value	Name
0b	Do not wait
1b	Wait

Programming Notes

MSG_CTL		
		Restriction : BIOS must set this to 1b after the PCH display BDF has been enabled and before enabling any function in the PCH display.
	0	**Wait for PCH FLR Ack** Set to 1b to make north display wait for south display to acknowledge a Function Level Reset. By default north display will not wait. **Value** / **Name** 0b – Do not wait 1b – Wait **Programming Notes** Restriction : BIOS must set this to 1b after the PCH display BDF has been enabled and before enabling any function in the PCH display.

3.8 Display Watermarks

The watermark registers are used to control the display to memory request timing. The watermarks must be programmed according to the rules provided in the “Programming Watermarks” document. The default values of the watermarks should allow the display to operate in any high power mode supported by the memory configuration. However, the default watermarks are not optimized for power or memory bandwidth efficiency.

Watermarks must enable from the bottom up, meaning if WM_LP2 is disabled, WM_LP3 must also be disabled, and if WM_LP1 is disabled, both WM_LP2 and WM_LP3 must also be disabled. Watermark latency values must increase from the bottom up, meaning WM_LP1 (if enabled) must have higher latency than WM_PIPE, and so on.

The memory latency values are provided by the MCHBAR PCU 0:0:0 0x5D10 SSKPD config register.

3.8.1 WM_PIPE-Pipe Main Watermarks

WM_PIPE	
Register Space:	MMIO: 0/2/0
Project:	
Default Value:	0x00783818
Access:	R/W
Size (in bits):	32
Address:	45100h-45103h
Name:	Pipe A Watermarks
ShortName:	WM_PIPE_A
Address:	45104h-45107h
Name:	Pipe B Watermarks
ShortName:	WM_PIPE_B
Address:	45200h-45203h

WM_PIPE		
Name:	Pipe C Watermarks	
ShortName:	WM_PIPE_C	
These are the normal watermark values.		
DWord	**Bit**	**Description**
0	31:23	**Reserved**
	22:16	**Pipe Primary Watermark** Default Value: 1111000b Number in 64Bs of data in FIFO below which the Pipe Primary Plane stream will generate requests to memory
	15	**Reserved**
	14:8	**Pipe Sprite Watermark** Default Value: 0111000b Number in 64Bs of data in FIFO below which the Pipe Sprite Plane stream will generate requests to memory
	7:6	**Reserved**
	5:0	**Pipe Cursor Watermark** Default Value: 011000b Number in 64Bs of data in FIFO below which the Pipe Cursor Plane stream will generate requests to memory

3.8.2 WM_LP-Low Power Watermarks

The Low Power (LP) watermark register will be used when only one pipe is enabled, sprite scaling is not enabled, and the power controller has requested display go into the LP state.

WM_LP	
Register Space:	MMIO: 0/2/0
Project:	
Default Value:	0x00000000
Access:	R/W
Size (in bits):	32
Address:	45108h-4510Bh
Name:	Low Power 1 Watermarks
ShortName:	WM_LP1
Address:	4510Ch-4510Fh
Name:	Low Power 2 Watermarks
ShortName:	WM_LP2
Address:	45110h-45113h
Name:	Low Power 3 Watermarks

WM_LP		
ShortName:	WM_LP3	
These are Low Power (LP) watermark values which will be used when display is in a LP state.		
DWord	**Bit**	**Description**
0	31	**Enabled** Enables this LP watermark. This bit allows the associated LP state to be used.
	30:24	**Latency** The latency associated with this LP watermark in half usecs.
	23:20	**FBC LP Watermark** Number of equivalent lines of the primary display for this watermark
	19:18	**Reserved**
	17:8	**LP Primary Watermark** Number in 64Bs of data in the display data buffer below which the Primary Plane stream will generate requests to memory.
	7:0	**LP Cursor Watermark** Number in 64Bs of data in the display data buffer below which the Cursor Plane stream will generate requests to memory.
		Programming Notes / **Project** Workaround (WaDoubleCursorLP3Latency) : The WM_LP3 cursor watermark calculation must use twice the latency value found in the SSKPD WM3 field. The primary watermark, FBC watermark, and latency value fields are not affected.

3.8.3 WM_LP_SPR-Low Power Sprite Watermark

The Low Power Sprite (LP_SPR) watermark register will be used when one pipe is enabled, a sprite is enabled, sprite scaling is not enabled, and the power controller has requested display go into the LP state. This will be used together with the associated LP watermarks for FBC, Primary, and Cursor.

WM_LP_SPR	
Register Space:	MMIO: 0/2/0
Project:	
Default Value:	0x00000000
Access:	R/W
Size (in bits):	32
Address:	45120h-45123h
Name:	Low Power 1 Sprite Watermarks
ShortName:	WM_LP1_SPR
Address:	45124h-45127h
Name:	Low Power 2 Sprite Watermarks
ShortName:	WM_LP2_SPR
Address:	45128h-4512Bh
Name:	Low Power 3 Sprite Watermarks
ShortName:	WM_LP3_SPR
This is a Low Power Sprite (LP_SPR) watermark value which will be used when display is in a LP state.	

DWord	Bit	Description
0	31:10	**Reserved**
	9:0	**LP Sprite Watermark** Number in 64Bs of data in the display data buffer below which the Sprite Plane stream will generate requests to memory.

3.9 Backlight Control

3.9.1 BLC_PWM_CTL-Backlight PWM Control

BLC_PWM_CTL

Register Space:	MMIO: 0/2/0
Project:	
Default Value:	0x00000000
Access:	R/W
Size (in bits):	32

Address:	48250h-48253h
Name:	Backlight PWM Control
ShortName:	BLC_PWM_CTL

DWord	Bit	Description
0	31	**PWM Enable** This bit enables the PWM counter logic

Value	Name	Description
0b	Disable **[Default]**	PWM disabled (drives 0 always)
1b	Enable	PWM enabled

Bit	Description
30:29	**PWM Pipe assignment** This bit assigns PWM to a pipe. The PWM function must be disabled in order to change the value of this field.

Value	Name	Description
00b	Pipe A **[Default]**	Pipe A
01b	Pipe B	Pipe B
10b	Pipe C	Pipe C
11b	Reserved	Reserved

Bit	Description
28:27	**Reserved**
26	**Phase In Interrupt Status**

Access:	R/WC

This bit will be set by hardware when a Phase-In interrupt has occurred. Clear this bit by writing a '1', which will reset the interrupt generation.

Value	Name
0b	Interrupt
1b	No interrupt

Bit	Description
25	**Phase In Enable** Setting this bit enables a PWM phase in based on the programming of the Phase In registers below. This bit clears itself when the phase in is completed.

Value	Name

BLC_PWM_CTL			
		0b	Disable
		1b	Enable
	24	**Phase In Interrupt Enable** Setting this bit enables an interrupt to be generated when the PWM phase in is completed. **Value** / **Name** 0b – Disable 1b – Enable	
	23:16	**Phase In time base** This field determines the number of VBLANK events that pass before one increment occurs. **Value** / **Name** / **Description** 00h – Invalid **[Default]** – Invalid 01h-FFh – Count – VBlank Count	
	15:8	**Phase In Count** This field determines the number of increment events in this phase in. Writes to this register should only be done when hardware-phase-ins are disabled. A value of 0 is invalid. Reads to this register can be done any time, where the value in this field indicates the number of increment events remaining to fully apply a phase-in request as hardware automatically decrements this value.	
	7:0	**Phase In Increment** This field indicates the amount to adjust the PWM duty cycle register on each increment event. This is a two's complement number.	

3.9.2 BLC_PWM_DATA-Backlight PWM Data

BLC_PWM_DATA		
Register Space:	MMIO: 0/2/0	
Project:		
Default Value:	0x00000000	
Access:	R/W (DWORD access only, no byte access)	
Size (in bits):	32	
Address:	48254h-48257h	
Name:	Backlight PWM Data	
ShortName:	BLC_PWM_DATA	

DWord	Bit	Description
0	31:16	**Reserved**
	15:0	**Backlight Duty Cycle** This field determines the number of time base events for the active portion of the PWM backlight control. This should never be larger than the frequency field. A value of zero will turn the backlight off. A value equal to the backlight modulation frequency field will be full on. This field should be updated when it is desired to change the intensity of the backlight, it will take affect at the end of the current PWM cycle. This value represents the active time of the PWM stream in PCH display raw clock periods multiplied by 128.

BLC_PWM_DATA		
		Programming Notes
		Restriction : This register must be written only as a full 32 bit dword. Byte or word writes are not supported.

3.9.3 BLM_HIST_CTL-Image Enhancement Control

BLM_HIST_CTL			
Register Space:			MMIO: 0/2/0
Project:			
Default Value:			0x00000000
Access:			R/W
Size (in bits):			32
Address:		48260h-48263h	
Name:		Image Enhancement Control	
ShortName:		BLM_HIST_CTL	

DWord	Bit	Description
0	31	**IE Histogram Enable** This bit enables the Image Enhancement histogram logic to collect data.

Value	Name	Description
0b	Disable **[Default]**	Image histogram is disabled
1b	Enable	The image histogram is enabled. When this bit is changed from a zero to a one, histogram calculations will begin after the next VBLANK of the assigned pipe.

Bit	Description
30:29	**IE Pipe** This bit assigns the IE function to a pipe. IE events will be synchronized to the VBLANK of the selected pipe. The IE function must be disabled in order to change the value of this field.

Value	Name	Description
00b	Pipe A **[Default]**	Pipe A
01b	Pipe B	Pipe B
10b	Pipe C	Pipe C
11b	Reserved	Reserved

Bit	Description
28	**Reserved**
27	**IE Modification Table Enable** This bit enables the Image Enhancement modification table.

Value	Name	Description
0b	Disable **[Default]**	Disabled
1b	Enable	Enabled. When this bit is changed from a zero to a one, modifications begin after the next VBLANK of the assigned pipe.

Bit	Description
26:25	**Reserved**
24	**Histogram Mode Select**

Value	Name	Description
0b	YUV **[Default]**	YUV Luma Mode
1b	HSV	HSV Intensity Mode

BLM_HIST_CTL

	23:16	**Sync to Phase In Count** This field indicates the phase in count number on which the Image Enhancement table will be loaded if the Sync to Phase in is enabled.
	15	**IE Table Value Format** This field indicates what format is used for the image enhancement table values.

Value	Name	Description
0b	1.9 **[Default]**	1 integer and 9 fractional bits
1b	2.8	2 integer and 8 fractional bits

	14:13	**Enhancement mode**

Value	Name	Description
00b	Direct **[Default]**	Direct look up mode
01b	Additive	Additive mode
10b	Multiplicative	Multiplicative mode
11b	Reserved	Reserved

	12	**Sync to Phase In** Setting this bit enables the double buffered registers to be loaded on the phase in count value specified instead of the next vblank.
	11	**Bin Register Function Select** This field indicates what data is being written to or read from the bin data register.

Value	Name	Description
0b	TC **[Default]**	Threshold Count. A read from the bin data register returns that bin's threshold value from the most recent vblank load event (guardband threshold trip). Valid range for the Bin Index is 0 to 31.
1b	IE	Image Enhancement Value. Valid range for the Bin Index is 0 to 32

	10:7	**Reserved**
	6:0	**Bin Register Index** This field indicates the bin number whose data can be accessed through the bin data register. This value is automatically incremented by a read or a write to the bin data register if the busy bit is not set.

3.9.4 BLM_HIST_BIN-Image Enhancement Bin Data

BLM_HIST_BIN	
Register Space:	MMIO: 0/2/0
Project:	
Default Value:	0x00000000
Access:	R/W
Size (in bits):	32
Double Buffer Update Point:	Image Enhancement: Next vblank if in normal mode, or on phase in Sync event frame if it is enabled
Address:	48264h-48267h
Name:	Image Enhancement Bin Data
ShortName:	BLM_HIST_BIN

Access to this address are steered to the correct register by programming the Bin Register Function Select and the Bin Register Index.

DWord	Bit	Description
0	31	**Busy Bit** If (BLM_HIST_CTL:Bin Register Function Select = Threshold Count) {This is a read only bit. If set, the engine is busy and the rest of the register is undefined. If clear, the register contains valid data.} Else (Image Enhancement){This bit is reserved.}
	30:22	**Reserved**
	21:0	**Bin Count or Correction Factor** If (BLM_HIST_CTL:Bin Register Function Select = Threshold Count){Bits 21:0 are read only bits. They indicate the total number of pixels in this bin, value is updated when guardband interrupt delay is met.} Else (Image Enhancement){Bits 21:10 are reserved. Bits 9:0 are read/write. The program the correction value for this bin. Writes to this register are double buffered on the next vblank if in normal mode, or on the phase in Sync event frame if it is enabled. The value written here is the 10bit corrected channel value for the lowest point of the bin. The correction value is taken as a positive number.}

3.9.5 BLM_HIST_GUARD-Histogram Threshold Guardband

BLM_HIST_GUARD

Register Space:	MMIO: 0/2/0
Project:	
Default Value:	0x00000000
Access:	R/W
Size (in bits):	32
Double Buffer Update Point:	Start of vertical blank

Address:	48268h-4826Bh
Name:	Histogram Threshold Guardband
ShortName:	BLM_HIST_GUARD

DWord	Bit	Description
0	31	**Histogram Interrupt enable** Value / Name / Description: 0b – Disable **[Default]** – Disabled 1b – Enable – This generates a histogram interrupt once a Histogram event occurs. Software must always program 1.
	30	**Histogram Event status** Access: R/WC When a Histogram event has occurred, this will get set by the hardware. For any more Histogram events to occur, the software needs to clear this bit by writing a '1'. The default state for this bit is '0' Value / Name / Description: 0b – Not Occurred **[Default]** – Histogram event has not occurred 1b – Occured – Histogram event has occurred
	29:22	**Guardband Interrupt Delay** An interrupt is always generated after this many consecutive frames of the guardband threshold being surpassed. This value is double buffered on start of vblank. A value of 0 is invalid.
	21:0	**Threshold Guardband** This value is used to determine the guardband for the threshold interrupt generation. This single value is used for all the segments. This value is double buffered on start of vblank

3.10 Color Space Conversion

These registers contain the coefficients of the pipe color space converter.

The high color channel is the most significant bits of the color. The low color channel is the least significant bits of the color. The medium color channel is the bits between high and low. For example: In RGB modes Red is in the High channel, Green in Medium, and Blue in Low. In YUV modes, U is in the High channel, Y in Medium, and V in Low.

The color space conversion registers are double buffered and are updated on the start of vertical blank following a write to the CSC Mode register for the respective pipe.

The matrix equations are as follows:

OutputHigh = (CoefficientRU * InputHigh) + (CoefficientGU * InputMedium) + (CoefficientBU * InputLow)

OutputMedium = (CoefficientRY * InputHigh) + (CoefficientGY * InputMedium) + (CoefficientBY * InputLow)

OutputLow = (CoefficientRV * InputHigh) + (CoefficientGV * InputMedium) + (CoefficientBV * InputLow)

Example programming for RGB to YUV is in the following table:

The input is RGB on high, medium, and low channels respectively.

The output is VYU on high, medium, and low channels respectively.

Program CSC_MODE to put gamma before CSC.

Program the CSC Post-Offsets to +1/2, +1/16, and +1/2 for high, medium, and low channels respectively.

The coefficients and pre and post offsets can be scaled if desired.

	Bt.601		Bt.709	
	Value	Program	Value	Program
RU	0.2990	0x1990	0.21260	0x2D98
GU	0.5870	0x0968	0.71520	0x0B70
BU	0.1140	0x3E98	0.07220	0x3940
RV	-0.1687	0xAAC8	-0.11460	0xBEA8
GV	-0.3313	0x9A98	-0.38540	0x9C58
BV	0.5000	0x0800	0.50000	0x0800
RY	0.5000	0x0800	0.50000	0x0800
GY	-0.4187	0x9D68	-0.45420	0x9E88
BY	-0.0813	0xBA68	-0.04580	0xB5E0

Example programming for YUV to RGB is in the following table:

The input is VYU on high, medium, and low channels respectively.

The output is RGB on high, medium, and low channels respectively.

Program CSC_MODE to put gamma after CSC.

Program the CSC Pre-Offsets to -1/2, -1/16, and -1/2 for high, medium, and low channels respectively.

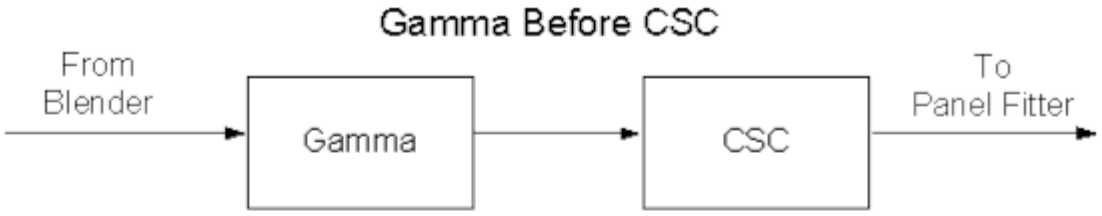
The coefficients and pre and post offsets can be scaled if desired.

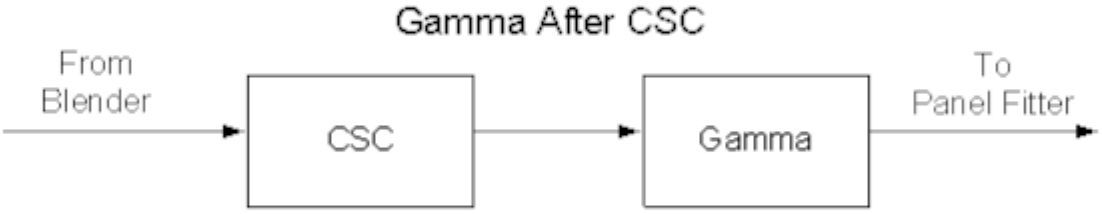
	Bt.601 Reverse		Bt.709 Reverse	
	Value	Program	Value	Program
GY	1.000	0x7800	1.000	0x7800
BY	0.000	0x0000	0.000	0x0000
RY	1.371	0x7AF8	1.574	0x7C98
GU	1.000	0x7800	1.000	0x7800
BU	-0.336	0x9AC0	-0.187	0xABF8
RU	-0.698	0x8B28	-0.468	0x9EF8
GV	1.000	0x7800	1.000	0x7800
BV	1.732	0x7DD8	1.855	0x7ED8
RV	0.000	0x0000	0.000	0x0000

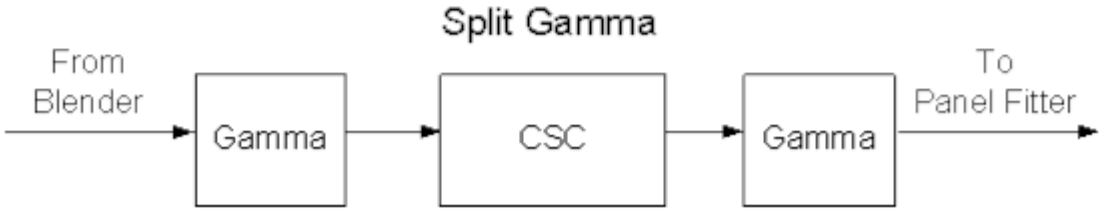
The pipe gamma and color space conversion blocks can be placed in three different arrangements:

- Gamma before CSC, selected through the CSC Mode register. This is mostly used for RGB to YUV conversion.
- Gamma after CSC, selected through the CSC Mode register. This is mostly used for YUV to RGB conversion or linear RGB to RGB conversion. This mode can be used with pipe color gamut enhancement.
- Split gamma, selected through the Pipe Config register. This is mostly used for RGB to RGB conversion. This mode can be used with pipe color gamut enhancement. In this mode, the pipe gamma enable per plane will control whether a plane will go through both gamma blocks. It is not possible to send a plane through one gamma block and not the other.

In either arrangement, the final output of the pipe gamma and CSC and gamut enhancement logic is clamped to fit in the 0 to 1.0 range before going to the ports.


Gamma Before CSC
From Blender
Gamma
CSC
To Panel Fitter


Gamma After CSC
From Blender
CSC
Gamma
To Panel Fitter


Split Gamma
From Blender
Gamma
CSC
Gamma
To Panel Fitter

3.10.1 CSC_COEFF-CSC Coefficients

CSC COEFFICIENT FORMAT

Project:	
Size (in bits):	16
Default Value:	0x00000000

Coefficients for the CSC are stored in sign-exponent-mantissa format. Two CSC coefficients are stored in each dword, the table below show the data packing in each dword.

DWord	Bit	Description
0	15	**Sign** Value: 0b – Positive – Positive Value: 1b – Negative – Negative
	14:12	**Exponent bits** Represented as 2^(-n) Value: 110b – Name: 4 – Description: 4 or mantissa is bb.bbbbbbb Value: 111b – Name: 2 – Description: 2 or mantissa is b.bbbbbbbb Value: 000b – Name: 1 – Description: 1 or mantissa is 0.bbbbbbbbb Value: 001b – Name: 0.5 – Description: 0.5 or mantissa is 0.0bbbbbbbbb Value: 010b – Name: 0.25 – Description: 0.25 or mantissa is 0.00bbbbbbbbb Value: 011b – Name: 0.125 – Description: 0.125 or mantissa is 0.000bbbbbbbbb Value: Others – Name: Reserved – Description: Reserved
	11:3	**Mantissa**
	2:0	**Reserved**

CSC_COEFF

Register Space:	MMIO: 0/2/0
Project:	
Default Value:	0x00000000, 0x00000000, 0x00000000, 0x00000000, 0x00000000, 0x00000000
Access:	R/W
Size (in bits):	6x32
Double Buffer Update Point:	Start of vertical blank after armed
Double Buffer Armed By:	Write to CSC_MODE
Address:	49010h-49027h
Name:	CSC A Coefficients
ShortName:	CSC_COEFF_[1-6]_A
Address:	49110h-49127h
Name:	CSC B Coefficients
ShortName:	CSC_COEFF_[1-6]_B
Address:	49210h-49227h
Name:	CSC C Coefficients

CSC_COEFF		
ShortName:	CSC_COEFF_[1-6]_C	
DWord	**Bit**	**Description**
0	31:16	**RY** Format: CSC COEFFICIENT FORMAT
	15:0	**GY** Format: CSC COEFFICIENT FORMAT
1	31:16	**BY** Format: CSC COEFFICIENT FORMAT
	15:0	**Reserved** Format: MBZ
2	31:16	**RU** Format: CSC COEFFICIENT FORMAT
	15:0	**GU** Format: CSC COEFFICIENT FORMAT
3	31:16	**BU** Format: CSC COEFFICIENT FORMAT
	15:0	**Reserved** Format: MBZ
4	31:16	**RV** Format: CSC COEFFICIENT FORMAT
	15:0	**GV** Format: CSC COEFFICIENT FORMAT
5	31:16	**BV** Format: CSC COEFFICIENT FORMAT
	15:0	**Reserved** Format: MBZ

3.10.2 CSC_MODE-CSC Mode

CSC_MODE	
Register Space:	MMIO: 0/2/0
Project:	
Default Value:	0x00000000
Access:	R/W
Size (in bits):	32
Double Buffer Update Point:	Start of vertical blank
Address:	49028h-4902Bh
Name:	CSC A Mode
ShortName:	CSC_MODE_A
Address:	49128h-4912Bh
Name:	CSC B Mode
ShortName:	CSC_MODE_B

CSC_MODE

Address:	49228h-4922Bh
Name:	CSC C Mode
ShortName:	CSC_MODE_C

Writes to this register arm CSC registers for this pipe

DWord	Bit	Description
0	31:2	**Reserved**
	1	**CSC Position** Selects the CSC position in the pipe. This is ignored when split gamma mode is selected in the pipe config register.

Value	Name	Description
0b	CSC After **[Default]**	CSC is after gamma
1b	CSC Before	CSC is before gamma

	0	**Reserved**

Format:	MBZ

3.10.3 CSC_PREOFF-CSC Pre-Offsets

CSC_PREOFF

Register Space:	MMIO: 0/2/0
Project:	
Default Value:	0x00000000, 0x00000000, 0x00000000
Access:	R/W
Size (in bits):	3x32
Double Buffer Update Point:	Start of vertical blank after armed
Double Buffer Armed By:	Write to CSC_MODE

Address:	49030h-4903Bh
Name:	CSC A Pre-Offsets
ShortName:	CSC_PREOFF_[1-3]_A

Address:	49130h-4913Bh
Name:	CSC B Pre-Offsets
ShortName:	CSC_PREOFF_[1-3]_B

Address:	49230h-4923Bh
Name:	CSC C Pre-Offsets
ShortName:	CSC_PREOFF_[1-3]_C

The pre-offset is intended to remove an offset from 0 on the Y or RGB channels and to convert UV channels from excess 0.5 to 2's complement as they enter pipe color space conversion (CSC).

DWord	Bit	Description
0	31:13	**Reserved**

Format:	MBZ

	12:0	**PreCSC High Offset** This 2's complement value is used to give an offset to the high color channel as it enters CSC logic.

CSC_PREOFF		
		The value is a 2's complement fraction allowing offsets between -1 and +1 (exclusive).
1	31:13	**Reserved**
		Format: MBZ
	12:0	**PreCSC Medium Offset** This 2's complement value is used to give an offset to the medium color channel as it enters CSC logic. The value is a 2's complement fraction allowing offsets between -1 and +1 (exclusive).
2	31:13	**Reserved**
		Format: MBZ
	12:0	**PreCSC Low Offset** This 2's complement value is used to give an offset to the low color channel as it enters CSC logic. The value is a 2's complement fraction allowing offsets between -1 and +1 (exclusive).

3.10.4 CSC_POSTOFF-CSC Post-Offsets

CSC_POSTOFF	
Register Space:	MMIO: 0/2/0
Project:	
Default Value:	0x00000000, 0x00000000, 0x00000000
Access:	R/W
Size (in bits):	3x32
Double Buffer Update Point:	Start of vertical blank after armed
Double Buffer Armed By:	Write to CSC_MODE
Address:	49040h-4904Bh
Name:	CSC A Post-Offsets
ShortName:	CSC_POSTOFF_[1-3]_A
Address:	49140h-4914Bh
Name:	CSC B Post-Offsets
ShortName:	CSC_POSTOFF_[1-3]_B
Address:	49240h-4924Bh
Name:	CSC C Post-Offsets
ShortName:	CSC_POSTOFF_[1-3]_C

The post-offset is intended to add an offset from 0 on the Y or RGB channels and to convert UV channels from 2's complement to excess 0.5 as they exit pipe color space conversion (CSC).

DWord	Bit	Description
0	31:13	**Reserved**
		Format: MBZ
	12:0	**PostCSC High Offset** This 2's complement value is used to give an offset to the high color channel as it exits CSC logic. The value is a 2's complement fraction allowing offsets between -1 and +1 (exclusive).
1	31:13	**Reserved**
		Format: MBZ
	12:0	**PostCSC Medium Offset**

CSC_POSTOFF		
		This 2's complement value is used to give an offset to the medium color channel as it exits CSC logic. The value is a 2's complement fraction allowing offsets between -1 and +1 (exclusive).
2	31:13	**Reserved** Format: MBZ
	12:0	**PostCSC Low Offset** This 2's complement value is used to give an offset to the low color channel as it exits CSC logic. The value is a 2's complement fraction allowing offsets between -1 and +1 (exclusive).

3.11 Pipe Palette and Gamma

The display palette provides a means to correct the gamma of an image stored in a frame buffer to match the gamma of the monitor or presentation device. Additionally, the display palette provide a method for converting indexed data values to color values for VGA and 8-bpp indexed display modes. The display palette is located after the plane blender. Using the individual plane gamma enables, the blended pixels can go through or bypass the palette on a pixel by pixel basis.

The display palette can be accessed through multiple methods and operate in one of four different modes.

8 bit legacy palette/gamma mode:

This provides a palette mode for indexed pixel data formats (VGA and primary plane 8 bpp) and gamma correction for legacy programming requirements.

All input values are clamped to the 0.0 to 1.0 range before the palette/gamma calculation. It is not recommended to use legacy palette mode with extended range formats.

For input values greater than or equal to 0 and less than 1.0, the input value is used to directly lookup the result value from one of the 256 palette/gamma entries. The 256 entries are stored in the legacy palette with 8 bits per color in a 0.8 format with 0 integer and 8 fractional bits.

The legacy palette is programmable through both MMIO and VGA I/O registers. Through VGA I/O, the palette can look as though there are only 6 bits per color component, depending on programming of other VGA I/O registers.

Workaround: Set MMIO GTTMMADDR offsets 0x70064, 0x71064, and 0x72064 bits 6 to 1b prior to accessing the palette through VGA I/O. Clear the bits when done with palette access.

10 bit gamma mode:

This provides the highest quality gamma for pixel data formats of 30 bits per pixel or less.

All input values are clamped to the greater than -3.0 and less than 3.0 range before the gamma calculation.

For input values greater than or equal to 0 and less than 1.0, the input value is used to directly lookup the result value from one of the first 1024 gamma entries. The first 1024 entries are stored in the precision palette with 10 bits per color in a 0.10 format with 0 integer and 10 fractional bits.

For input values greater than or equal to 1.0 and less than 3.0, the input value is used to linearly interpolate between the 1024th and 1025th gamma entries to create the result value. The 1025th entry is stored in the PAL_EXT_GC_MAX register with 19 bits per color in a 3.16 format with 3

integer and 16 fractional bits (maximum value <4.0 when pipe CSC is enabled and after pipe gamma, <8.0 otherwise).

For negative input values, gamma is mirrored along the X-axis, giving the same result as positive input values, except for a negative sign. When gamma input may be negative, the first gamma point should be programmed to a value of 0.0 in order to have a symmetric mirroring.

Split gamma mode:

Split gamma mode is composed of two gamma functions. The first gamma is before pipe color space conversion (CSC) and the second is after CSC. This split gamma mode permits mapping to linear gamma, then color space conversion, then mapping to monitor gamma. This provides the highest quality pipe color space conversion and gamma correction for inputs with non-linear gamma.

First gamma (before CSC):

All input values are clamped to the greater than -3.0 and less than 3.0 range before the gamma calculation.

For input values greater than or equal to 0 and less than 1.0, the input value is used to directly lookup the result value from one of the first 512 gamma entries. The first 512 entries are stored in the precision palette indexes 0 to 511 with 10 bits per color in a 0.10 format with 0 integer and 10 fractional bits.

For input values greater than or equal to 1.0 and less than 3.0, the input value is used to linearly interpolate between the 512th and 513th gamma entries to create the result value. The 513th entry is stored in the PAL_EXT_GC_MAX register with 19 bits per color in a 3.16 format with 3 integer and 16 fractional bits (maximum value <4.0).

For negative input values, gamma is mirrored along the X-axis, giving the same result as positive input values, except for a negative sign. When gamma input may be negative, the first gamma point should be programmed to a value of 0.0 in order to have a symmetric mirroring.

Second gamma (after CSC):

All input values are clamped to the 0.0 to 1.0 range before the gamma calculation.

For input values greater than or equal to 0 and less than 1.0, the input value is used to directly lookup the result value from one of the first 512 gamma entries. The first 512 entries are stored in the precision palette indexes 512 to 1023 with 10 bits per color in a 0.10 format with 0 integer and 10 fractional bits.

12 bit interpolated gamma mode:

This provides the highest quality gamma for pixel data formats greater than 30 bits per pixel.

The gamma correction curve is represented by specifying a set of gamma entry reference points spaced equally along the curve for values between -1 and 1. For extended values there is an extended gamma entry reference point at the maximum alowed input value.

All input values are clamped to the greater than -3.0 and less than 3.0 range before the gamma calculation.

For input values greater than or equal to 0 and less than 1.0, the input value is used to linearly interpolate between two adjacent points of the first 513 gamma entries to create the result value.

The first 512 entries are stored in the precision palette with 16 bits per color in a 0.16 format with 0 integer and 16 fractional bits (upper 10 bits in odd indexes, lower 6 bits in even indexes). The 513th entry is stored in the PAL_GC_MAX register with 17 bits per color in a 1.16 format with 1 integer and 16 fractional bits.

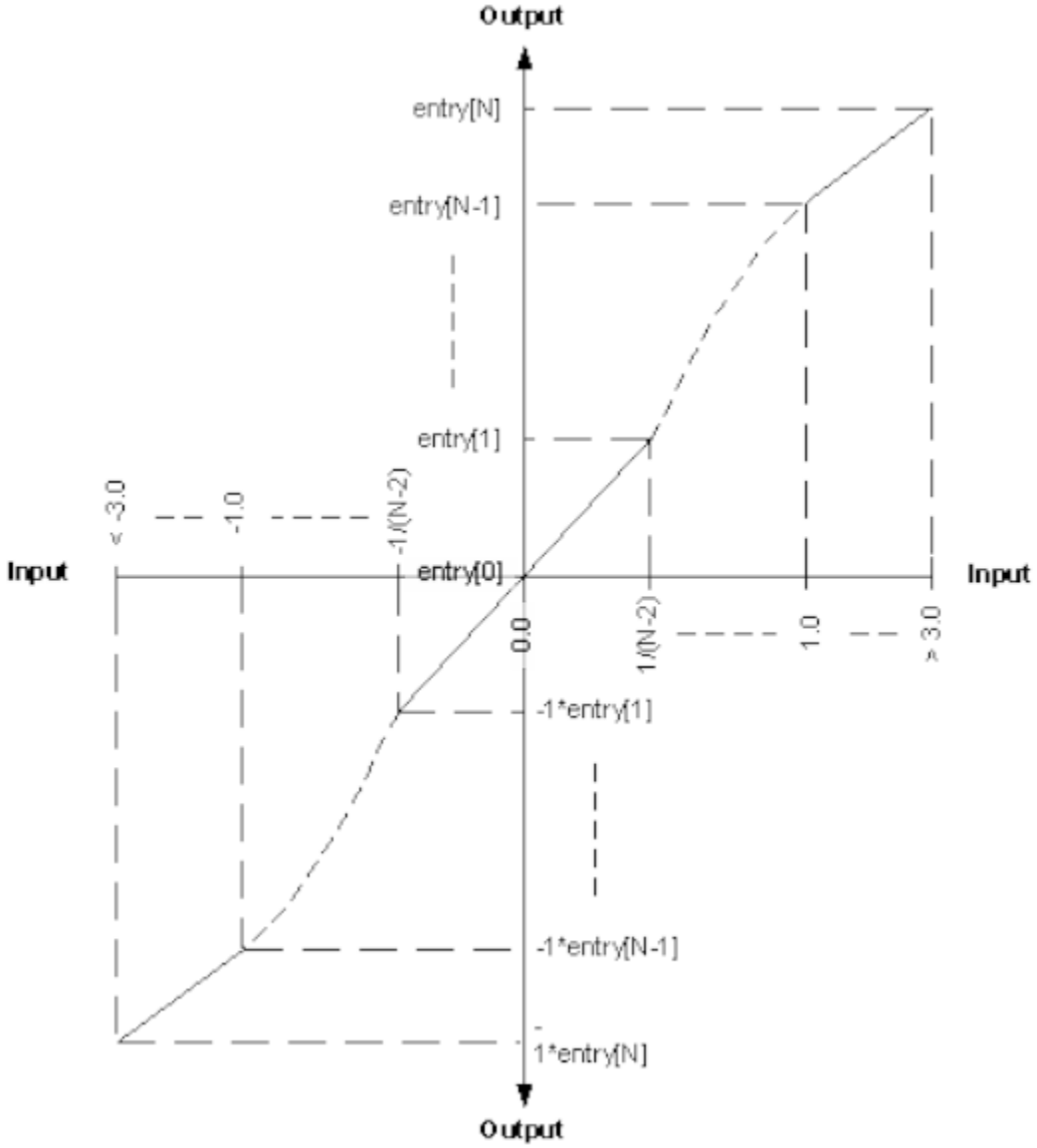
For input values greater than or equal to 1.0 and less than 3.0, the input value is used to linearly interpolate between the 513th and 514th gamma entries to create the result value. The 514th entry is stored in the PAL_EXT_GC_MAX register with 19 bits per color in a 3.16 format with 3 integer and 16 fractional bits (maximum value <4.0 when pipe CSC is enabled and after pipe gamma, <8.0 otherwise).

For negative input values, gamma is mirrored along the X-axis, giving the same result as positive input values, except for a negative sign. When gamma input may be negative, the first gamma point should be programmed to a value of 0.0 in order to have a symmetric mirroring.

To program the gamma correction entries, calculate the desired gamma curve for inputs from 0 to 3.0. The curve must be flat or increasing, never decreasing. For inputs of 0 to 1.0, multiply the input value by 512 to find the gamma entry number, then store the desired gamma result in that entry. For inputs greater than 1.0 and less than or equal to 3.0, store the result for an input of 3.0 in the 514th gamma entry.

Note: If any gamma value to be programmed exceeds the maximum allowable value in the associated gamma register, then the programmed value must be clamped to the maximum allowable value.

Example Pipe Gamma Correction Curve



3.11.1 PAL_LGC-Legacy Palette

PAL_LGC	
Register Space:	MMIO: 0/2/0
Project:	
Default Value:	0x00000000, 0x00000000
Access:	R/W (DWORD access only, no byte access)
Address:	4A000h-4A3FFh
Name:	Legacy Palette A
ShortName:	PAL_LGC_[0-255]_A
Address:	4A800h-4ABFFh
Name:	Legacy Palette B

PAL_LGC		
ShortName:	PAL_LGC_[0-255]_B	
Address:	4B000h-4B3FFh	
Name:	Legacy Palette C	
ShortName:	PAL_LGC_[0-255]_C	
DWord	**Bit**	**Description**
0..255	31:24	**Reserved** Format: MBZ
	23:16	**Red Legacy Palette Entry** Default Value: UUh Red legacy palette entry value.
	15:8	**Green Legacy Palette Entry** Default Value: UUh Green legacy palette entry value.
	7:0	**Blue Legacy Palette Entry** Default Value: UUh Blue legacy palette entry value.

3.11.2 PAL_PREC_INDEX-Precision Palette Index

PAL_PREC_INDEX		
Register Space:		MMIO: 0/2/0
Project:		
Default Value:		0x00000000
Access:		R/W
Size (in bits):		32
Address:	4A400h-4A403h	
Name:	Precision Palette A Index	
ShortName:	PAL_PREC_INDEX_A	
Address:	4AC00h-4AC03h	
Name:	Precision Palette B Index	
ShortName:	PAL_PREC_INDEX_B	
Address:	4B400h-4B403h	
Name:	Precision Palette C Index	
ShortName:	PAL_PREC_INDEX_C	
This index controls access to the array of precision palette data values.		

DWord	Bit	Description
0	31	**Precision Palette Format** This field selects the format of the precision palette data. It must be set when reading or writing precision palette entries for split gamma mode. It must be cleared before programming the legacy

PAL_PREC_INDEX		
		palette. **Value** / **Name** / **Description** 0b / Non-split **[Default]** / 10 bpc or 12 bpc gamma format 1b / Split / Split gamma format
	30:16	**Reserved** Format: MBZ
	15	**Index Auto Increment** This field enables the index auto increment. **Value** / **Name** / **Description** 0b / No Increment **[Default]** / Do not automatically increment the index value. 1b / Auto Increment / Increment the index value with each read or write to the data register. **Programming Notes** Restriction : Index auto increment mode should not be used.
	14:10	**Reserved** Format: MBZ
	9:0	**Index Value** This field indicates the data location to be accessed through the data register. This value can be automatically incremented by a read or a write to the data register if the index auto increment bit is set. When automatically incrementing, the current automatically calculated index value can be read here. When automatically incrementing, the index will roll over to 0 after reaching the end of the allowed range. **Value** / **Name** [0,1023] /

3.11.3 PAL_PREC_DATA-Precision Palette Data

PAL_PREC_DATA	
Register Space:	MMIO: 0/2/0
Project:	
Default Value:	0x00000000
Access:	R/W (DWORD access only, no byte access)
Size (in bits):	32
Address:	4A404h-4A407h
Name:	Precision Palette A Data
ShortName:	PAL_PREC_DATA_A
Address:	4AC04h-4AC07h
Name:	Precision Palette B Data
ShortName:	PAL_PREC_DATA_B
Address:	4B404h-4B407h
Name:	Precision Palette C Data
ShortName:	PAL_PREC_DATA_C

PAL_PREC_DATA

These are the precision palette entries used for the 10 bpc, split, and 12 bpc gamma. The Precision Palette Index Value indicates the precision palette location to be accessed through this register

DWord	Bit	Description
0	31:30	**Reserved**
	29:20	**Red Precision Palette Entry** Default Value: UUUUUUUUUUb For 10 bpc, program with the red 10 bit palette entry fraction value. For 12 bpc gamma odd indexes, program with the upper 10 bits of the red palette entry fraction value. For 12 bpc gamma even indexes, program the MSbs with the lower 6 bits of the red palette entry fraction value, then program all 0s in the LSbs. For split gamma indexes 0 to 511, program with the first gamma (before CSC) red 10 bit palette entry fraction value. For split gamma indexes 512 to 1023, program with the second gamma (after CSC) red 10 bit palette entry fraction value.
	19:10	**Green Precision Palette Entry** Default Value: UUUUUUUUUUb For 10 bpc, program with the green 10 bit palette entry fraction value. For 12 bpc gamma odd indexes, program with the upper 10 bits of the green palette entry fraction value. For 12 bpc gamma even indexes, program the MSbs with the lower 6 bits of the green palette entry fraction value, then program all 0s in the LSbs. For split gamma indexes 0 to 511, program with the first gamma (before CSC) green 10 bit palette entry fraction value. For split gamma indexes 512 to 1023, program with the second gamma (after CSC) green 10 bit palette entry fraction value.
	9:0	**Blue Precision Palette Entry** Default Value: UUUUUUUUUUb For 10 bpc, program with the blue 10 bit palette entry fraction value. For 12 bpc gamma odd indexes, program with the upper 10 bits of the blue palette entry fraction value. For 12 bpc gamma even indexes, program the MSbs with the lower 6 bits of the blue palette entry fraction value, then program all 0s in the LSbs. For split gamma indexes 0 to 511, program with the first gamma (before CSC) blue 10 bit palette entry fraction value. For split gamma indexes 512 to 1023, program with the second gamma (after CSC) blue 10 bit palette entry fraction value.

3.11.4 PAL_GC_MAX-Gamma Correction Max

PAL_GC_MAX	
Register Space:	MMIO: 0/2/0
Project:	
Default Value:	0x00010000, 0x00010000, 0x00010000
Access:	R/W
Size (in bits):	3x32
Address:	4A410h-4A41Bh
Name:	Gamma Correction A Max
ShortName:	PAL_GC_MAX_[1-3]_A
Address:	4AC10h-4AC1Bh
Name:	Gamma Correction B Max
ShortName:	PAL_GC_MAX_[1-3]_B
Address:	4B410h-4B41Bh
Name:	Gamma Correction C Max
ShortName:	PAL_GC_MAX_[1-3]_C

DWord	Bit	Description
0	31:17	**Reserved** Format: MBZ
	16:0	**Red Max GC Point** Default Value: 10000000000000000b Format: U1.16 The 513th entry for the red color channel of the 12 bit interpolated gamma correction. This value is represented in a 1.16 format with 1 integer and 16 fractional bits. **Programming Notes** Restriction : The value should always be programmed to be less than or equal to 1.0.
1	31:17	**Reserved** Format: MBZ
	16:0	**Green Max GC Point** Default Value: 10000000000000000b Format: U1.16 The 513th entry for the green color channel of the 12 bit interpolated gamma correction. This value is represented in a 1.16 format with 1 integer and 16 fractional bits. **Programming Notes** Restriction : The value should always be programmed to be less than or equal to 1.0.
2	31:17	**Reserved** Format: MBZ
	16:0	**Blue Max GC Point** Default Value: 10000000000000000b Format: U1.16 The 513th entry for the blue color channel of the 12 bit interpolated gamma correction. This value is represented in a 1.16 format with 1 integer and 16 fractional bits. **Programming Notes** Restriction : The value should always be programmed to be less than or equal to 1.0.

3.11.5 PAL_EXT_GC_MAX-Extended Gamma Correction Max

PAL_EXT_GC_MAX

Register Space:	MMIO: 0/2/0
Project:	
Default Value:	0x0007FFFF, 0x0007FFFF, 0x0007FFFF
Access:	R/W
Size (in bits):	3x32
Address:	4A420h-4A42Bh
Name:	Extended Gamma Correction A Max
ShortName:	PAL_EXT_GC_MAX_[1-3]_A
Address:	4AC20h-4AC2Bh
Name:	Extended Gamma Correction B Max
ShortName:	PAL_EXT_GC_MAX_[1-3]_B
Address:	4B420h-4B42Bh
Name:	Extended Gamma Correction C Max
ShortName:	PAL_EXT_GC_MAX_[1-3]_C

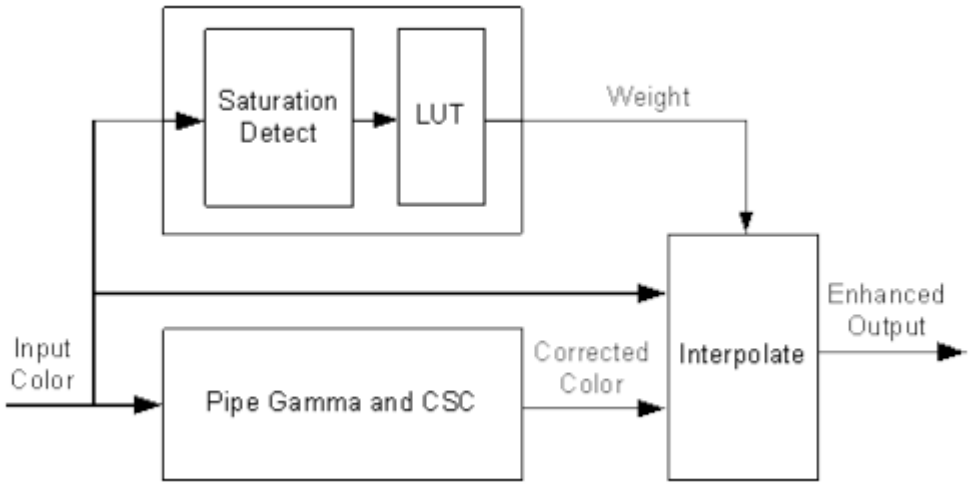
DWord	Bit	Description
0	31:19	**Reserved** Format: MBZ
	18:0	**Red Ext Max GC Point** Default Value: 11111111111111111b Format: U3.16 The extended point for red color channel gamma correction. This value is represented in a 3.16 format with 3 integer and 16 fractional bits. **Programming Notes** Restriction : The value should always be programmed to be less than 8.0. The value must be programmed to be less than 4.0 when pipe color space conversion is enabled and pipe gamma is placed before pipe color space conversion or split gamma is used.
1	31:19	**Reserved** Format: MBZ
	18:0	**Green Ext Max GC Point** Default Value: 11111111111111111b Format: U3.16 The extended point for green color channel gamma correction. This value is represented in a 3.16 format with 3 integer and 16 fractional bits. **Programming Notes** Restriction : The value should always be programmed to be less than 8.0. The value must be programmed to be less than 4.0 when pipe color space conversion is enabled and pipe gamma is placed before pipe color space conversion or split gamma is used.
2	31:19	**Reserved** Format: MBZ
	18:0	**Blue Ext Max GC Point** Default Value: 11111111111111111b Format: U3.16 The extended point for blue color channel gamma correction. This value is represented in a 3.16 format with 3 integer and 16 fractional bits.

PAL_EXT_GC_MAX	
	Programming Notes
	Restriction : The value should always be programmed to be less than 8.0. The value must be programmed to be less than 4.0 when pipe color space conversion is enabled and pipe gamma is placed before pipe color space conversion or split gamma is used.

3.12 Pipe Color Gamut Enhancement

Pipe color gamut enhancement is used to enhance display of standard gamut content on wide gamut displays. It processes the color value from before and after the pipe gamma and color space correction blocks to create the color gamut enhanced output. The typical usage is to output the pipe gamma and CSC corrected color for areas of low saturated content and the input (not gamma or CSC corrected) color for areas of high saturated content. It is not recommended to use color gamut enhancement with wide gamut inputs.

The pipe Gamma and CSC must be programmed to either the split gamma mode or gamma after CSC mode when using pipe color gamut enhancement.



The saturation level of the pipe gamma and CSC input color is detected and used to index into a look up table (LUT) containing programmable weights. The saturation values are linearly distributed across the LUT indexes from the lowest index for lowest saturation to the highest index for highest saturation.

The enhanced output color is created by using the weight value to interpolate between the input color and corrected color. See the following table of weights to amount of input or corrected color used to create the enhanced output color.

Weighting of input and corrected colors

Weight from LUT	Amount of Input Color in Enhanced Output	Amount of Corrected Color in Enhanced Output
00 0000b (minimum)	0%	100%
...	...	...
00 1000b	25%	75%
...	...	...
01 0000b	50%	50%

Weight from LUT	Amount of Input Color in Enhanced Output	Amount of Corrected Color in Enhanced Output
...	...	...
01 1000b	75%	25%
...	...	...
10 0000b (maximum)	100%	0%

Example weight programming

CGE LUT Index	CGE Weight Value Decimal	CGE Weight Value Binary	CGE Weight Percent Input Color	CGE Weight Percent Corrected Color
0 (lowest saturation)	0	00 0000b	0%	100%
1	0	00 0000b	0%	100%
2	0	00 0000b	0%	100%
3	0	00 0000b	0%	100%
4	0	00 0000b	0%	100%
5	0	00 0000b	0%	100%
6	1.6	00 0010b	5%	95%
7	3.2	00 0011b	10%	90%
8	4.8	00 0101b	15%	85%
9	6.4	00 0110b	20%	80%
10	8.64	00 1001b	27%	73%
11	12.8	00 1101b	40%	60%
12	19.2	01 0011b	60%	40%
13	25.6	01 1010b	80%	20%
14	28.8	01 1101b	90%	10%
15	32	10 0000b	100%	0%
16 (highest saturation)	32	10 0000b	100%	0%

3.12.1 CGE_CTRL-Color Gamut Enhancement Control

CGE_CTRL		
Register Space:		MMIO: 0/2/0
Project:		
Default Value:		0x00000000
Access:		R/W
Size (in bits):		32
Double Buffer Update Point:		Start of vertical blank
Address:	49080h-49083h	
Name:	Color Gamut Enhancement A Control	
ShortName:	CGE_CTRL_A	
Address:	49180h-49183h	
Name:	Color Gamut Enhancement B Control	
ShortName:	CGE_CTRL_B	
Address:	49280h-49283h	
Name:	Color Gamut Enhancement C Control	
ShortName:	CGE_CTRL_C	

DWord	Bit	Description
0	31	**CGE Enable** This bit enables the Color Gamut Enhancement logic.

Value	Name	Description
0b	Disable **[Default]**	Disable CGE
1b	Enable	Enable CGE

DWord	Bit	Description
0	30:0	**Reserved**

Format:	MBZ

3.12.2 CGE_WEIGHT-Color Gamut Enhancement Weight

CGE_WEIGHT	
Register Space:	MMIO: 0/2/0
Project:	
Default Value:	0x00000000, 0x00000000, 0x00000000, 0x00000000, 0x00000000
Access:	R/W
Size (in bits):	4x32
Address:	49090h-490A3h
Name:	Color Gamut Enhancement A Weights
ShortName:	CGE_WEIGHT_[0-4]_A
Address:	49190h-491A3h
Name:	Color Gamut Enhancement B Weights
ShortName:	CGE_WEIGHT_[0-4]_B
Address:	49290h-492A3h
Name:	Color Gamut Enhancement C Weights
ShortName:	CGE_WEIGHT_[0-4]_C

These are the weights contained in the lookup up table (LUT) used in pipe color gamut enhancement.
LUT index 0 contains the weight for the least saturated colors, and LUT index 16 contains the weight for the most saturated colors.
Weight values can range from 00000b (100% of the enhanced output color is from the pipe gamma and CSC output corrected color) to 100000b (100% of the enhanced output color is from the pipe gamma and CSC input color).

Programming Notes

Restriction : The weight values should only be changed while color gamut enhancement is disabled, otherwise screen artifacts may show temporarily.

DWord	Bit	Description
0	31:30	**Reserved** Format: MBZ
	29:24	**CGE Weight Index 3** This is the weight value for this color gamut enhancement LUT index.
	23:22	**Reserved** Format: MBZ
	21:16	**CGE Weight Index 2** This is the weight value for this color gamut enhancement LUT index.
	15:14	**Reserved** Format: MBZ
	13:8	**CGE Weight Index 1** This is the weight value for this color gamut enhancement LUT index.
	7:6	**Reserved** Format: MBZ
	5:0	**CGE Weight Index 0** This is the weight value for this color gamut enhancement LUT index.
1	31:30	**Reserved** Format: MBZ
	29:24	**CGE Weight Index 7**

CGE_WEIGHT		
		This is the weight value for this color gamut enhancement LUT index.
	23:22	**Reserved** Format: MBZ
	21:16	**CGE Weight Index 6** This is the weight value for this color gamut enhancement LUT index.
	15:14	**Reserved** Format: MBZ
	13:8	**CGE Weight Index 5** This is the weight value for this color gamut enhancement LUT index.
	7:6	**Reserved** Format: MBZ
	5:0	**CGE Weight Index 4** This is the weight value for this color gamut enhancement LUT index.
2	31:30	**Reserved** Format: MBZ
	29:24	**CGE Weight Index 11** This is the weight value for this color gamut enhancement LUT index.
	23:22	**Reserved** Format: MBZ
	21:16	**CGE Weight Index 10** This is the weight value for this color gamut enhancement LUT index.
	15:14	**Reserved** Format: MBZ
	13:8	**CGE Weight Index 9** This is the weight value for this color gamut enhancement LUT index.
	7:6	**Reserved** Format: MBZ
	5:0	**CGE Weight Index 8** This is the weight value for this color gamut enhancement LUT index.
3	31:30	**Reserved** Format: MBZ
	29:24	**CGE Weight Index 15** This is the weight value for this color gamut enhancement LUT index.
	23:22	**Reserved** Format: MBZ
	21:16	**CGE Weight Index 14** This is the weight value for this color gamut enhancement LUT index.
	15:14	**Reserved** Format: MBZ
	13:8	**CGE Weight Index 13** This is the weight value for this color gamut enhancement LUT index.
	7:6	**Reserved** Format: MBZ
	5:0	**CGE Weight Index 12** This is the weight value for this color gamut enhancement LUT index.
4	31:6	**Reserved**

CGE_WEIGHT			
		Format:	MBZ
	5:0	**CGE Weight Index 16** This is the weight value for this color gamut enhancement LUT index.	

3.13 Software Flags

SWF	
Register Space:	MMIO: 0/2/0
Project:	
Default Value:	0x00000000, 0x00000000
Access:	R/W
Size (in bits):	36x32
Address:	4F000h-4F08Fh
Name:	Software Flags
ShortName:	SWF_[0-35]

These registers are used as scratch pad data storage space and have no direct effect on hardware operation. The use of these registers is defined by the software architecture.

DWord	Bit	Description
0..35	31:0	**Software Flags**

3.14 GTSCRATCH-GT Scratchpad

GTSCRATCH	
Register Space:	MMIO: 0/2/0
Project:	
Default Value:	0x00000000, 0x00000000, 0x00000000, 0x00000000, 0x00000000, 0x00000000, 0x00000000, 0x00000000
Access:	R/W
Size (in bits):	8x32
Address:	4F100h-4F11Fh
Name:	GT Scratchpad

GTSCRATCH		
ShortName:	GTSCRATCH_[0-7]	
These registers are used by hardware and must not be used by software.		
DWord	**Bit**	**Description**
0..7	31:0	**GT Scratchpad**

4. North Display Engine Pipe and Port Controls

4.1 Pipe Timing

4.1.1 HTOTAL-Horizontal Total

HTOTAL		
Register Space:		MMIO: 0/2/0
Project:		
Default Value:		0x00000000
Access:		R/W
Size (in bits):		32
Address:	60000h-60003h	
Name:	Pipe A Horizontal Total	
ShortName:	PIPE_HTOTAL_A	
Address:	61000h-61003h	
Name:	Pipe B Horizontal Total	
ShortName:	PIPE_HTOTAL_B	
Address:	62000h-62003h	
Name:	Pipe C Horizontal Total	
ShortName:	PIPE_HTOTAL_C	

DWord	Bit	Description
0	31:29	**Reserved** Format: MBZ
	28:16	**Horizontal Total** This field specifies Horizontal Total size. This should be equal to the sum of the horizontal active and the horizontal blank sizes. This field is programmed to the number of pixels desired minus one. **Programming Notes** Restriction : The number of pixels (before the minus one) needs to be a multiple of two when driving the LVDS port in two channel mode. Restriction : This register must always be programmed to the same value as the Horizontal Blank End.
	15:12	**Reserved** Format: MBZ
	11:0	**Horizontal Active** This field specifies Horizontal Active Display size. The first horizontal active display pixel is considered pixel number 0. This field is programmed to the number of pixels desired minus one. **Programming Notes** Restriction : The number of pixels (before the minus one) needs to be a multiple of two when driving

HTOTAL		
		the LVDS port in two channel mode. The minimum horizontal active display size is 64 pixels. This register must always be programmed to the same value as the Horizontal Blank Start.

4.1.2 HBLANK-Horizontal Blank

HBLANK		
Register Space:		MMIO: 0/2/0
Project:		
Default Value:		0x00000000
Access:		R/W
Size (in bits):		32
Address:	60004h-60007h	
Name:	Pipe A Horizontal Blank	
ShortName:	PIPE_HBLANK_A	
Address:	61004h-61007h	
Name:	Pipe B Horizontal Blank	
ShortName:	PIPE_HBLANK_B	
Address:	62004h-62007h	
Name:	Pipe C Horizontal Blank	
ShortName:	PIPE_HBLANK_C	

DWord	Bit	Description
0	31:29	**Reserved**
	28:16	**Horizontal Blank End** This field specifies Horizontal Blank End position relative to the horizontal active display start. **Programming Notes** Restriction : The number of pixels within horizontal blank needs to be a multiple of two when driving the LVDS port in two channel mode. The minimum horizontal blank size is 32 pixels. This register must always be programmed to the same value as the Horizontal Total.
	15:13	**Reserved**
	12:0	**Horizontal Blank Start** This field specifies the Horizontal Blank Start position relative to the horizontal active display start. **Programming Notes** Restriction : This register must always be programmed to the same value as the Horizontal Active.

4.1.3 HSYNC-Horizontal Sync

HSYNC		
Register Space:		MMIO: 0/2/0
Project:		
Default Value:		0x00000000
Access:		R/W
Size (in bits):		32
Address:	60008h-6000Bh	
Name:	Pipe A Horizontal Sync	
ShortName:	PIPE_HSYNC_A	
Address:	61008h-6100Bh	
Name:	Pipe B Horizontal Sync	
ShortName:	PIPE_HSYNC_B	
Address:	62008h-6200Bh	
Name:	Pipe C Horizontal Sync	
ShortName:	PIPE_HSYNC_C	

DWord	Bit	Description
0	31:29	**Reserved** Format: MBZ
	28:16	**Horizontal Sync End** This field specifies the Horizontal Sync End position relative to the horizontal active display start. It is programmed with HorizontalActive+FrontPorch+Sync-1. **Programming Notes** Restriction : The number of pixels within horizontal sync needs to be a multiple of two when driving the LVDS port in two channel mode. This value must be greater than the horizontal sync start and less than Horizontal Total. Restriction : HDMI and DVI with audio are not supported when HSYNC Start is programmed equal to HBLANK Start.
	15:13	**Reserved** Format: MBZ
	12:0	**Horizontal Sync Start** This field specifies the Horizontal Sync Start position relative to the horizontal active display start. It is programmed with HorizontalActive+FrontPorch-1. **Programming Notes** Restriction : The number of pixels from active to horizontal sync needs to be a multiple of two when driving the LVDS port in two channel mode. This value must be greater than Horizontal Active.

4.1.4 VTOTAL-Vertical Total

VTOTAL		
Register Space:		MMIO: 0/2/0
Project:		
Default Value:		0x00000000
Access:		R/W
Size (in bits):		32
Address:	6000Ch-6000Fh	
Name:	Pipe A Vertical Total	
ShortName:	PIPE_VTOTAL_A	
Address:	6100Ch-6100Fh	
Name:	Pipe B Vertical Total	
ShortName:	PIPE_VTOTAL_B	
Address:	6200Ch-6200Fh	
Name:	Pipe C Vertical Total	
ShortName:	PIPE_VTOTAL_C	

DWord	Bit	Description
0	31:29	**Reserved**
	28:16	**Vertical Total** This field specifies Vertical Total size. This should be equal to the sum of the vertical active and the vertical blank sizes. For progressive display modes, this field is programmed to the number of lines desired minus one. For interlaced display modes, this field is programmed with the number of lines desired minus two. The vertical counter is incremented on the leading edge of the horizontal sync. For interlaced display, hardware uses this value to calculate the vertical total in each field. Both even and odd vertical totals are supported. **Programming Notes** Restriction : This register must always be programmed to the same value as the Vertical Blank End.
	15:12	**Reserved**
	11:0	**Vertical Active** This field specifies Vertical Active Display size. The first vertical active display line is considered pixel number 0. This field is programmed to the number of lines desired minus one. For interlaced display, hardware uses this value to calculate the vertical active in each field. **Programming Notes** Restriction : When using the internal panel fitting logic, the minimum vertical active area must be seven lines. This register must always be programmed to the same value as the Vertical Blank Start.

4.1.5 VBLANK-Vertical Blank

VBLANK		
Register Space:		MMIO: 0/2/0
Project:		
Default Value:		0x00000000
Access:		R/W
Size (in bits):		32
Address:	60010h-60013h	
Name:	Pipe A Vertical Blank	
ShortName:	PIPE_VBLANK_A	
Address:	61010h-61013h	
Name:	Pipe B Vertical Blank	
ShortName:	PIPE_VBLANK_B	
Address:	62010h-62013h	
Name:	Pipe C Vertical Blank	
ShortName:	PIPE_VBLANK_C	

DWord	Bit	Description
0	31:29	**Reserved**
	28:16	**Vertical Blank End** This field specifies Vertical Blank End position relative to the vertical active display start. For interlaced display, hardware uses this value to calculate the vertical blank end in each field. **Programming Notes** Restriction : This register must always be programmed to the same value as the Vertical Total. The minimum vertical blank size is 5 lines.
	15:13	**Reserved**
	12:0	**Vertical Blank Start** This field specifies the Vertical Blank Start position relative to the vertical active display start. For interlaced display, hardware uses this value to calculate the vertical blank start in each field. **Programming Notes** Restriction : This register must always be programmed to the same value as the Vertical Active

4.1.6 VSYNC-Vertical Sync

VSYNC		
Register Space:		MMIO: 0/2/0
Project:		
Default Value:		0x00000000
Access:		R/W
Size (in bits):		32
Address:	60014h-60017h	
Name:	Pipe A Vertical Sync	
ShortName:	PIPE_VSYNC_A	
Address:	61014h-61017h	
Name:	Pipe B Vertical Sync	
ShortName:	PIPE_VSYNC_B	
Address:	62014h-62017h	
Name:	Pipe C Vertical Sync	
ShortName:	PIPE_VSYNC_C	

DWord	Bit	Description
0	31:29	**Reserved**
	28:16	**Vertical Sync End** This field specifies the Vertical Sync End position relative to the vertical active display start. It is programmed with VerticalActive+FrontPorch+Sync-1. For interlaced display, hardware uses this value to calculate the vertical sync start in each field. **Programming Notes** Restriction : This value must be greater than the vertical sync start and less than Vertical Total.
	15:13	**Reserved**
	12:0	**Vertical Sync Start** This field specifies the Vertical Sync Start position relative to the vertical active display start. It is programmed with VerticalActive+FrontPorch-1. For interlaced display, hardware uses this value to calculate the vertical sync end in each field. **Programming Notes** Restriction : This value must be greater than Vertical Active.

4.1.7 SRCSZ-Source Image Size

SRCSZ		
Register Space:		MMIO: 0/2/0
Project:		
Default Value:		0x00000000
Access:		R/W
Size (in bits):		32
Double Buffer Update Point:		Start of vertical blank
Address:	6001Ch-6001Fh	
Name:	Pipe A Source Image Size	
ShortName:	PIPE_SRCSZ_A	
Address:	6101Ch-6101Fh	
Name:	Pipe B Source Image Size	
ShortName:	PIPE_SRCSZ_B	
Address:	6201Ch-6201Fh	
Name:	Pipe C Source Image Size	
ShortName:	PIPE_SRCSZ_C	

In VGA display mode, this register is ignored and the VGA size from the VGA registers is used instead.

DWord	Bit	Description
0	31:28	**Reserved** Format: MBZ
	27:16	**Horizontal Source Size** This field specifies Horizontal Source Size. This determines the horizontal size of the image created by the display planes. This field is programmed to the number of pixels desired minus one. **Programming Notes** Restriction : This register must always be programmed to the same value as the Horizontal Active, except when panel fitting is enabled.
	15:12	**Reserved** Format: MBZ
	11:0	**Vertical Source Size** This field specifies Vertical Source Size. This determines the vertical size of the image created by the display planes. This field is programmed to the number of lines desired minus one. For interlaced display, hardware divides this number by 2 and adds any necessary half lines to get the vertical blank end for each field. **Programming Notes** Restriction : This register must always be programmed to the same value as the Vertical Active, except when panel fitting is enabled.

4.1.8 VSYNCSHIFT-Vertical Sync Shift

VSYNCSHIFT		
Register Space:		MMIO: 0/2/0
Project:		
Default Value:		0x00000000
Access:		R/W
Size (in bits):		32
Address:	60028h-6002Bh	
Name:	Pipe A Vertical Sync Shift	
ShortName:	PIPE_VSYNCSHIFT_A	
Address:	61028h-6102Bh	
Name:	Pipe B Vertical Sync Shift	
ShortName:	PIPE_VSYNCSHIFT_B	
Address:	62028h-6202Bh	
Name:	Pipe C Vertical Sync Shift	
ShortName:	PIPE_VSYNCSHIFT_C	

DWord	Bit	Description
0	31:13	**Reserved**
	12:0	**Second Field VSync Shift** This value specifies the vertical sync alignment for the start of the interlaced second field, expressed in terms of the absolute pixel number relative to the horizontal active display start. This value will only be used if the transcoder is in an interlaced mode. Typically, the interlaced second field vertical sync should start one pixel after the point halfway between successive horizontal syncs, so the value of this register should be programmed to: horizontal sync start - floor[horizontal total / 2] (use the actual horizontal sync start and horizontal total values and not the minus one values programmed into the registers) This vertical sync shift only occurs during the interlaced second field. In all other cases the vertical sync start position is aligned with horizontal sync start.

4.2 Pipe M/N Values

These values are used for the embedded DisplayPort and FDI.

For dynamic switching between multiple refresh rates, both the M1/N1 data and link values and the M2/N2 link values must be programmed. The PIPE_CONF Refresh Rate Switch setting can be changed on the fly and then alternate M/N values will be used in the next frame that is output.

Calculation of TU, Data M, and Data N is as follows:

For modes that divide into the link frequency evenly,

Active/TU = Payload/Capacity = Data M/N = dot clock * bytes per pixel / ls_clk * number of lanes

Default value to program TU size is "111111" for TU size of 64.

Calculation of Link M and Link N is as follows:

Link M/N = dot clock / ls_clk

Restriction on clocks and number of lanes:

Number of lanes >= INT(dot clock * bytes per pixel / ls_clk)

Pcdclk * number of lanes >= dot clock * bytes per pixel

Please note that in the DisplayPort specifcation, dot clock is referred to as strm_clk.

4.2.1 DATAM-Data M Value

DATAM		
Register Space:		MMIO: 0/2/0
Project:		
Default Value:		0x00000000
Access:		R/W
Size (in bits):		32
Double Buffer Update Point:		Start of vertical blank
Double Buffer Armed By:		Writing the LINKN
Address:	60030h-60033h	
Name:	Pipe A Data M Value 1	
ShortName:	PIPE_DATAM1_A	
Address:	60038h-6003Bh	
Name:	Pipe A Data M Value 2	
ShortName:	PIPE_DATAM2_A	
Address:	61030h-61033h	
Name:	Pipe B Data M Value 1	
ShortName:	PIPE_DATAM1_B	
Address:	61038h-6103Bh	
Name:	Pipe B Data M Value 2	
ShortName:	PIPE_DATAM2_B	
Address:	62030h-62033h	
Name:	Pipe C Data M Value 1	
ShortName:	PIPE_DATAM1_C	
Address:	62038h-6203Bh	
Name:	Pipe C Data M Value 2	
ShortName:	PIPE_DATAM2_C	

DWord	Bit	Description	
0	31	**Reserved**	
		Format:	MBZ

DATAM		
	30:25	**TU Size** This field is the size of the transfer unit, minus one.
	24	**Reserved** Format: MBZ
	23:0	**Data M value** This field is the m value for internal use of the DDA.

4.2.2 DATAN-Data N Value

DATAN		
Register Space:		MMIO: 0/2/0
Project:		
Default Value:		0x00000000
Access:		R/W
Size (in bits):		32
Double Buffer Update Point:		Start of vertical blank
Double Buffer Armed By:		Writing the LINKN
Address:	60034h-60037h	
Name:	Pipe A Data N Value 1	
ShortName:	PIPE_DATAN1_A	
Address:	6003Ch-6003Fh	
Name:	Pipe A Data N Value 2	
ShortName:	PIPE_DATAN2_A	
Address:	61034h-61037h	
Name:	Pipe B Data N Value 1	
ShortName:	PIPE_DATAN1_B	
Address:	6103Ch-6103Fh	
Name:	Pipe B Data N Value 2	
ShortName:	PIPE_DATAN2_B	
Address:	62034h-62037h	
Name:	Pipe C Data N Value 1	
ShortName:	PIPE_DATAN1_C	
Address:	6203Ch-6203Fh	
Name:	Pipe C Data N Value 2	
ShortName:	PIPE_DATAN2_C	

DWord	Bit	Description
0	31:24	**Reserved** Format: MBZ
	23:0	**Data N value**

DATAN		
		This field is the n value for internal use of the DDA.

4.2.3 LINKM-Link M Value

LINKM		
Register Space:		MMIO: 0/2/0
Project:		
Default Value:		0x00000000
Access:		R/W
Size (in bits):		32
Double Buffer Update Point:		Start of vertical blank
Double Buffer Armed By:		Writing the LINKN
Address:	60040h-60043h	
Name:	Pipe A Link M Value 1	
ShortName:	PIPE_LINKM1_A	
Address:	60048h-6004Bh	
Name:	Pipe A Link M Value 2	
ShortName:	PIPE_LINKM2_A	
Address:	61040h-61043h	
Name:	Pipe B Link M Value 1	
ShortName:	PIPE_LINKM1_B	
Address:	61048h-6104Bh	
Name:	Pipe B Link M Value 2	
ShortName:	PIPE_LINKM2_B	
Address:	62040h-62043h	
Name:	Pipe C Link M Value 1	
ShortName:	PIPE_LINKM1_C	
Address:	62048h-6204Bh	
Name:	Pipe C Link M Value 2	
ShortName:	PIPE_LINKM2_C	

DWord	Bit	Description
0	31:24	**Reserved** Format: MBZ
	23:0	**Link M value** This field is the m value for external transmission in the Main Stream Attributes.

4.2.4 LINKN-Link N Value

LINKN		
Register Space:		MMIO: 0/2/0
Project:		
Default Value:		0x00000000
Access:		R/W
Size (in bits):		32
Double Buffer Update Point:		Start of vertical blank
Address:	60044h-60047h	
Name:	Pipe A Link N Value 1	
ShortName:	PIPE_LINKN1_A	
Address:	6004Ch-6004Fh	
Name:	Pipe A Link N Value 2	
ShortName:	PIPE_LINKN2_A	
Address:	61044h-61047h	
Name:	Pipe B Link N Value 1	
ShortName:	PIPE_LINKN1_B	
Address:	6104Ch-6104Fh	
Name:	Pipe B Link N Value 2	
ShortName:	PIPE_LINKN2_B	
Address:	62044h-62047h	
Name:	Pipe C Link N Value 1	
ShortName:	PIPE_LINKN1_C	
Address:	6204Ch-6204Fh	
Name:	Pipe C Link N Value 2	
ShortName:	PIPE_LINKN2_C	

Writes to this register arm M/N registers for this pipe.

DWord	Bit	Description
0	31:24	**Reserved** Format: MBZ
	23:0	**Link N value** This field is the n value for external transmission in the Main Stream Attributes and VB-ID.

4.3 FDI Transmit

4.3.1 FDI_TX_CTL-FDI Tx Control

FDI_TX_CTL

Register Space:		MMIO: 0/2/0
Project:		
Default Value:		0x00040000
Access:		R/W
Size (in bits):		32
Double Buffer Update Point:		Depends on Bit
Address:	60100h-60103h	
Name:	FDI A Tx Control	
ShortName:	FDI_TX_CTL_A	
Address:	61100h-61103h	
Name:	FDI B Tx Control	
ShortName:	FDI_TX_CTL_B	
Address:	62100h-62103h	
Name:	FDI C Tx Control	
ShortName:	FDI_TX_CTL_C	

FDI B maximum port width is 4 lanes when FDI C is disabled, 2 lanes when FDI C is enabled.
FDI C maximum port width is 2 lanes.

DWord	Bit	Description
0	31	**FDI Tx Enable** Disabling this port will put it in its lowest power state. Value: 0b – Disable **[Default]** – Disable the FDI Tx interface Value: 1b – Enable – Enable the FDI Tx interface
	30:28	**Reserved** Format: MBZ
	27:22	**Vswing and Preemp** These bits are used for setting link voltage swing and pre-emphasis The settings require additional programming in the DPAFE_DL<0,1>_IREFCAL registers. DPAFE_DL0_IREFCAL is used for FDI A. DPAFE_DL1_IREFCAL is used for FDI B and FDI C. "Non-adjustable" settings use hard-coded voltage swing values. "Adjustable" settings allow voltage swing values to be optimized with fine tuning through the DPAFE_DL<0,1>_IREFCAL registers. The voltages listed in the description are nominal at 1V supply and recommended board design and default DPAFE_BMUFNC de-emphasis settings. Certain settings require additional programming in the DPAFE_DL0_IREFCAL and DPAFE_DL1_IREFCAL registers. Value – Name – Description

		000000b	Adjustable 500mV+0dB **[Default]**	Adjustable Design: Non-transition 500mV, Transition 500mV Additional programming required: DPAFE_DL<0,1>_IREFCAL[16:2]=00010 10111 10111b Alternate: Non-transition 400mV, Transition 400mV Additional programming required: DPAFE_DL<0,1>_IREFCAL[16:2]=00110 00111 00111b
		111010b	Adjustable 500mV+6dB	Adjustable Design: Non-transition 500mV, Transition 1000mV Additional programming required: DPAFE_DL<0,1>_IREFCAL[16:2]=00010 10111 10111b Alternate: Non-transition 400mV, Transition 800mV Additional programming required: DPAFE_DL<0,1>_IREFCAL[16:2]=00110 00111 00111b
		111001b	666mV+3.5dB	Non-adjustable Design: Non-transition 666mV, Transition 1000mV Additional programming required: DPAFE_DL<0,1>_IREFCAL[16:2]=00010 10111 10111b
		111000b	1000mV+0dB	Non-adjustable Design: Non-transition 1000mV, Transition 1000mV Additional programming required: DPAFE_DL<0,1>_IREFCAL[16:2]=00010 10111 10111b
		Others	Reserved	Reserved
	21:19	**Port Width Selection** This field selects the number of lanes to be enabled on the link. Port width change must be done as a part of mode set. The field is locked once port is enabled and only updates when the port is disabled then re-enabled.		

Value	Name	Description
000b	X1 **[Default]**	x1 Mode
001b	X2	x2 Mode
010b	X3	x3 Mode
011b	X4	x4 Mode

FDI_TX_CTL

	Others	Reserved	Reserved

Programming Notes

Restriction : FDI B and FDI C share lanes. FDI C maximum port width is 2 lanes. FDI B maximum port width is 4 lanes when FDI C is disabled, 2 lanes when FDI C is enabled.

18 **Enhanced Framing Enable**

This bit selects enhanced framing. The field is locked once port is enabled and only updates when the port is disabled then re-enabled.

Value	Name	Description
0b	Disabled	Enhanced framing disabled
1b	Enabled **[Default]**	Enhanced framing enabled

17:15 **Reserved**

Format:	MBZ

14 **FDI PLL enable**

This bit enables the FDI PLL.

This bit is ORed with the PLL enable bit from any other FDI Tx Control registers.

Value	Name	Description
0b	Disable **[Default]**	FDI PLL not enabled through this FDI Tx
1b	Enable	FDI PLL enabled

Programming Notes

Restriction : Wait for the PLL warmup cycle before enabling the port through bit 31 of this register.

13:12 **Reserved**

Format:	MBZ

11 **Composite Sync Select**

This bit selects between composite Sync and separate Fsync/Lsync on this port.

Value	Name	Description
0b	Separate **[Default]**	Separate Fsync/Lsync
1b	Composite	Composite Sync

Programming Notes

Restriction : FDI A can use either separate sync or composite sync. FDI A must use composite sync when FDI C is enabled.

FDI B can use either separate sync or composite sync. FDI B must use composite sync when FDI C is enabled.

FDI C can only use composite sync.

It is recommended to always use composite sync, even when just using FDI A or FDI B, so that FDI C can be enabled later without needing to temporarily disable FDI A and FDI B in order to change them to composite.

10 **Auto Train Enable**

This bit enables auto-training on this port. See the mode set enable sequence for usage. Do not change this bit while the port is enabled.

Value	Name	Description
0b	Disable **[Default]**	Disable auto-training
1b	Enable	Enable auto-training

FDI_TX_CTL

		Programming Notes Workaround : When disabling FDI, clear the FDI Transmitter Auto Train Enable bit in the same write as the FDI Tx enable is cleared, and clear the FDI Receiver Auto Train Enable bit in the same write as the FDI Rx enable is cleared.
	9:8	**Link training pattern enable** These bits are used for manual link initialization. The link must first be configured prior to sending training patterns. Manual link training can not be used when auto-training is enabled.

Value	Name	Description
00b	Pattern 1 **[Default]**	Pattern 1 enabled
01b	Pattern 2	Pattern 2 enabled
10b	Idle	Idle Pattern enabled
11b	Normal	Send normal pixels

Programming Notes
Restriction : When enabling the port, it must be turned on with pattern 1 enabled. When retraining a link, the port must be disabled, then re-enabled with pattern 1 enabled.

	6	**Reserved** Format: MBZ
	4:2	**Reserved** Format: MBZ
	1	**Auto Train Done** Access: RO This bit indicates auto-training completed on this port.

Value	Name	Description
0b	Not Done **[Default]**	Auto-training is not complete or not started
1b	Done	Auto-training is complete

Adjustable Voltage Swing Programming for FDI

Vdiff Single Ended Swing (Transition Bits)	DP_AFE_DL<0,1>_IREFCAL <16:12> TXIRefSel	DP_AFE_DL<0,1>_IREFCAL <11:7> TXVcmSel <6:2> TXVrefSel (Both fields set to same value)
500 mV	2 decimal	23 decimal
400 mV	6 decimal	7 decimal
This is only for use with the FDI_TX_CTL<27:22> Vswing_and_Preemp selections marked as "adjustable". FDI_TX_CTL<27:22> Vswing_and_Preemp must be set to 000000b or 111010b.		

4.4 DisplayPort

4.4.1 DP_CTL-DisplayPort Control

DP_CTL

Register Space:	MMIO: 0/2/0
Project:	
Default Value:	0x00000018
Access:	R/W
Size (in bits):	32
Address:	64000h-64003h
Name:	DisplayPort A Control
ShortName:	DP_CTL_A

DWord	Bit	Description
0	31	**DisplayPort Enable** Disabling this port will put it in its lowest power state. Value / Name / Description: 0b – Disable **[Default]** – Disable the DisplayPort interface 1b – Enable – Enable the DisplayPort interface
	30:29	**Pipe Select** This bit determines from which display pipe the source data will originate. Value / Name / Description: 00b – Pipe A **[Default]** – Pipe A 01b – Pipe B – Pipe B 10b – Pipe C – Pipe C 11b – Reserved – Reserved
	28	**Reserved** Format: MBZ
	27:22	**Vswing Emp Set** These bits are used for setting link voltage swing and emphasis. Certain settings require additional programming in the DPAFE_DP_IREFCAL register. "Non-adjustable" settings use hard-coded voltage swing values. These are the recommended settings for general use. "Adjustable" settings allow voltage swing values to be optimized with fine tuning through the DPAFE_DP_IREFCAL register. The voltages listed are nominal at 1V supply and recommended board design and default DPAFE_BMUFNC de-emphasis settings. 27 = AFE fullSwingMode 26:24 = AFE sel 2:0 23 = AFE dmpen (1 enable de-emphasis, 0 disable de-emphasis)

22 = AFE dmplv (1 set 6dB de-emphasis, 0 set 3.5dB de-emphasis)

Value	Name	Description
100100b	0.4V+0dB	Non-adjustable DP spec equivalent: 0.4V + 0dB emphasis Design: Non-transition 300mV, Transition 300mV Additional programming required: DPAFE_DP_IREFCAL[31]=0b
101010b	0.4V+3.5dB	Non-adjustable DP spec equivalent: 0.4V + 3.5dB emphasis Design: Non-transition 300mV, Transition 450mV Additional programming required: DPAFE_DP_IREFCAL[31]=0b
101111b	0.4V+6dB	Non-adjustable DP spec equivalent: 0.4V + 6dB emphasis Design: Non-transition 300mV, Transition 600mV Additional programming required: DPAFE_DP_IREFCAL[31]=0b
110000b	0.6V+0dB	Non-adjustable DP spec equivalent: 0.6V + 0dB emphasis Design: Non-transition 450mV, Transition 450mV Additional programming required: DPAFE_DP_IREFCAL[31]=0b
110110b	0.6V+3.5dB	Non-adjustable DP spec equivalent: 0.6V + 3.5dB emphasis Design: Non-transition 450mV, Transition 675mV Additional programming required: DPAFE_DP_IREFCAL[31]=0b
111000b	0.8V+0dB	Non-adjustable

DP_CTL			
			DP spec equivalent: 0.8V + 0dB emphasis Design: Non-transition 600mV, Transition 600mV Additional programming required: DPAFE_DP_IREFCAL[31]=0b
		111110b	0.8V+3.5dB	Non-adjustable DP spec equivalent: 0.8V + 3.5dB emphasis Design: Non-transition 600mV, Transition 900mV Additional programming required: DPAFE_DP_IREFCAL[31]=0b
		000000b	0.5V+0dB **[Default]**	Non-adjustable Design: Non-transition 500mV, Transition 500mV Additional programming required: DPAFE_DP_IREFCAL[31]=1b DPAFE_DP_IREFCAL[16:2]=00010 11101 11101b
		000010b	0.5V-3.5dB	Non-adjustable Design: Non-transition 500mV, Transition 333mV Additional programming required: DPAFE_DP_IREFCAL[31]=1b DPAFE_DP_IREFCAL[16:2]=00010 11101 11101b
		100000b	Adjustable MaxV+0dB	Adjustable: Non-transition maxV, Transition maxV Additional programming required: DPAFE_DP_IREFCAL[31]=1b DPAFE_DP_IREFCAL[16:2]=Adjustable
		100010b	Adjustable MaxV-3.5dB	Adjustable: Non-transition maxV, Transition maxV - 3.5dB Additional programming required: DPAFE_DP_IREFCAL[31]=1b DPAFE_DP_IREFCAL[16:2]=Adjustable
		100011b	Adjustable MaxV-6dB	Adjustable: Non-transition maxV, Transition maxV - 6dB

DP_CTL

			Additional programming required: DPAFE_DP_IREFCAL[31]=1b DPAFE_DP_IREFCAL[16:2]=Adjustable
	Others	Reserved	Reserved

21:19 Port Width Selection

This field selects the number of lanes to be enabled on the DisplayPort link.
Port width change must be done as a part of mode set.
The field is locked once port is enabled and only updates when the port is disabled then re-enabled.

Value	Name	Description
000b	x1	x1 Mode
001b	x2	x2 Mode
011b	x4	x4 Mode
Others	Reserved	Reserved

18 Enhanced Framing Enable

This bit selects enhanced framing. The field is locked once port is enabled and only updates when the port is disabled then re-enabled.

Value	Name	Description
0b	Disable **[Default]**	Enhanced framing disabled
1b	Enable	Enhanced framing enabled.

17:16 DP PLL Frequency

This bit selects the DisplayPort PLL frequency.

Value	Name	Description
00b	270mhz	270mhz
01b	162mhz	162mhz.
Others	Reserved	Reserved

15 Port reversal

This bit enables lane reversal within the port: lane 0 mapped to lane 3, lane 1 mapped to lane 2.
Port reversal does not affect AUX channel lane mapping.
The field is locked once port is enabled and only updates when the port is disabled then re-enabled.

Value	Name	Description
0b	Not reversed **[Default]**	Port not reversed
1b	Reversed	Port reversed

14 DisplayPort PLL enable

This bit enables the DisplayPort PLL.

Value	Name	Description
0b	Disable **[Default]**	PLL not enabled
1b	Enable	PLL enabled

Programming Notes
Restriction : Wait for the PLL warmup cycle before enabling the port through bit 31 of this register.

13:10 Reserved

Format:	MBZ

9:8 Link training pattern enable

These bits are used for link initialization as defined in the DisplayPort specification.
The link must first be configured prior to sending training patterns.

DP_CTL

Value	Name	Description
00b	Pattern 1 **[Default]**	Pattern 1 enabled
01b	Pattern 2	Pattern 2 enabled
10b	Idle	Idle Pattern enabled
11b	Normal	Send normal pixels

Programming Notes

Restriction : When enabling the port, it must be turned on with pattern 1 enabled.
When retraining a link, the port must be disabled, then re-enabled with pattern 1 enabled.

6 **Alternate SR Enable**

This bit enables the embedded DisplayPort Alternate Scrambler Reset.

Value	Name	Description
0b	Disable **[Default]**	Alternate SR disabled
1b	Enable	Alternate SR enabled

4:3 **Sync Polarity**

This bit indicates the polarity of Hsync and Vsync to be transmitted in MSA.

Value	Name	Description
00b	Low	VS and HS are active low (inverted)
01b	VS Low, HS High	VS is active low (inverted), HS is active high
10b	VS High, HS Low	VS is active high, HS is active low (inverted)
11b	High **[Default]**	VS and HS are active high

2 **Digital Display Detected**

Access:	RO

This bit indicates whether a digital display was detected during initialization.
It signifies the level of the detect pin at boot. This bit is valid regardless of whether the port is enabled.

Value	Name	Description
0b	Not Detected **[Default]**	Digital display not detected during initialization
1b	Detected	Digital display detected during initialization

1:0 **Reserved**

Format:	MBZ

Adjustable Voltage Swing Programming for Embedded DisplayPort

Vdiff Single Ended Swing (Transition Bits)	DP_AFE_DP_IREFCAL <16:12> TXIRefSel	DP_AFE_DP_IREFCAL <11:7> TXVcmSel <6:2> TXVrefSel (Both fields set to same value)
500 mV	2 decimal	3 decimal
475 mV	3 decimal	28 decimal
450 mV	4 decimal	12 decimal
425 mV	5 decimal	20 decimal
400 mV	6 decimal	4 decimal
375 mV	7 decimal	29 decimal
350 mV	8 decimal	13 decimal
325 mV	9 decimal	21 decimal
300 mV	10 decimal	5 decimal
275 mV	11 decimal	30 decimal

Vdiff Single Ended Swing (Transition Bits)	DP_AFE_DP_IREFCAL <16:12> TXIRefSel	DP_AFE_DP_IREFCAL <11:7> TXVcmSel <6:2> TXVrefSel (Both fields set to same value)
250 mV	12 decimal	14 decimal
225 mV	13 decimal	22 decimal
200 mV	14 decimal	5 decimal
175 mV	15 decimal	31 decimal
150 mV	16 decimal	15 decimal
125 mV	17 decimal	23 decimal
100 mV	18 decimal	7 decimal

This is only for use with the DP_CTL_A <27> Vswing_Emp_Set selections marked as "adjustable".

DPAFE_DP_IREFCAL <31> SwingCtlDis must be set to 1.

DP_CTL_A <27> Vswing_Emp_Set fullSwingMode must be set to 1.

DP_CTL_A <23:22> Vswing_Emp_Set dmpen and dmplv are set to 00b for no de-emphasis, 10b for 3.5dB de-emphasis, or 11b for 6dB de-emphasis.

4.4.2 DP_AUX_CTL-DisplayPort AUX Channel Control

DP_AUX_CTL

Register Space:	MMIO: 0/2/0
Project:	
Default Value:	0x000300C8
Access:	R/W Special
Size (in bits):	32

Address:	64010h-64013h
Name:	DisplayPort A AUX Channel Control
ShortName:	DP_AUX_CTL_A

DWord	Bit	Description
0	31	**Send Busy** Access: R/W Special Setting this bit to a one initiates the transaction, when read this bit will be a 1 until the transmission completes. The transaction is completed when the response is received or when a timeout occurs. Do not write a 1 again until transaction completes. **Programming Notes** Restriction : Do not change any fields while Send/Busy bit 31 is asserted.
	30	**Done** Access: R/WC A sticky bit that indicates the transaction has completed. Write a 1 to this bit to clear the event Value: 0b — Name: Not done Value: 1b — Name: Done
	29	**Interrupt on Done**

DP_AUX_CTL

Access:	R/W

Enable an interrupt in the hotplug status register when the transaction completes or times out.

Value	Name
0b	Enable
1b	Disable

28 **Time out error**

Access:	R/WC

A sticky bit that indicates the transaction has timed out.
Write a 1 to this bit to clear the event.

27:26 **Time out timer value**

Access:	R/W

Used to determine how long to wait for receiver response before timing out.

Value	Name	Description
00b	400us **[Default]**	400us
01b	600us	600us
10b	800us	800us
11b	1600us	1600us

25 **Receive error**

Access:	R/WC

A sticky bit that indicates that the data received was corrupted, not in multiples of a full byte, or more than 20 bytes.
Write a 1 to this bit to clear the event.

Value	Name
0b	Error **[Default]**
1b	Not Error

24:20 **Message Size**

This field is used to indicate the total number bytes to transmit (including the header). It also indicates the number of bytes received in a transaction (including the header).
Reads of this field are valid only when the done bit is set and timeout or receive error has not occurred.
Sync/Stop are not part of the message or the message size.
Reads of this field will give the response message size.
The read value will not be valid while Send/Busy bit 31 is asserted.
Message sizes of 0 or >20 are not allowed.

19:16 **Precharge Time**

Default Value:	0011b 6us
Access:	R/W

Used to determine the precharge time for the Aux Channel drivers. During this time the Aux Channel will drive the SYNC pattern. Every microsecond gives one additional SYNC pulse beyond the hard coded 26 SYNC pulses.
The value is the number of microseconds times 2.
Default is 3 decimal which gives 6us of precharge which is 6 extra SYNC pulses for a total of 32.

10:0 **2X Bit Clock divider**

Default Value:	00 1100 1000b 200

Used to determine the 2X bit clock the Aux Channel logic runs on.
This value divides the input clock frequency down to 2X bit clock rate. It should be programmed to get as close as possible to the ideal rate of 2MHz.

DP_AUX_CTL		
		Default is 200 decimal which divides the 400MHz input clock to become 2MHz bit clock.

4.4.3 DP_AUX_DATA-DisplayPort AUX Channel Data

DP Aux Channel Data Format		
Project:		
Default Value:		0x00000000
DWord	**Bit**	**Description**
0	31:0	**AUX CH DATA** A DWord of the message. Writes give the data to transmit during the transaction. The MSbyte is transmitted first. Reads will give the response data after transaction complete.

DP_AUX_DATA		
Register Space:	MMIO: 0/2/0	
Project:		
Default Value:	0x00000000, 0x00000000, 0x00000000, 0x00000000, 0x00000000	
Access:	R/W	
Size (in bits):	5x32	
Address:	64014h-64027h	
Name:	DisplayPort A AUX Channel Data	
ShortName:	DP_AUX_DATA_[1-5]_A	

The read value will not be valid while the DisplayPort Aux Channel Control Register Send/Busy bit is asserted.

DWord	Bit	Description
0	31:0	**AUX CH DATA1** Format: DP Aux Channel Data Format
1	31:0	**AUX CH DATA2** Format: DP Aux Channel Data Format
2	31:0	**AUX CH DATA3** Format: DP Aux Channel Data Format
3	31:0	**AUX CH DATA4** Format: DP Aux Channel Data Format
4	31:0	**AUX CH DATA5** Format: DP Aux Channel Data Format

4.5 Panel Fitter

4.5.1 PF_PWR_GATE-Panel Fitter Power Gate Control

PF_PWR_GATE	
Register Space:	MMIO: 0/2/0
Project:	
Default Value:	0x00000000
Access:	R/W
Size (in bits):	32
Double Buffer Update Point:	Start of vertical blank after armed
Double Buffer Armed By:	Write to PF_WIN_SZ
Address:	68060h-68063h
Name:	PF 0 Power Gate Control
ShortName:	PF_PWR_GATE_0
Address:	68860h-68863h
Name:	PF 1 Power Gate Control
ShortName:	PF_PWR_GATE_1
Address:	69060h-69063h
Name:	PF 2 Power Gate Control
ShortName:	PF_PWR_GATE_2

DWord	Bit	Description
0	31:5	**Reserved** Format: MBZ
	4:3	**Settling Time** Time for RAMs in a given filter group to settle after they are powered up. 00b – 32 cdclks – 80ns 01b – 64 cdclks – 160ns 10b – 96 cdclks – 240ns 11b – 128 cdcclks – 320ns
	2	**Reserved** Format: MBZ
	1:0	**SLPEN Delay** Delay between sleep enables of individual banks of RAMs. 00b – 8 cdclks – 20ns 01b – 16 cdclks – 40ns 10b – 24 cdclks – 60ns 11b – 32 cdclks – 80ns

4.5.2 PF_WIN_POS-Panel Fitter Window Position

PF_WIN_POS	
Register Space:	MMIO: 0/2/0
Project:	
Default Value:	0x00000000
Access:	R/W
Size (in bits):	32
Double Buffer Update Point:	Start of vertical blank after armed
Double Buffer Armed By:	Write to PF_WIN_SZ
Address:	68070h-68073h
Name:	PF 0 Window Position
ShortName:	PF_WIN_POS_0
Address:	68870h-68873h
Name:	PF 1 Window Position
ShortName:	PF_WIN_POS_1
Address:	69070h-69073h
Name:	PF 2 Window Position
ShortName:	PF_WIN_POS_2

DWord	Bit	Description
0	31:29	**Reserved** Format: MBZ
	28:16	**XPOS** The X coordinate in pixels of the upper left most pixel of the panel fitted display window. **Programming Notes** Restriction : The X position must not be programmed to be 1 (28:16=0 0000 0000 0001b).
	15:12	**Reserved** Format: MBZ
	11:0	**YPOS** The Y coordinate in lines of the upper left most pixel of the panel fitter display window. **Programming Notes** Restriction : LSB must be zero for interlaced modes.

4.5.3 PF_WIN_SZ-Panel Fitter Window Size

PF_WIN_SZ		
Register Space:		MMIO: 0/2/0
Project:		
Default Value:		0x00000000
Access:		R/W
Size (in bits):		32
Double Buffer Update Point:		Start of vertical blank
Address:	68074h-68077h	
Name:	PF 0 Window Size	
ShortName:	PF_WIN_SZ_0	
Address:	68874h-68877h	
Name:	PF 1 Window Size	
ShortName:	PF_WIN_SZ_1	
Address:	69074h-69077h	
Name:	PF 2 Window Size	
ShortName:	PF_WIN_SZ_2	

Coordinates are determined with a value of (0,0) being the upper left corner of the display device (rotation does not affect this).
Writes to this register arm PF registers on this pipe.

DWord	Bit	Description
0	31:29	**Reserved** Format: MBZ
	28:16	**XSIZE** The horizontal size in pixels of the desired panel fitted window.
	15:12	**Reserved** Format: MBZ
	11:0	**YSIZE** The vertical size in pixels of the desired panel fitted window. **Programming Notes** Restriction : LSB must be zero for interlaced modes.

4.5.4 PF_CTRL-Panel Fitter Control

PF_CTRL	
Register Space:	MMIO: 0/2/0
Project:	
Default Value:	0x00000000
Access:	R/W
Size (in bits):	32
Double Buffer Update Point:	Start of vertical blank after armed
Double Buffer Armed By:	Write to PF_WIN_SZ
Address:	68080h-68083h
Name:	PF 0 Control
ShortName:	PF_CTRL_0
Address:	68880h-68883h
Name:	PF 1 Control
ShortName:	PF_CTRL_1
Address:	69080h-69083h
Name:	PF 2 Control
ShortName:	PF_CTRL_2

There are three panel fitters:
Panel fitter 0 is always 7x5 filter capable.
Panel fitter 1 is always 3x3 filter capable.
Panel fitter 2 is always 3x3 filter capable.

The 3x3 capable filter can support pipe horizontal source sizes less than or equal to 2048 pixels. It must not be enabled when the pipe horizontal source size is greater than 2048 pixels.

The 7x5 capable filter can support pipe horizontal source sizes of less than or equal to 4096 pixels. When the pipe horizontal source size is greater than 2048 pixels, or the hard-coded 3x3 filter coefficients are selected, the filter will automatically switch to a 3x3 filter mode. It must not be enabled when the pipe horizontal source size is greater than 4096 pixels.

When using panel fitter down scaling (pipe source size is larger than panel fitter window size) the maximum supported pixel rate will be reduced by the down scale amount and the watermark for planes on the same pipe has to increase by the down scale amount.

Programming Notes

Restriction : The 3x3 capable filter must not be enabled when the pipe horizontal source size is greater than 2048 pixels.
The 7x5 capable filter must not be enabled when the pipe horizontal source size is greater than 4096 pixels.

Down scaling is only supported up to 1.125 (pipe source size / panel fitter window size) in each direction.

DWord	Bit	Description		
0	31	**Enable Pipe Scaler**		
		Value	**Name**	**Description**
		0b	Disable **[Default]**	Data bypasses the scaler
		1b	Enable	The scaler is enabled
	30:29	**Pipe Select**		

PF_CTRL

This bit determines which display pipe this panel fitter will connect to. Do not enable and connect more than one panel fitter to a pipe.

Value	Name	Description
00b	Pipe A **[Default]**	Pipe A
01b	Pipe B	Pipe B
10b	Pipe C	Pipe C
11b	Reserved	Reserved

24:23 **FILTER SELECT**

Selects filter coefficients.

Value	Name	Description
00b	Programmed	Programmed Coefficients (Recommended for 7x5 capable panel fitters, not available for 3x3 capable panel fitters
01b	Hardcoded Med	Hardcoded Coefficients for Medium 3x3 Filtering
10b	Hardcoded Edge Enhance	Hardcoded Coefficients for Edge Enhancing 3x3 Filtering
11b	Hardcoded Edge Soften	Hardcoded Coefficients for Edge Softening 3x3 Filtering

Programming Notes

Restriction : Programmed coefficients only work with 7x5 capable panel fitters.
For panel fitters that are only 3x3 capable, this field must be programmed to select one of the hardcoded coefficient sets.

22 **Reserved**

Format:	MBZ

16:0 **Reserved**

Format:	MBZ

4.5.5 PF_COEF_INDEX-Panel Fitter Coefficients Index

Horizontal coefficients are accessed through the index and data registers following the mapping shown below. 17 phase of 7 taps requires 119 coefficients in 60 dwords per set. The letter represents the filter tap (D is the center tap) and the number represents the coefficient set for a phase (0-16).

Horizontal Luma/Red Coefficient Mapping			Horizontal Chroma/Green/Blue Coefficient Mapping		
Index Value	**Data Value Coefficient2**	**Data Value Coefficient1**	**Index Value**	**Data Value Coefficient2**	**Data Value Coefficient1**
00h	B0	A0	3Ch	B0	A0
01h	D0	C0	3Dh	D0	C0
02h	F0	E0	3Eh	F0	E0
03h	A1	G0	3Fh	A1	G0
04h	C1	B1	40h	C1	B1
...	...	...	...	...	...
38h	B16	A16	74h	B16	A16
39h	D16	C16	75h	D16	C16
3Ah	F16	E16	76h	F16	E16
3Bh	Reserved	G16	77h	Reserved	G16

Vertical coefficients are accessed through the index and data registers following the mapping shown below. 17 phase of 5 taps requires 85 coefficients in 43 dwords per set. The letter represents the filter tap (C is the center tap) and the number represents the coefficient set for a phase (0-16).

Vertical Luma/Red Coefficient Mapping

Index Value	Data Value Coefficient2	Data Value Coefficient1
00h	B0	A0
01h	D0	C0
02h	A1	E0
03h	C1	B1
...	...	...
27h	B16	A16
28h	D16	C16
2Ah	Reserved	E16

Vertical Chroma/Green/Blue Coefficient Mapping

Index Value	Data Value Coefficient2	Data Value Coefficient1
2Bh	B0	A0
2Ch	D0	C0
2Dh	A1	E0
2Eh	C1	B1
...	...	...
53h	B16	A16
54h	D16	C16
55h	Reserved	E16

PF_COEF_INDEX

Register Space:	MMIO: 0/2/0
Project:	
Default Value:	0x00000000
Access:	R/W
Size (in bits):	32

Address:	680A0h-680A3h
Name:	PF 0 Horizontal Coefficients Index
ShortName:	PF_HCOEF_INDEX_0

Address:	680B0h-680B3h
Name:	PF 0 Vertical Coefficients Index
ShortName:	PF_VCOEF_INDEX_0

This index controls access to the array of panel fitter coefficient data values. See the coefficient mapping tables for information on mapping of index to data values for each set of coefficients.

DWord	Bit	Description
0	31:16	**Reserved** Format: MBZ
	15	**Index Auto Increment** This field enables the index auto increment. Value / Name / Description: 0b / No Increment **[Default]** / Do not automatically increment the index value. 1b / Auto Increment / Increment the index value with each read or write to the data register.
	14:7	**Reserved** Format: MBZ
	6:0	**Index Value** This field indicates the data location to be accessed through the data register. This value can be automatically incremented by a read or a write to the data register if the index auto increment bit is set. When automatically incrementing, the current automatically calculated index value can be read here. When automatically incrementing, the index will roll over to 0 after reaching the end of the allowed range.

4.5.6 PF_COEF_DATA-Panel Fitter Coefficients Data

Panel Fitter Coefficient Format

Project:

Default Value: 0x00000000

DWord	Bit	Description
0	15	**Sign** Value: 0b – Positive – Positive Value: 1b – Negative – Negative
	14	**Reserved** Format: MBZ
	13:12	**Exponent** The meaning of the exponent bits varies for center tap or non-center tap coefficients. 00b – 2 or 0.125 – Center taps: 2 or mantissa is b.bbbbbbbb; Non-center taps: 0.125 or mantissa is 0.000bbbbbbb 01b – 1 – 1 or mantissa is 0.bbbbbbb.. 10b – 0.5 – 0.5 or mantissa is 0.0bbbbbbb.. 11b – 0.25 – 0.25 or mantissa is 0.00bbbbbbb.. Others – Reserved – Reserved
	11:3	**Mantissa** Size of the mantissa varies based on the filter, but the MSB of the mantissa is always bit 11. Center tap coefficients use all 9 bits of mantissa. Non-center tap coefficients use only the upper 7 bits of mantissa and the lower 2 bits are ignored.
	2:0	**Reserved** Format: MBZ

PF_COEF_DATA

Register Space:	MMIO: 0/2/0
Project:	
Default Value:	0x00000000
Access:	R/W (DWORD access only, no byte access)
Size (in bits):	32
Address:	680A4h-680A7h
Name:	PF 0 Horizontal Coefficients Data
ShortName:	PF_HCOEF_DATA_0
Address:	680B4h-680B7h
Name:	PF 0 Vertical Coefficients Data
ShortName:	PF_VCOEF_DATA_0

These are the coefficient values for panel fitter.
The Panel Fitter Coefficients Index Value indicates the Panel Fitter Coefficients location to be accessed through this register.

PF_COEF_DATA

Only 7x5 capable panel fitters can use programmed coefficients.

DWord	Bit	Description
0	31:16	**Coefficient2** Format: Panel Fitter Coefficient Format Specifies the value for the second coefficient stored in this dword.
	15:0	**Coefficient1** Format: Panel Fitter Coefficient Format Specifies the value for the first coefficient stored in this dword.

5. North Display Engine Pipe and Plane Controls

5.1 Pipe Control

5.1.1 PIPE_SCANLINE-Pipe Scan Line

PIPE_SCANLINE		
Register Space:		MMIO: 0/2/0
Project:		
Default Value:		0x00000000
Access:		RO
Size (in bits):		32
Address:	70000h-70003h	
Name:	Pipe A Scan Line	
ShortName:	PIPE_SCANLINE_A	
Address:	71000h-71003h	
Name:	Pipe B Scan Line	
ShortName:	PIPE_SCANLINE_B	
Address:	72000h-72003h	
Name:	Pipe C Scan Line	
ShortName:	PIPE_SCANLINE_C	

This register provides the read back of the display pipe vertical line counter.
The value increments at the leading edge of HSYNC and can be safely read any time.
The value resets to line zero at the first active line of the display.
In interlaced display timings, the scan line counter provides the current line in the field. One field can have a total number of lines that is one greater than the other field.

DWord	Bit	Description
0	31	**Current Field** Provides read back of the current field being displayed on the display pipe.
	30:13	**Reserved**
	12:0	**Line Counter for Display** Provides read back of the display pipe vertical line counter. This is an indication of the current display scan line.

Value	Name	Description
0b	Odd **[Default]**	First field (odd field)
1b	Even	Second field (even field)

5.1.2 PIPE_SCANLINECOMP-Pipe Scan Line Compare

PIPE_SCANLINECOMP	
Register Space:	MMIO: 0/2/0
Project:	
Default Value:	0x00000000
Access:	R/W
Size (in bits):	32
Address:	70004h-70007h
Name:	Pipe A Scan Line Range Compare
ShortName:	PIPE_SCANLINECOMP_A
Address:	71004h-71007h
Name:	Pipe B Scan Line Range Compare
ShortName:	PIPE_SCANLINECOMP_B
Address:	72004h-72007h
Name:	Pipe C Scan Line Range Compare
ShortName:	PIPE_SCANLINECOMP_C

Description	Project
The scan line number register is compared with the display line value from the pipe timing generator. The result of this comparison is used to generate interrupts and render responses. The value programmed should be desired value - 1, so for line 0, the value programmed is VTOTAL, and for line 1, the value programmed is 0. In interlaced display timings, the scan line is per field. One field can have a total number of lines that is one greater than the other field.	
Reserved. Scan line compare is now done through DE_LOAD_SL (pipe A 0x70068, pipe B 0x71068, pipe C 0x72068).	

DWord	Bit	Description
0	31:13	**Reserved** Format: MBZ
	12:0	**Reserved** Project:
	12:0	**Scan Line Number** Range: 0..Vertical Total This field specifies the scan line number on which to generate scan line interrupt and render response.

5.1.3 PIPE_CONF-Pipe Configuration

PIPE_CONF	
Register Space:	MMIO: 0/2/0
Project:	
Default Value:	0x00000000
Access:	R/W
Size (in bits):	32
Double Buffer Update Point:	Start of vertical blank OR pipe disabled
Address:	70008h-7000Bh
Name:	Pipe A Config
ShortName:	PIPE_CONF_A
Address:	71008h-7100Bh
Name:	Pipe B Config
ShortName:	PIPE_CONF_B
Address:	72008h-7200Bh
Name:	Pipe C Config
ShortName:	PIPE_CONF_C

DWord	Bit	Description
0	31	**Pipe Enable** Setting this bit to the value of one, turns on this pipe. Turning the pipe off disables the timing generator and synchronization pulses to the display will not be maintained. Pipe timing registers must contain valid values before this bit is enabled.

Value	Name	Description
0b	Disable **[Default]**	Disable
1b	Enable	Enable

DWord	Bit	Description
	30	**Pipe State** This read only bit indicates the actual state of the pipe.

Value	Name	Description
0b	Disabled **[Default]**	Pipe is disabled
1b	Enabled	Pipe is enabled

DWord	Bit	Description
	29	**Reserved**
	26	**Reserved**
	25:24	**Pipe Palette Gamma Mode** These bits select which mode the pipe gamma correction logic works in. See the Display Palette Registers section for information on the different palette/gamma modes. Other gamma units such as in the sprite are unaffected by this bit.

Value	Name	Description
00b	8 bit **[Default]**	8-bit Legacy Palette Mode
01b	10 bt	10-bit Precision Palette Mode
10b	12 bit	12-bit Interpolated Gamma Mode
11b	Split	Split Gamma Mode (separate pipe gamma functions before and after pipe CSC)

DWord	Bit	Description
	23:21	**Interlaced Mode** These bits are used for software control of the pipe interlaced mode.

Value	Name	Description

000b	PF-PD **[Default]**	Progressive Fetch with Progressive Display.
001b	PF-ID	Progressive Fetch with Interlaced Display. Requires 7x5 capable panel fitter to be enabled and in 7x5 mode
011b	IF-ID	Interlaced Fetch with Interlaced Display
Others	Reserved	Reserved

Programming Notes

Restriction : VGA display modes do not work while in interlaced fetch modes.

20 **Display Power Mode Switch**

This bit selects the the software controlled progressive-to-progressive power saving mode (software controlled DRRS).
Link and data M/N 1 values are used for normal settings, M/N 2 values for low power settings.
Pixel clock FP0 values are used for normal settings, FP1 values for low power settings.

Value	Name	Description
0b	Normal **[Default]**	Normal progressive refresh rate
1b	Low Power	Low power progressive refresh rate

Programming Notes

Workaround : If this pipe is connected to a port on the PCH and this power savings mode will be used, then before the pipe and transcoder are enabled, the frame start delay in the pipe and transcoder must be set to 11b. When these conditions are no longer true, the frame start delay must be returned to the previous value after the pipe and transcoder are disabled.
Frame start delay register locations:
Pipe A frame start delay 0x70008 bits 28:27
Pipe B frame start delay 0x71008 bits 28:27
Pipe C frame start delay 0x72008 bits 28:27
Cougarpoint/Pantherpoint transcoder A frame start delay 0xF0064 bits 28:27
Cougarpoint/Pantherpoint transcoder B frame start delay 0xF1064 bits 28:27
Cougarpoint/Pantherpoint transcoder C frame start delay 0xF2064 bits 28:27

Workaround (WaFrameStartDelayWaForSDRRS) : If this pipe is connected to the LVDS port and LVDS clockgating is disabled (bits 14, 30 set to 1b in 0xC2020), then clockgating must be temporarily enabled (bits 14, 30 cleared to 0b) when toggling Display Power Mode Switch followed by wait of 2 vblanks and then disabled again.

19:18 **MSA Timing Delay**

This field selects the vertical blank line on which MSA is sent.
It is intended for use with embedded DisplayPort panels that support sDRRS.
The sDRRS timing switch shall occur on same line as the MSA.

Value	Name	Description
00b	Line1 **[Default]**	MSA and sDRRS timing switch occur within the first line of vertical blank
01b	Line2	MSA and sDRRS timing switch occur within the second line of vertical blank
10b	Line3	MSA and sDRRS timing switch occur within the third line of vertical blank
11b	Line4	MSA and sDRRS timing switch occur within the fourth line of vertical blank

17:16 **Reserved**

Format:	MBZ

15:14 **Display Rotation Info**

These are informative bits set by software to indicate this pipe is being rotated.
Software should set these for both hardware and software rotation cases.

PIPE_CONF

Hardware rotation is not enabled through these bits.

Value	Name	Description
00b	None **[Default]**	No rotation on this pipe
01b	90	90 degree rotation on this pipe
10b	180	180 degree rotation on this pipe
11b	270	270 degree rotation on this pipe

13 **Color Range Select**

This bit is used to select the color range of outputs.
When CE color range is selected the pipe output will be compressed and offset to the CE range.

Value	Name	Description
0b	Full **[Default]**	Apply full color range to the output
1b	CE	Apply CE color range to the output

12:11 **Pipe output color space select**

This field informs the ports of the pipe output color space.
Plane data formats and CSC need to be programmed to match what is selected here.

Value	Name	Description
00b	RGB **[Default]**	RGB
01b	YUV 601	YUV 601
10b	YUV 709	YUV 709
11b	Reserved	Reserved

10 **xcYCC Color Range Limit**

This bit is used to limit the color range of the port outputs from 1 to 254 for 8-bit components, 4 to 109 for 10bit components, or 16 to 4079 for 12-bit components.
Values outside of the range will be clamped to fit within the range.
There is no need to set the equivalent bit in the south display transcoder configuration register if the bit is set in this register.

Value	Name	Description
0b	Full **[Default]**	Do not limit the range
1b	Limit	Limit range

9 **Reserved**

Format:	MBZ

8 **BFI enable**

This bit enables black frame insertion on this pipe.
This bit should not be changed while the pipe or port are enabled.

Value	Name	Description
0b	Disable **[Default]**	BFI disabled
1b	Enable	BFI enabled

7:5 **Bits Per Color**

This field selects the number of bits per color output on ports connected to this pipe.
Software should enable dithering if selecting a pixel color depth higher or lower than the pixel color depth of the frame buffer.

Value	Name	Description
000b	8 bpc **[Default]**	8 bits per color
001b	10 bpc	10 bits per color
010b	6 bpc	6 bits per color
011b	12 bpc	12 bits per color
Others	Reserved	Reserved

4 **Dithering enable**

This bit enables dithering

PIPE_CONF				
		Value	Name	Description
		0b	Disable **[Default]**	Dithering disabled
		1b	Enable	Dithering enabled
	3:2	**Dithering type** These bits select dithering type.		
		Value	Name	Description
		00b	Spatial **[Default]**	Spatial
		10b	ST2	Spatio-Temporal 2
	1:0	**Reserved**		
		Format:		MBZ

5.1.4 PIPE_FRMCNT-Pipe Frame Count

PIPE_FRMCNT		
Register Space:		MMIO: 0/2/0
Project:		
Default Value:		0x00000000
Access:		RO
Size (in bits):		32
Address:	70040h-70043h	
Name:	Pipe A Frame Count	
ShortName:	PIPE_FRMCNT_A	
Address:	71040h-71043h	
Name:	Pipe B Frame Count	
ShortName:	PIPE_FRMCNT_B	
Address:	72040h-72043h	
Name:	Pipe C Frame Count	
ShortName:	PIPE_FRMCNT_C	

DWord	Bit	Description
0	31:0	**Pipe Frame Counter** Provides read back of the display pipe frame counter. This counter increments on every start of vertical blank and rolls over back to 0 after (2^32)-1 frames.

5.1.5 PIPE_FLIPCNT-Pipe Flip Count

PIPE_FLIPCNT		
Register Space:		MMIO: 0/2/0
Project:		
Default Value:		0x00000000
Access:		RO
Size (in bits):		32
Address:	70044h-70047h	
Name:	Pipe A Flip Count	
ShortName:	PIPE_FLIPCNT_A	
Address:	71044h-71047h	
Name:	Pipe B Flip Count	
ShortName:	PIPE_FLIPCNT_B	
Address:	72044h-72047h	
Name:	Pipe C Flip Count	
ShortName:	PIPE_FLIPCNT_C	

DWord	Bit	Description
0	31:0	**Pipe Flip Counter** This field provides read back of the display pipe flip counter. The counter increments on the start of each flip to the primary plane of this pipe. The start of flip is when the plane surface address is updated, not when the flip completes. The flip can be through command streamer asynchronous and synchronous flips or MMIO writes to the primary plane surface address. It rolls over back to 0 after (2^32)-1 flips.

5.1.6 PIPE_FRMTMSTMP-Pipe Frame Time Stamp

PIPE_FRMTMSTMP		
Register Space:		MMIO: 0/2/0
Project:		
Default Value:		0x00000000
Access:		R/W
Size (in bits):		32
Address:	70048h-7004Bh	
Name:	Pipe A Frame Time Stamp	
ShortName:	PIPE_FRMTMSTMP_A	
Address:	71048h-7104Bh	
Name:	Pipe B Frame Time Stamp	
ShortName:	PIPE_FRMTMSTMP_B	
Address:	72048h-7204Bh	
Name:	Pipe C Frame Time Stamp	
ShortName:	PIPE_FRMTMSTMP_C	

DWord	Bit	Description
0	31:0	**Pipe Frame Time Stamp** This field provides read back of the display pipe frame time stamp. The time stamp value is sampled at every start of vertical blank. The TIMESTAMP_CTR register has the current time stamp value.

5.1.7 PIPE_FLIPTMSTMP-Pipe Flip Time Stamp

PIPE_FLIPTMSTMP		
Register Space:		MMIO: 0/2/0
Project:		
Default Value:		0x00000000
Access:		R/W
Size (in bits):		32
Address:	7004Ch-7004Fh	
Name:	Pipe A Flip Time Stamp	
ShortName:	PIPE_FLIPTMSTMP_A	
Address:	7104Ch-7104Fh	
Name:	Pipe B Flip Time Stamp	
ShortName:	PIPE_FLIPTMSTMP_B	
Address:	7204Ch-7204Fh	
Name:	Pipe C Flip Time Stamp	
ShortName:	PIPE_FLIPTMSTMP_C	

DWord	Bit	Description
0	31:0	**Pipe Flip Time Stamp** This field provides read back of the display pipe flip time stamp. The time stamp value is sampled on the start of each flip to the primary plane of this pipe. The start of flip is when the plane surface address is updated, not when the flip completes. The flip can be through command streamer asynchronous and synchronous flips or MMIO writes to the primary plane surface address. The TIMESTAMP_CTR register has the current time stamp value.

5.2 Cursor Plane

The CUR_CTL and CUR_FBC_CTL active registers will be updated on the vertical blank or when pipe is disabled, or when cursor is not yet enabled – thus providing an atomic update of those registers together with the CUR_BASE register.

5.2.1 CUR_CTL-Cursor Control

CUR_CTL

Register Space:	MMIO: 0/2/0
Project:	
Default Value:	0x00000000
Access:	R/W
Size (in bits):	32
Double Buffer Update Point:	Start of vertical blank or pipe disabled or cursor disabled, after armed
Double Buffer Armed By:	Write to CUR_BASE

Address:	70080h-70083h
Name:	Cursor A Control
ShortName:	CUR_CTL_A

Address:	71080h-71083h
Name:	Cursor B Control
ShortName:	CUR_CTL_B

Address:	72080h-72083h
Name:	Cursor C Control
ShortName:	CUR_CTL_C

The cursor is enabled by programming a valid cursor mode in the cursor mode select fields. The cursor is disabled by programming all 0s in the cursor mode select fields.

DWord	Bit	Description
0	31:28	**Reserved**
	26	**Gamma Enable** This bit enables pipe gamma correction for the cursor pixel data.

Value	Name	Description
0b	Disable **[Default]**	Cursor pixel data bypasses pipe gamma correction
1b	Enable	Cursor pixel data passes through pipe gamma correction

	25	**Reserved**
	24	**Pipe CSC Enable** This bit enables pipe color space conversion for the cursor pixel data.

Value	Name	Description
0b	Disable **[Default]**	Cursor pixel data bypasses pipe color space conversion
1b	Enable	Cursor pixel data passes through pipe color space conversion

	23:16	**Reserved**
	15	**180 Rotation** This mode causes the cursor image to be rotated 180 degrees. In addition to setting this bit, software must also adjust the cursor position to match the physical orientation of the display.

Value	Name	Description
0b	None **[Default]**	No rotation
1b	180	180 degree rotation (only for 32 bit per pixel cursors)

Programming Notes

CUR_CTL

Restriction : Only 32 bits per pixel cursors can be rotated. This field must be zero when the cursor format is 2 bits per pixel.

14 **Trickle Feed Enable**

Value	Name	Description
0b	Enable **[Default]**	Trickle Feed Enabled - Data requests are sent whenever there is space in the Display Data Buffer
1b	Disable	Trickle Feed Disabled - Data requests are sent in bursts

13:12 **Reserved**

11:10 **Force Alpha Plane Select**

This field selects which planes the cursor alpha value will be forced for. It is used together the the Force_Alpha_Value field.

Value	Name	Description
00b	Disable **[Default]**	Disable alpha forcing
01b	Sprite	Enable alpha forcing where cursor overlaps sprite pixels
10b	Primary	Enable alpha forcing where cursor overlaps primary pixels
11b	Both	Enable alpha forcing where cursor overlaps either sprite or primary pixels.

9:8 **Force Alpha Value**

This field controls the behavior of cursor when alpha blending onto certain plane pixels. Force Alpha is only for use with ARGB cursor formats. It is used together with the Force_Alpha_Plane_Select field.

Value	Name	Description
00b	Disable **[Default]**	Cursor pixels alpha blend normally over any plane
01b	50	Cursor pixels with alpha >= 50% are made fully opaque where they overlap the selected plane(s). Cursor pixels with alpha < 50% are made fully transparent where they overlap the selected plane(s).
10b	75	Cursor pixels with alpha >= 75% are made fully opaque where they overlap the selected plane(s). Cursor pixels with alpha < 75% are made fully transparent where they overlap the selected plane(s).
11b	100	Cursor pixels with alpha = 100% are made fully opaque where they overlap the selected plane(s). Cursor pixels with alpha < 100% are made fully transparent where they overlap the selected plane(s).

7:6 **Reserved**

5 **Cursor Mode Select 5**

This bit together with bits 2:0 select the mode for cursor as shown in the cursor mode select table below.

Bit 5	Bits 2:0	Mode
0	000	Cursor is disabled
0	001	Reserved
0	010	128 x 128 32bpp AND/INVERT See description off 64 x 64 32bpp AND/INVERT format
0	011	256 x 256 32bpp AND/INVERT See description off 64 x 64 32bpp AND/INVERT format
0	100	64 x 64 2bpp Indexed 3-color and transparency mode
0	101	64 x 64 2bpp Indexed AND/XOR 2-plane mode
0	110	64 x 64 2bpp Indexed 4-color mode
0	111	64 x 64 32bpp AND/INVERT Least significant three bytes provides cursor RGB 888 color information Most Significant Byte: All Ones: Opaque, show the cursor color All Zeros: Transparent (color must also equal zero) Other: Invert the underlying display pixel data (ignore the color)

CUR_CTL				
		1	000	Reserved
		1	001	Reserved
		1	010	128 x 128 32bpp ARGB (8:8:8:8 MSB-A:R:G:B)
		1	011	256 x 256 32bpp ARGB (8:8:8:8 MSB-A:R:G:B)
		1	100	64 x 64 32bpp AND/XOR Least significant three bytes provides cursor RGB 888 color information Most Significant Byte: All Ones: Opaque, show the cursor color All Zeros: Transparent (color must also equal zero) Other: XOR the cursor color with the underlying display pixel data
		1	101	128 x 128 32bpp AND/XOR See description off 64 x 64 32bpp AND/XOR format
		1	110	256 x 256 32bpp AND/XOR See description off 64 x 64 32bpp AND/XOR format
		1	111	64 x 64 32bpp ARGB (8:8:8:8 MSB-A:R:G:B)
		Note: The cursor vertical size can be overriden by the size reduction mode Note: INVERT, XOR, and alpha blends may not look as expected when the plane underlying the cursor is YUV or extended range RGB. Out of range RGB values will be clamped prior to alpha blending, INVERT, or XOR with cursor. It is recommended to use Force_Alpha when cursor is alpha blending onto an plane of a different color space or extended gamut		
	4:3	**Reserved**		
	2:0	**Cursor Mode Select 2 0** These three bits together with bit 5 select the mode for cursor as shown in the table described in bit 5.		

5.2.2 CUR_BASE-Cursor Base Address

CUR_BASE	
Register Space:	MMIO: 0/2/0
Project:	
Default Value:	0x00000000
Access:	R/W
Size (in bits):	32
Double Buffer Update Point:	Start of vertical blank or pipe disabled
Address:	70084h-70087h
Name:	Cursor A Base Address
ShortName:	CUR_BASE_A
Address:	71084h-71087h
Name:	Cursor B Base Address
ShortName:	CUR_BASE_B
Address:	72084h-72087h
Name:	Cursor C Base Address
ShortName:	CUR_BASE_C
Writes to this register arm cursor registers for this pipe. This register specifies the graphics memory address at which the cursor image data is located.	

DWord	Bit	Description
0	31:12	**Cursor Base** Format: GraphicsAddress[31:12] This field specifies bits 31:12 of the graphics address of the base of the cursor. The cursor surface address must be 4K byte aligned. The cursor must be in linear memory, it cannot be tiled. When performing 180 degree rotation, this address does not need to change, hardware will internally offset to start from the last pixel of the last line of the cursor. **Programming Notes** Workaround : To prevent false VT-d type 6 errors, use 64KB address alignment and allocate an extra 2 Page Table Entries (PTEs) beyond the end of the displayed surface. Only the PTEs will be used, not the pages themselves.
	11:3	**Reserved**
	1:0	**Reserved**

5.2.3 CUR_POS-Cursor Position

CUR_POS	
Register Space:	MMIO: 0/2/0
Project:	
Default Value:	0x00000000
Access:	R/W
Size (in bits):	32
Double Buffer Update Point:	Start of vertical blank or pipe disabled
Address:	70088h-7008Bh
Name:	Cursor A Position
ShortName:	CUR_POS_A
Address:	71088h-7108Bh
Name:	Cursor B Position
ShortName:	CUR_POS_B
Address:	72088h-7208Bh
Name:	Cursor C Position
ShortName:	CUR_POS_C

This register specifies the screen position of the cursor. The cursor must have at least a single pixel positioned over the pipe source area.
The origin of the cursor position is always the upper left corner of the display pipe source image area.
When performing 180 degree rotation, the cursor image is rotated by hardware, but the position is not, so it must be adjusted by software if it is desired to maintain the same apparent position on a physically rotated display.

DWord	Bit	Description
0	31	**Y Position Sign** This specifies the sign of the vertical position of the cursor upper left corner.
	30:28	**Reserved** Format: MBZ
	27:16	**Y Position Magnitude** This specifies the magnitude of the vertical position of the cursor upper left corner in lines.

CUR_POS		
	15	**X Position Sign** This specifies the sign of the horizontal position of the cursor upper left corner.
	14:12	**Reserved** Format: MBZ
	11:0	**X Position Magnitude** This specifies the magnitude of the horizontal position of the cursor upper left corner in pixels.

5.2.4 CUR_PAL-Cursor Palette

Cursor Palette Format		
Project:		
Default Value:		0x00000000
DWord	**Bit**	**Description**
0	31:24	**Reserved** Format: MBZ
	23:16	**Palette Red** These registers specify the cursor palette. The data can be pre-gamma corrected and bypass the pipe gamma correction logic or pass through the pipe gamma correction.
	15:8	**Palette Green**
	7:0	**Palette Blue**

CUR_PAL	
Register Space:	MMIO: 0/2/0
Project:	
Default Value:	0x00000000, 0x00000000, 0x00000000, 0x00000000
Access:	R/W
Size (in bits):	4x32
Double Buffer Update Point:	Start of vertical blank or pipe disabled
Address:	70090h-7009Fh
Name:	Cursor A Palette
ShortName:	CUR_PAL_[0-3]_A
Address:	71090h-7109Fh
Name:	Cursor B Palette
ShortName:	CUR_PAL_[0-3]_B
Address:	72090h-7209Fh
Name:	Cursor C Palette
ShortName:	CUR_PAL_[0-3]_C

CUR_PAL

The cursor palette provides color information when using one of the indexed modes. The two-bit index selects one of the four colors or two of the colors when in the AND/XOR cursor mode. The cursor palette provides color information when using one of the indexed modes. The two-bit index selects one of the four colors or two of the colors when in the AND/XOR cursor mode.

The table below describes the palette usage for different cursor modes and indexes.

Index	2 color	3color	4color
00	palette 0	palette 0	palette 0
01	palette 1	palette 1	palette 1
10	transparent	transparent	palette 2
11	invert destination (palette 3 all 1s)	palette 3	palette 3

Palette 3 must be programmed with all 1s for invert destination.

DWord	Bit	Description	
0	31:0	**CUR PAL0**	
		Format:	Cursor Palette Format
1	31:0	**CUR PAL1**	
		Format:	Cursor Palette Format
2	31:0	**CUR PAL2**	
		Format:	Cursor Palette Format
3	31:0	**CUR PAL3**	
		Format:	Cursor Palette Format

5.2.5 CUR_FBC_CTL-Cursor FBC Control

CUR_FBC_CTL	
Register Space:	MMIO: 0/2/0
Project:	
Default Value:	0x00000000
Access:	R/W
Size (in bits):	32
Double Buffer Update Point:	Start of vertical blank or pipe disabled or cursor disabled, after armed
Double Buffer Armed By:	Write to CUR_BASE
Address:	700A0h-700A3h
Name:	Cursor A FBC Control
ShortName:	CUR_FBC_CTL_A
Address:	710A0h-710A3h
Name:	Cursor B FBC Control
ShortName:	CUR_FBC_CTL_B
Address:	720A0h-720A3h
Name:	Cursor C FBC Control
ShortName:	CUR_FBC_CTL_C

DWord	Bit	Description
0	31	**Size Reduction Enable** This enables cursor size reduction logic. The cursor engine will fetch and display the programmed reduced number of lines, then go transparent for the rest of the frame. The reduced scan lines value must be programmed when cursor size reduction is enabled.

Value	Name	Description
0b	Disable **[Default]**	Disable cursor size reduction
1b	Enable	Enable cursor size reduction

Programming Notes
Restriction : Cursor size reduction is not allowed with 2bpp cursor formats or cursor 180 degree rotation.

	30:8	**Reserved**
	7:0	**Reduced Scan Lines** This specifies the number of scan lines of cursor data to fetch and display when cursor size reduction is enabled. The value programmed is the size minus one.

Programming Notes
Restriction : The minimum size is 8 lines, programmed as 07h.
Restriction : The maximum size can not be greater than the normal size when size reduction is not enabled.

5.2.6 PLANE_SURFLIVE-Plane Live Base Address

PLANE_SURFLIVE		
Register Space:		MMIO: 0/2/0
Project:		
Default Value:		0x00000000
Access:		RO
Size (in bits):		32
Address:	700ACh-700AFh	
Name:	Cursor A Live Base Address	
ShortName:	CUR_SURFLIVE_A	
Address:	701ACh-701AFh	
Name:	Primary A Live Base Address	
ShortName:	PRI_SURFLIVE_A	
Address:	702ACh-702AFh	
Name:	Sprite A Live Base Address	
ShortName:	SPR_SURFLIVE_A	
Address:	710ACh-710AFh	
Name:	Cursor B Live Base Address	
ShortName:	CUR_SURFLIVE_B	
Address:	711ACh-711AFh	
Name:	Primary B Live Base Address	
ShortName:	PRI_SURFLIVE_B	
Address:	712ACh-712AFh	
Name:	Sprite B Live Base Address	
ShortName:	SPR_SURFLIVE_B	
Address:	720ACh-720AFh	
Name:	Cursor C Live Base Address	
ShortName:	CUR_SURFLIVE_C	
Address:	721ACh-721AFh	
Name:	Primary C Live Base Address	
ShortName:	PRI_SURFLIVE_C	
Address:	722ACh-722AFh	
Name:	Sprite C Live Base Address	
ShortName:	SPR_SURFLIVE_C	

DWord	Bit	Description
0	31:0	**Live Surface Base Address** This gives the live value of the surface base address as being currently used for the plane.

5.3 Primary Plane

Many of the plane control active registers will be updated on the vertical blank or when pipe is disabled, after the surface base address register is written, or when the plane is not yet enabled – thus providing an atomic update of those registers together with the surface base address register.

5.3.1 PRI_CTL-Primary Control

PRI_CTL

Register Space:	MMIO: 0/2/0
Project:	
Default Value:	0x00000000
Access:	R/W
Size (in bits):	32
Double Buffer Update Point:	Start of vertical blank or pipe disabled or primary disabled, after armed
Double Buffer Armed By:	Write to PRI_SURF

Address:	70180h-70183h
Name:	Primary A Control
ShortName:	PRI_CTL_A

Address:	71180h-71183h
Name:	Primary B Control
ShortName:	PRI_CTL_B

Address:	72180h-72183h
Name:	Primary C Control
ShortName:	PRI_CTL_C

DWord	Bit	Description
0	31	**Primary Plane Enable** Format: Enable When this bit is set, the primary plane will generate pixels for display. When set to zero, primary plane memory fetches cease and plane output is transparent. When in Self Refresh Big FIFO mode, a write to enable the plane will be internally buffered and delayed while Big FIFO mode is exiting.
	30	**Gamma Enable** This bit enables pipe gamma correction for the plane pixel data. For 8-bit indexed display data, this bit should be set to a one. Value / Name / Description 0b / Disable **[Default]** / Plane pixel data bypasses pipe gamma correction 1b / Enable / Plane pixel data passes through pipe gamma correction
	29:26	**Source Pixel Format** This field selects the source pixel format for the primary plane. The 8-bpp indexed format will use the pipe palette. Before entering the blender, each source format is converted to the pipe pixel format. Alpha values are ignored. Value / Name / Description 0010b / 8-bit Indexed / 8-bit Indexed 0101b / 16-bit BGRX 5:6:5 / 16-bit BGRX (5:6:5 MSB-R:G:B)

PRI_CTL

	0110b	32-bit BGRX 8:8:8	32-bit BGRX (8:8:8:8 MSB-X:R:G:B)
	1000b	32-bit RGBX 10:10:10	32-bit RGBX (2:10:10:10 MSB-X:B:G:R)
	1001b	32-bit XR_BIAS RGBX 10:10:10	32-bit Extended Range Bias RGBX (2:10:10:10 MSB-X:B:G:R)
	1010b	32-bit BGRX 10:10:10	32-bit BGRX (2:10:10:10 MSB-X:R:G:B)
	1100b	64-bit RGBX FP	64-bit RGBX Floating Point(16:16:16:16 MSB-X:B:G:R)
	1110b	32-bit RGBX 8:8:8	32-bit RGBX (8:8:8:8 MSB-X:B:G:R)
	Others	Reserved	Reserved

Programming Notes

Workaround (WaFP16GammaEnabling) : When using the 64-bit format, the plane output on each color channel has one quarter amplitude. It can be brought up to full amplitude by using pipe gamma correction or pipe color space conversion to multiply the plane output by four.

25 **Reserved**

24 **Pipe CSC Enable**

This bit enables pipe color space conversion for the plane pixel data.

Value	Name	Description
0b	Disable **[Default]**	Plane pixel data bypasses the pipe color space conversion
1b	Enable	Plane pixel data passes through the pipe color space conversion

23:16 **Reserved**

15 **180 Display Rotation**

This mode causes the plane image to be rotated 180 degrees.
In addition to setting this bit, software must also set the surface address offset (lineary or tiled offset registers depending on tiled surface select) to the lower right corner of the unrotated image.

Value	Name	Description
0b	None **[Default]**	No rotation
1b	180	180 degree rotation

14 **Trickle Feed Enable**

Value	Name	Description
0b	Enable **[Default]**	Trickle Feed Enabled - Plane data requests are sent whenever there is space in the Display Data Buffer.
1b	Disable	Trickle Feed Disabled - Plane data requests are sent in bursts

13:11 **Reserved**

10 **Tiled Surface**

This bit indicates that the surface data is in tiled memory. The tile pitch is specified in bytes in the plane stride register. Only X tiling is supported.
When this bit is set, it affects the interpretation of the offset and surface address registers.
This bit may be updated through MMIO writes or through a command streamer initiated synchronous flip.

Value	Name	Description
0b	Linear **[Default]**	Plane uses linear memory
1b	X-Tiled	Planes uses X-Tiled memory

9 **Async Address Update Enable**

This bit will enable asynchronous updates of the plane surface address when written by MMIO (MMIO asynchronous flips).
The surface address will change with the next plane TLB request or when start of vertical blank is reached.
Updates during vertical blank may not complete until after the first few active lines are displayed.
This bit is not double-buffered, so it will update immediately after being written.

PRI_CTL

Value	Name	Description
0b	Sync **[Default]**	Surface Address MMIO writes will update synchronous to start of vertical blank (synchronous flips)
1b	Async	Surface Address MMIO writes will update asynchronously (asynchronous flips)

Programming Notes

Restriction : No command streamer initiated flips to this plane are allowed when this bit is enabled.
Each surface address write must be followed by a wait for flip done indication before writing the surface address register again.
Linear memory does not support async updates.

8:0 **Reserved**

Format: MBZ

Plane Source Pixel Format Mapping of Bits to Colors:

Format	Ignored	Red	Green	Blue
16-bit BGRX 5:6:5	N/A	15:11	10:5	4:0
32-bit BGRX 8:8:8	31:24	23:16	15:8	7:0
32-bit RGBX 10:10:10	31:30	9:0	19:10	29:20
32-bit BGRX 10:10:10	31:30	29:20	19:10	9:0
64-bit RGBX Float 16:16:16 Each component is 1:5:10 MSb-sign:exponent:fraction	63:48	15:0	31:16	47:32
32-bit RGBX 8:8:8	31:24	7:0	15:8	23:16
32-bit XR_BIAS RGBX 10:10:10	31:30	9:0	19:10	29:20

5.3.2 PRI_LINOFF-Primary Linear Offset

PRI_LINOFF	
Register Space:	MMIO: 0/2/0
Project:	
Default Value:	0x00000000
Access:	R/W
Size (in bits):	32
Double Buffer Update Point:	Start of vertical blank or pipe disabled
Address:	70184h-70187h
Name:	Primary A Linear Offset
ShortName:	PRI_LINOFF_A
Address:	71184h-71187h
Name:	Primary B Linear Offset
ShortName:	PRI_LINOFF_B
Address:	72184h-72187h
Name:	Primary C Linear Offset
ShortName:	PRI_LINOFF_C

DWord	Bit	Description
0	31:0	**Linear Offset** This register specifies the panning for the plane surface in linear memory. This value is added to the surface address to get the address of the first pixel to be displayed. This offset must be at least pixel aligned for RGB formats. When performing 180 degree rotation, the unpanned offset must be the difference between the last pixel of the last line of the display data in its unrotated orientation and the display surface address. When the surface is tiled, the tiled offset is programmed and the contents of this register are ignored.

5.3.3 PRI_STRIDE-Primary Stride

PRI_STRIDE	
Register Space:	MMIO: 0/2/0
Project:	
Default Value:	0x00000000
Access:	R/W
Size (in bits):	32
Double Buffer Update Point:	Start of vertical blank or pipe disabled or primary disabled, after armed
Double Buffer Armed By:	Write to PRI_SURF
Address:	70188h-7018Bh
Name:	Primary A Stride
ShortName:	PRI_STRIDE_A
Address:	71188h-7118Bh
Name:	Primary B Stride
ShortName:	PRI_STRIDE_B
Address:	72188h-7218Bh
Name:	Primary C Stride
ShortName:	PRI_STRIDE_C

DWord	Bit	Description
0	31:16	**Reserved**
	15:6	**Stride** This is the stride for the plane in bytes. This value is used to determine the line to line increment for the plane. When using linear memory, this must be at least 64 byte aligned. When using tiled memory, this must be at least 512 byte aligned. This register may be updated through MMIO writes or through a command streamer initiated synchronous flip. The stride is limited to a maximum of 32K bytes.
	5:0	**Reserved** Format: MBZ

5.3.4 PRI_SURF-Primary Surface Base Address

PRI_SURF	
Register Space:	MMIO: 0/2/0
Project:	
Default Value:	0x00000000
Access:	R/W
Size (in bits):	32
Double Buffer Update Point:	Start of vertical blank, pipe disabled, or next plane line request if asynchronous flip
Address:	7019Ch-7019Fh
Name:	Primary A Base Address
ShortName:	PRI_SURF_A
Address:	7119Ch-7119Fh
Name:	Primary B Base Address
ShortName:	PRI_SURF_B
Address:	7219Ch-7219Fh
Name:	Primary C Base Address
ShortName:	PRI_SURF_C

Writes to this register arm primary registers for this pipe

DWord	Bit	Description
0	31:12	**Surface Base Address** Format: GraphicsAddress[31:12] This address specifies the surface base address. It represents an offset from the graphics memory aperture base and is mapped to physical pages through the global GTT. It must be at least 4KB aligned. When performing asynchronous flips and the display surface is in tiled memory, this address must be 256KB aligned. This register may be updated through MMIO writes or through a command streamer initiated synchronous or asynchronous flip. **Programming Notes** Workaround : To prevent false VT-d type 6 errors, use 256KB address alignment and allocate an extra 128 Page Table Entries (PTEs) beyond the end of the displayed surface. Only the PTEs will be used, not the pages themselves.
	11:3	**Reserved**
	1:0	**Reserved**

5.3.5 PRI_TILEOFF-Primary Tiled Offset

PRI_TILEOFF	
Register Space:	MMIO: 0/2/0
Project:	
Default Value:	0x00000000
Access:	R/W
Size (in bits):	32
Double Buffer Update Point:	Start of vertical blank or pipe disabled
Address:	701A4h-701A7h
Name:	Primary A Tiled Offset
ShortName:	PRI_TILEOFF_A
Address:	711A4h-711A7h
Name:	Primary B Tiled Offset
ShortName:	PRI_TILEOFF_B
Address:	721A4h-721A7h
Name:	Primary C Tiled Offset
ShortName:	PRI_TILEOFF_C

This register specifies the panning for the plane surface in tiled memory.
When the surface is in linear memory, the linear offset is programmed and the contents of this register are ignored.
When the surface is tiled, the start position is specified in this register as a (x, y) offset from the beginning of the surface.
When performing 180 degree rotation, the unpanned offset must be programmed to the last pixel of the last line of the display data.

DWord	Bit	Description
0	31:28	**Reserved** Format: MBZ
	27:16	**Start Y Position** The vertical offset in lines of the beginning of the active display plane relative to the display surface.
	15:12	**Reserved** Format: MBZ
	11:0	**Start X Position** The horizontal offset in pixels of the beginning of the active display plane relative to the display surface.

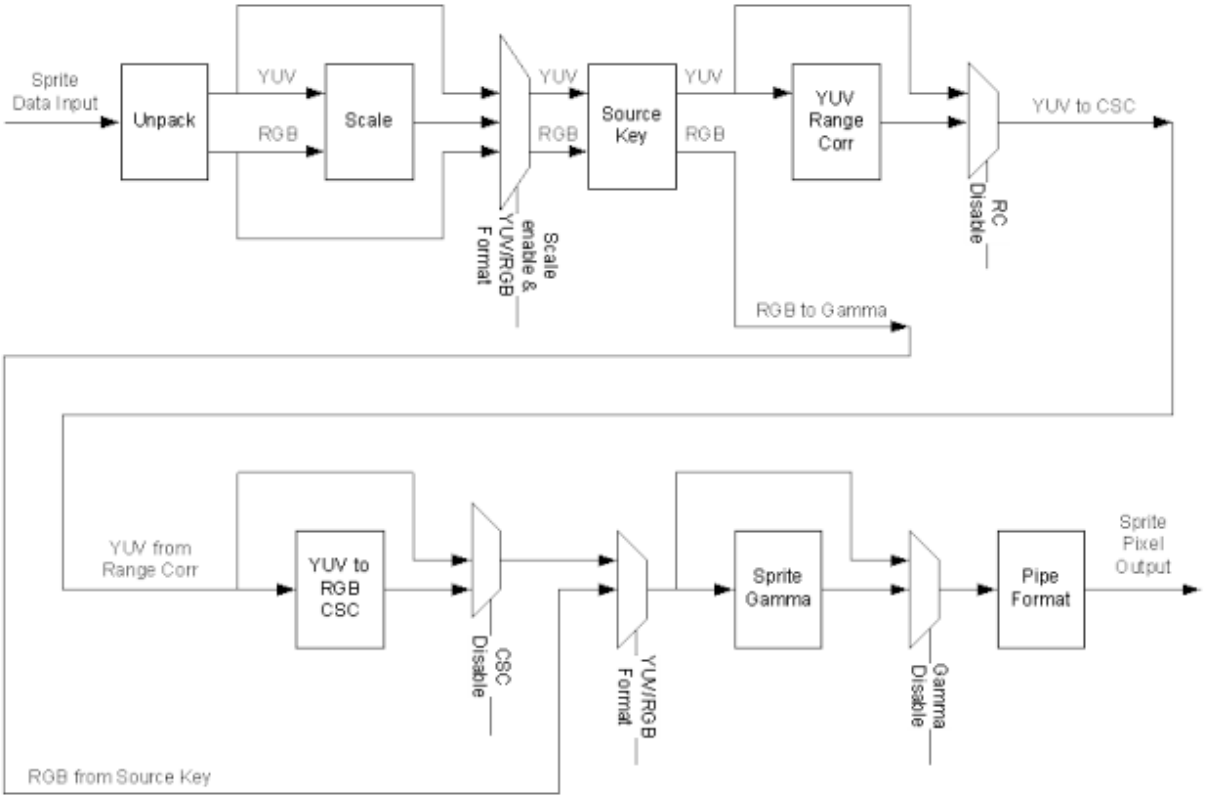
5.4 Sprite Plane

Many of the plane control active registers will be updated on the vertical blank or when pipe is disabled, after the surface base address register is written, or when the plane is not yet enabled – thus providing an atomic update of those registers together with the surface base address register.

Data flow through the sprite plane (Steps 2-6 may be enabled or disabled by programming control bits):

1. Unpack data into pixels
2. Scale

3. Source Key
4. YUV Range Correction (can only be used by YUV source pixel formats)
5. YUV to RGB Color Space Conversion (can only be used by YUV source pixel formats)
6. Sprite Gamma Correction
7. Conversion to pipe data format



5.4.1 SPR_CTL-Sprite Control

SPR_CTL	
Register Space:	MMIO: 0/2/0
Project:	
Default Value:	0x00000000
Access:	R/W
Size (in bits):	32
Double Buffer Update Point:	Start of vertical blank or pipe disabled or sprite disabled, after armed
Double Buffer Armed By:	Write to SPR_SURF
Address:	70280h-70283h
Name:	Sprite A Control
ShortName:	SPR_CTL_A
Address:	71280h-71283h
Name:	Sprite B Control
ShortName:	SPR_CTL_B
Address:	72280h-72283h
Name:	Sprite C Control
ShortName:	SPR_CTL_C

DWord	Bit	Description
0	31	**Sprite Enable** Format: Enable When this bit is set, the sprite plane will generate pixels for display. When set to zero, sprite plane memory fetches cease and plane output is transparent. When in Self Refresh Big FIFO mode, a write to enable the plane will be internally buffered and delayed while Big FIFO mode is exiting.
	30	**Pipe Gamma Enable** This bit enables pipe gamma correction for the sprite pixel data. Value / Name / Description 0b / Disable **[Default]** / Plane pixel data bypasses pipe gamma correction 1b / Enable / Plane pixel data passes through pipe gamma correctopm
	29	**Reserved** Format: MBZ
	28	**YUV Range Correction Disable** Setting this bit disables the YUV range correction logic inside the sprite. The range correction logic is used to expand the compressed range YUV to full range YUV. The Y channel is expanded from the 8 bit +16 to +235 range to full range. The U and V channels are expanded from the 8 bit -112 to +112 range to full range. Extended range values will be preserved after the expansion. This bit has no effect on RGB source pixel formats since they automatically bypass range correction. Value / Name / Description 0b / Enable **[Default]** / Range correction enabled 1b / Disable / No range correction
	27:25	**Source Pixel Format** This field selects the source pixel format for the sprite plane.

SPR_CTL

Before entering the blender, each source format is converted to the pipe pixel format.
Alpha values are ignored.
YUV 4:2:2 byte order is programmed separately.
YUV 4:4:4 byte order is not programmable.
RGB color order is programmed separately, except RGB XR_BIAS byte order is not programmable.

Value	Name	Description
000b	YUV 16-bit 4:2:2	YUV 16-bit 4:2:2 packed
001b	RGB 32-bit 2:10:10:10	RGB 32-bit 2:10:10:10
010b	RGB 32-bit 8:8:8:8	RGB 32-bit 8:8:8:8
011b	RGB 64-bit 16:16:16:16	RGB 64-bit 16:16:16:16 Floating Point
100b	YUV 32-bit 4:4:4	YUV 32-bit 4:4:4 packed (8:8:8:8 MSB-X:Y:U:V)
101b	RGB 32-bit XR_BIAS 10:10:10	RGB 32-bit Extended Range Bias RGBX (2:10:10:10 MSB-X:B:G:R)
Others	Reserved	Reserved

Programming Notes

Workaround (WaFP16GammaEnabling) : When using the 64-bit format, the sprite output on each color channel has one quarter amplitude. It can be brought up to full amplitude by using sprite internal gamma correction, pipe gamma correction, or pipe color space conversion to multiply the sprite output by four.

Workaround (DisableSpritePassThroughMode) : When using YUV formats and the sprite internal CSC is disabled, the sprite output will not have a 1/2 offset on the U and V channels. An offset on U and V channels is typically required by receivers. It can be added using the pipe CSC.

If pipe CSC is already in use for RGB to YUV conversion, then the sprite internal CSC can be used to convert the sprite output to RGB, and the sprite can be sent through the pipe CSC, programmed for RGB to YUV conversion with pipe CSC post-offset of +1/2.

If pipe CSC is not already in use, then the sprite output can be kept as YUV, and the sprite can be sent through the pipe CSC, programmed for 1:1 pass through with pipe CSC post-offset of +1/2.

24 **Pipe CSC Enable**

This bit enables pipe color space conversion for the plane pixel data. This is separate from the color conversion logic within the sprite plane.

Value	Name	Description
0b	Disable **[Default]**	Plane pixel data bypasses the pipe color space conversion
1b	Enable	Plane pixel data passes through the pipe color space conversion

23 **Reserved**

Format:	MBZ

22 **Sprite Source Key Enable**

This bit enables source color keying. Sprite pixel values that match (within range) the key will become transparent. Source key can not be enabled if destination key is enabled.

Value	Name	Description
0b	Disable **[Default]**	Sprite source key is disabled
1b	Enable	Sprite source key is enabled

21 **Reserved**

Format:	MBZ

20 **RGB Color Order**

This field is used to select the color order when using RGB data formats, except RGB 32-bit XR_BIAS 10:10:10. For other formats, this field is ignored.

SPR_CTL

Value	Name	Description
0b	BGRX **[Default]**	BGRX (MSB-X:R:G:B)
1b	RGBX	RGBX (MSB-X:B:G:R)

19 **Sprite YUV to RGB CSC Dis**

This bit controls the sprite internal YUV to RGB color space conversion. RGB source pixel formats automatically bypass the sprite internal color space conversion.

Value	Name	Description
0b	Enable **[Default]**	YUV pixel data goes through the sprite color conversion
1b	Disable	YUV pixel data bypasses the sprite color conversion

18 **Sprite YUV to RGB CSC Format**

This bit specifies the source YUV format for the sprite internal YUV to RGB color space conversion operation. This field is ignored when source data is RGB.

Value	Name	Description
0b	BT.601 **[Default]**	ITU-R Recommendation BT.601
1b	BT.709	ITU-R Recommendation BT.709

17:16 **YUV 422 Byte Order**

This field is used to select the byte order when using YUV 4:2:2 data formats. For other formats, this field is ignored.

Value	Name	Description
00b	YUYV **[Default]**	YUYV (8:8:8:8 MSB-V:Y2:U:Y1)
01b	UYVY	UYVY (8:8:8:8 MSB-Y2:V:Y1:U)
10b	YVYU	YVYU (8:8:8:8 MSB-U:Y2:V:Y1)
11b	VYUY	VYUY (8:8:8:8 MSB-Y2:U:Y1:V)

15 **180 Display Rotation**

This mode causes the plane image to be rotated 180 degrees.
In addition to setting this bit, software must also set the surface address offset (linear or tiled offset registers depending on tiled surface select) to the lower right corner of the unrotated and unscaled surface image and adjust the plane position to match the physical orientation of the display.

Value	Name	Description
0b	None **[Default]**	No rotation
1b	180	180 degree rotation

14 **Trickle Feed Enable**

Value	Name	Description
0b	Enable **[Default]**	Trickle Feed Enabled - Data requests are sent whenever there is space in the Display Data Buffer
1b	Disable	Trickle Feed Disabled - Data requests are sent in bursts.

13 **Sprite Gamma Disable**

This bit controls sprite internal gamma correction.

Value	Name	Description
1b	Disable	Disable sprite internal gamma correction
0b	Enable **[Default]**	Enable sprite internal gamma correction

12:11 **Reserved**

10 **Tiled Surface**

This bit indicates that the surface data is in tiled memory.
The tile pitch is specified in bytes in the plane stride register.
Only X tiling is supported.
When this bit is set, it affects the interpretation of the offset and surface address registers.
This bit may be updated through MMIO writes or through a command streamer initiated synchronous flip.

SPR_CTL				
		Value	**Name**	**Description**
		0b	Linear **[Default]**	Plane uses linear memory
		1b	X-Tiled	Planes uses X-Tiled memory
	9:3	**Reserved**		
		Format:		MBZ
	2	**Sprite Destination Key** This bit enables the destination key function. When blending together sprite and primary planes, if the primary plane pixel matches the key value, then the sprite pixel is output, otherwise the primary pixel is output. Destination Key can not be enabled if source key is enabled.		
		Value	**Name**	**Description**
		0b	Disable **[Default]**	Destination Key is disabled
		1b	Enable	Destination Key is enabled
	1:0	**Reserved**		
		Format:		MBZ

Sprite Source Pixel Format Mapping of Bits to Colors:

Note: For RGB formats, see the primary plane source pixel format mapping table

SPRITE YUV 4:2:2	**Y1**	**U**	**Y2**	**V**
YUV 4:2:2 YUYV	7:0	15:8	23:16	31:24
YUV 4:2:2 UYVY	15:8	7:0	31:24	23:16
YUV 4:2:2 YVYU	7:0	31:24	23:16	15:8
YUV 4:2:2 VYUY	15:8	23:16	31:24	7:0
SPRITE YUV 4:4:4	**Ignored**	**Y**	**U**	**V**
YUV 32-bit 4:4:4	31:24	23:16	15:8	7:0

5.4.2 SPR_LINOFF-Sprite Linear Offset

SPR_LINOFF	
Register Space:	MMIO: 0/2/0
Project:	
Default Value:	0x00000000
Access:	R/W
Size (in bits):	32
Double Buffer Update Point:	Start of vertical blank or pipe disabled
Address:	70284h-70287h
Name:	Sprite A Linear Offset
ShortName:	SPR_LINOFF_A
Address:	71284h-71287h
Name:	Sprite B Linear Offset
ShortName:	SPR_LINOFF_B

SPR_LINOFF		
Address:	72284h-72287h	
Name:	Sprite C Linear Offset	
ShortName:	SPR_LINOFF_C	
DWord	**Bit**	**Description**
0	31:0	**Linear Offset** This register specifies the panning for the plane surface in linear memory. This value is added to the surface address to get the address of the first pixel to be displayed. This offset must be at least pixel aligned for unrotated RGB or YUV 4:4:4 formats and even pixel aligned for unrotated YUV 4:2:2 formats. When performing 180 degree rotation, the unpanned offset must be the difference between the last pixel of the last line of the display data in its unrotated orientation and the display surface address. When the surface is tiled, the tiled offset is programmed and the contents of this register are ignored. When using sprite scaling the offset is done on the source pixels.

5.4.3 SPR_STRIDE-Sprite Stride

SPR_STRIDE		
Register Space:	MMIO: 0/2/0	
Project:		
Default Value:	0x00000000	
Access:	R/W	
Size (in bits):	32	
Double Buffer Update Point:	Start of vertical blank or pipe disabled or sprite disabled, after armed	
Double Buffer Armed By:	Write to SPR_SURF	
Address:	70288h-7028Bh	
Name:	Sprite A Stride	
ShortName:	SPR_STRIDE_A	
Address:	71288h-7128Bh	
Name:	Sprite B Stride	
ShortName:	SPR_STRIDE_B	
Address:	72288h-7228Bh	
Name:	Sprite C Stride	
ShortName:	SPR_STRIDE_C	
DWord	**Bit**	**Description**
0	31:15	**Reserved**
	14:6	**Stride** This is the stride for the plane in bytes. This value is used to determine the line to line increment for the plane. When using linear memory, this must be at least 64 byte aligned. When using tiled memory, this must be at least 512 byte aligned. This register may be updated through MMIO writes or through a command streamer initiated synchronous flip. The stride is limited to a maximum of 16K bytes when sprite scaling is not enabled, 4K bytes when sprite scaling is enabled.
	5:0	**Reserved**

5.4.4 SPR_POS-Sprite Position

SPR_POS	
Register Space:	MMIO: 0/2/0
Project:	
Default Value:	0x00000000
Access:	R/W
Size (in bits):	32
Double Buffer Update Point:	Start of vertical blank or pipe disabled or sprite disabled, after armed
Double Buffer Armed By:	Write to SPR_SURF
Address:	7028Ch-7028Fh
Name:	Sprite A Position
ShortName:	SPR_POS_A
Address:	7128Ch-7128Fh
Name:	Sprite B Position
ShortName:	SPR_POS_B
Address:	7228Ch-7228Fh
Name:	Sprite C Position
ShortName:	SPR_POS_C

This register specifies the screen position of the sprite. The sprite must be completely contained within the pipe source area. Pipe source size >= sprite position + sprite size The origin of the sprite position is always the upper left corner of the display pipe source image area. When performing 180 degree rotation, the sprite image is rotated by hardware, but the position is not, so it must be adjusted by software if it is desired to maintain the same apparent position on a physically rotated display.

DWord	Bit	Description
0	31:28	**Reserved** Format: MBZ
	27:16	**Y Position** This specifies the vertical position of the sprite upper left corner in lines.
	15:12	**Reserved** Format: MBZ
	11:0	**X Position** This specifies the horizontal position of the sprite upper left corner in pixels.

5.4.5 SPR_SIZE-Sprite Size

SPR_SIZE		
Register Space:	MMIO: 0/2/0	
Project:		
Default Value:	0x00000000	
Access:	R/W	
Size (in bits):	32	
Double Buffer Update Point:	Start of vertical blank or pipe disabled or sprite disabled, after armed	
Double Buffer Armed By:	Write to SPR_SURF	
Address:	70290h-70293h	
Name:	Sprite A Size	
ShortName:	SPR_SIZE_A	
Address:	71290h-71293h	
Name:	Sprite B Size	
ShortName:	SPR_SIZE_B	
Address:	72290h-72293h	
Name:	Sprite C Size	
ShortName:	SPR_SIZE_C	

This register specifies the size of the sprite. The sprite must be completely contained within the pipe source area. Pipe source size >= sprite position + sprite sizeThe sprite must be at least one pixel high and one pixel wide.

DWord	Bit	Description
0	31:28	**Reserved** Format: MBZ
	27:16	**Height** This specifies the height of the sprite in lines. The value in the register is the height minus one.
	15:12	**Reserved** Format: MBZ
	11:0	**Width** This specifies the width of the sprite in pixels. The value in the register is the width minus one. This should be less than or equal to the stride in pixels. The width (prior to minus one) must be even when a YUV 4:2:2 source pixel format is used.

5.4.6 SPR_SURF-Sprite Surface Base Address

SPR_SURF	
Register Space:	MMIO: 0/2/0
Project:	
Default Value:	0x00000000
Access:	R/W
Size (in bits):	32
Double Buffer Update Point:	Start of vertical blank or pipe disabled
Address:	7029Ch-7029Fh
Name:	Sprite A Surface Base Address
ShortName:	SPR_SURF_A
Address:	7129Ch-7129Fh
Name:	Sprite B Surface Base Address
ShortName:	SPR_SURF_B
Address:	7229Ch-7229Fh
Name:	Sprite C Surface Base Address
ShortName:	SPR_SURF_C

Writes to this register arm sprite registers for this pipe

DWord	Bit	Description
0	31:12	**Surface Base Address** Format: Graphicsdress[31:12] This address specifies the surface base address. It represents an offset from the graphics memory aperture base and is mapped to physical pages through the global GTT. It must be at least 4KB aligned. This register may be updated through MMIO writes or through a command streamer initiated synchronous flip. **Programming Notes** Workaround : To prevent false VT-d type 6 errors, use 128KB address alignment and allocate an extra 64 Page Table Entries (PTEs) beyond the end of the displayed surface. Only the PTEs will be used, not the pages themselves.
	11:3	**Reserved**
	1:0	**Reserved**

5.4.7 SPR_TILEOFF-Sprite Tiled Offset

SPR_TILEOFF	
Register Space:	MMIO: 0/2/0
Project:	
Default Value:	0x00000000
Access:	R/W
Size (in bits):	32
Double Buffer Update Point:	Start of vertical blank or pipe disabled
Address:	702A4h-702A7h
Name:	Sprite A Tiled Offset
ShortName:	SPR_TILEOFF_A
Address:	712A4h-712A7h
Name:	Sprite B Tiled Offset
ShortName:	SPR_TILEOFF_B
Address:	722A4h-722A7h
Name:	Sprite C Tiled Offset
ShortName:	SPR_TILEOFF_C

This register specifies the panning for the plane surface in tiled memory.
When the surface is in linear memory, the linear offset is programmed and the contents of this register are ignored.
When the surface is tiled, the start position is specified in this register as a (x, y) offset from the beginning of the surface.
When performing 180 degree rotation, the unpanned offset must be programmed to the last pixel of the last line of the display data.
When using sprite scaling the offset is done on the source pixels.
This offset must be even pixel aligned for unrotated YUV 4:2:2 formats.

DWord	Bit	Description
0	31:28	**Reserved** Format: MBZ
	27:16	**Start Y Position** The vertical offset in lines of the beginning of the active display plane relative to the display surface.
	15:12	**Reserved** Format: MBZ
	11:0	**Start X Position** The horizontal offset in pixels of the beginning of the active display plane relative to the display surface.

5.4.8 SPR_KEYVAL-Sprite Key Color Value

SPR_KEYVAL	
Register Space:	MMIO: 0/2/0
Project:	
Default Value:	0x00000000
Access:	R/W
Size (in bits):	32
Double Buffer Update Point:	Start of vertical blank or pipe disabled
Address:	70294h-70297h
Name:	Sprite A Key Color Value
ShortName:	SPR_KEYVAL_A
Address:	71294h-71297h
Name:	Sprite B Key Color Value
ShortName:	SPR_KEYVAL_B
Address:	72294h-72297h
Name:	Sprite C Key Color Value
ShortName:	SPR_KEYVAL_C

For source key when sprite source is YUV, this register specifies the source key YUV minimum color value to be used together with the YUV maximum color value and the color channel enable bits to determine if the sprite matches the source key color range.
For source key when sprite source is RGB, this register specifies the source key RGB color value to be used together with the color channel enable bits to determine if the sprite matches the source key color.
For destination key, this register specifies the destination key RGB color value to be used together with the RGB mask bits to determine if the primary matches the destination key color.
A key match can only occur for positive pixel values in the 0 to 1 range. Extended range pixel values will not match.

DWord	Bit	Description
0	31:24	**Reserved** Format: MBZ
	23:16	**V R Min Dest Key Value** Specifies the color key minimum value for the sprite V channel source key, the compare value for sprite Red channel source key, or the compare value for the primary Red channel destination key.
	15:8	**Y G Min Dest Key Value** Specifies the color key minimum value for the sprite Y channel source key, the compare value for sprite Green channel source key, or the compare value for the primary Green channel destination key.
	7:0	**U B Min Dest Key Value** Specifies the color key minimum value for the sprite U channel source key, the compare value for sprite Blue channel source key, or the compare value for the primary Blue channel destination key.

5.4.9 SPR_KEYMSK-Sprite Key Mask

SPR_KEYMSK	
Register Space:	MMIO: 0/2/0
Project:	
Default Value:	0x00000000
Access:	R/W
Size (in bits):	32
Double Buffer Update Point:	Start of vertical blank or pipe disabled
Address:	70298h-7029Bh
Name:	Sprite A Key Mask
ShortName:	SPR_KEYMSK_A
Address:	71298h-7129Bh
Name:	Sprite B Key Mask
ShortName:	SPR_KEYMSK_B
Address:	72298h-7229Bh
Name:	Sprite C Key Mask
ShortName:	SPR_KEYMSK_C

For source key, this register specifies which channels to perform key color checking on. A channel that is not enabled for key comparison will always match on the full range of values.
For destination key, this register specifies the key mask to be used with the color value bits to determine if the primary plane pixels match the key. A zero bit in the mask indicates that the corresponding bit match failure should be ignored when determining if the pixel matches.

Programming Notes

Restriction : Source key and destination key are mutually exclusive modes of operation, they can not be used simultaneously. For the function that is not enabled, the associated bits in this register should be programmed to zeroes.

DWord	Bit	Description
0	31:27	**Reserved** Format: MBZ
	26	**V R Source Key Channel Enable** Enables the V/Red channel for source key color comparison.
	25	**Y G Source Key Channel Enable** Enables the Y/Green channel for source key color comparison.
	24	**U B Source Key Channel Enable** Enables the U/Blue channel for source key color comparison.
	23:16	**R Dest Key Mask Value** Specifies the destination color key mask for the Red channel
	15:8	**G Dest Key Mask Value** Specifies the destination color key mask for the Green channel
	7:0	**B Dest Key Mask Value** Specifies the destination color key mask for the Blue channel

5.4.10 SPR_KEYMAX-Sprite Key Color Max

SPR_KEYMAX	
Register Space:	MMIO: 0/2/0
Project:	
Default Value:	0x00000000
Access:	R/W
Size (in bits):	32
Double Buffer Update Point:	Start of vertical blank or pipe disabled
Address:	702A0h-702A3h
Name:	Sprite A Key Color Max
ShortName:	SPR_KEYMAX_A
Address:	712A0h-712A3h
Name:	Sprite B Key Color Max
ShortName:	SPR_KEYMAX_B
Address:	722A0h-722A3h
Name:	Sprite C Key Color Max
ShortName:	SPR_KEYMAX_C

For source key when sprite source is YUV, this register specifies the source key YUV maximum color value to be used together with the YUV minimum color value and the color channel enable bits to determine if the sprite matches the source key color range.

DWord	Bit	Description
0	31:24	**Reserved** Format: MBZ
	23:16	**V Source Key Max Value** Specifies the color key maximum value for the sprite V channel source key
	15:8	**Y Source Key Max Value** Specifies the color key maximum value for the sprite Y channel source key
	7:0	**U Source Key Max Value** Specifies the color key maximum value for the sprite U channel source key

5.4.11 SPR_SCALE-Sprite Scaler Control

SPR_SCALE	
Register Space:	MMIO: 0/2/0
Project:	
Default Value:	0x00000000
Access:	R/W
Size (in bits):	32
Double Buffer Update Point:	Start of vertical blank or pipe disabled or sprite disabled, after armed
Double Buffer Armed By:	Write to SPR_SURF
Address:	70304h-70307h
Name:	Sprite A Scaler Control
ShortName:	SPR_SCALE_A
Address:	71304h-71307h
Name:	Sprite B Scaler Control
ShortName:	SPR_SCALE_B
Address:	72304h-72307h
Name:	Sprite C Scaler Control
ShortName:	SPR_SCALE_C

This register controls the sprite scaling.
When scaling is enabled, the SPR_SIZE register gives the destination (output to pipe) size of the sprite and this register gives the source (input to sprite) size of the sprite, then the source size will be scaled up or down to the destination size.

Programming Notes

Restriction : Sprite scaling should not be enabled with the RGB XR_BIAS 10:10:10 format, RGB 64-bit format, or any YUV format containing extended range data.
Source and destination sizes must be 3x3 (3x6 when interlacing) or greater when scaling is enabled.
Upscaling of any amount is allowed.
Downscaling less than 2X (source/destination) is allowed.
Downscaling greater than or equal to 2X is not supported.
Downscaling increases memory bandwidth requirements.
Horizontal downscaling limits the maximum pixel rate. See the section on Display Pixel Rate Limitations.

DWord	Bit	Description
0	31	**Scaling Enable** Format: Enable This field enables the scaling function. Source width can be no more than 4k bytes with scaling enabled. For best picture quality, disable when scaling is not required. Value: 0b — Name: Disable **[Default]** Value: 1b — Name: Enable **Programming Notes** Restriction : Scaling should not be left enabled when sprite is disabled. Workaround (WaCxSRDisabledForSpriteScaling) : Low Power watermarks must be disabled for at least one frame before enabling sprite scaling, and kept disabled until sprite scaling is disabled.

SPR_SCALE

Suggested sequence:
0. Driver notified that scaling will be enabled
1. Disable WM_LP (write WM_LP3 bit 31=0 first, then WM_LP2, then WM_LP1)
2. Wait for vertical blank
3. Enable scaling
....
4. Driver notified that scaling will not be enabled
5. Disable scaling
6. Restore WM_LP (restore WM_LP1 first, then WM_LP2, then WM_LP3)

30:29 **Filter Control**
Filter selection

Value	Name	Description
00b	Medium	Medium Filtering
01b	Enhancing	Edge Enhancing Filtering
10b	Softening	Edge Softening Filtering
11b	Reserved	Reserved

28 **Field Offset**
Select the vertical offset of the filtered data. Software is responsible for updating this to match the surface data.

Value	Name	Description
0b	0 **[Default]**	Vertical initial phase of 0
1b	0.5	Vertical initial phase of 0.5

27 **Field Enable**
Enable adjustment of the vertical offset of the filtered data.

Value	Name	Description
0b	Disable **[Default]**	Off (Vertical initial phase is 1/2 the scale factor)
1b	Enable	On (Vertical initial phase is selected by the Field Offset bit)

26:16 **Source Width**
The horizontal size of the source image to be scaled in pixels. Max number of pixels is 2048; minimum is 3. The value programmed is one less than the number of pixels. Source width can be no more than 4k bytes, counting from a 64 byte alignment. The sprite width (actual width, not the width minus one value) is limited to even values when YUV 4:2:2 source pixel format is used.

15:11 **Reserved**

Format:	MBZ

10:0 **Source Height**
The vertical size of the source image to be scaled in lines. If the source is a field, this is the number of lines in the field. Max number of lines is 2048; minimum is 3 (6 when interlacing). The value programmed is one less than the number of lines. The height must be even when sprite scaling is enabled and the pipe has set planes to interlaced fetch. That means the programmed value must be odd.

5.4.12 SPR_GAMC-Sprite Gamma Correction

SPR_GAMC REFERENCE POINT FORMAT

Project:	
Size (in bits):	30
Default Value:	0x00000000

This format is used to determine the first 16 reference points (points 0 to 15) for sprite gamma correction.
The values are represented in an unsigned 0.10 format with 0 integer and 10 fractional bits.
See SPR_GAMC for sprite gamma programming information.

DWord	Bit	Description
0	29:20	**Red Gamma Reference Point** Format: U0.10 This value specifies a reference point that is used for the red color channel sprite gamma correction.
	19:10	**Green Gamma Reference Point** Format: U0.10 This value specifies a reference point that is used for the green color channel sprite gamma correction.
	9:0	**Blue Gamma Reference Point** Format: U0.10 This value specifies a reference point that is used for the blue color channel sprite gamma correction.

SPR_GAMC

Register Space:	MMIO: 0/2/0
Project:	
Default Value:	0x00000000, 0x00000000, 0x00000000, 0x00000000, 0x00000000, 0x00000000, 0x00000000, 0x00000000, 0x00000000, 0x00000000, 0x00000000, 0x00000000, 0x00000000, 0x00000000, 0x00000000, 0x00000000
Access:	R/W
Size (in bits):	16x32

Address:	70400h-7043Fh
Name:	Sprite A Gamma Correction
ShortName:	SPR_GAMC_[0-15]_A

Address:	71400h-7143Fh
Name:	Sprite B Gamma Correction
ShortName:	SPR_GAMC_[0-15]_B

Address:	72400h-7243Fh
Name:	Sprite C Gamma Correction
ShortName:	SPR_GAMC_[0-15]_C

These registers are used to determine the characteristics of the gamma correction for the sprite pixel data pre-blending. Additional gamma correction can be done in the display pipe gamma if desired.

SPR_GAMC

The gamma correction curve is represented by specifying a set of gamma entry reference points spaced equally along the curve for values between -1 and 1. For extended values there is an extended gamma entry reference point at the maximum alowed input value.

All input values are clamped to the greater than -3.0 and less than 3.0 range before the gamma calculation.

* For input values greater than or equal to 0 and less than 1.0, the input value is used to linearly interpolate between two adjacent points of the first 17 gamma entries to create the result value. The first 16 entries are stored in SPR_GAMC with 10 bits per color in a 0.10 format with 0 integer and 10 fractional. The 17th entry is stored in the SPR_GAMC16 register with 11 bits per color in a 1.10 format with 1 integer and 10 fractional bits.

* For input values greater than or equal to 1.0 and less than 3.0, the input value is used to linearly interpolate between the 17th and 18th gamma entries to create the result value. The 18th entry is stored in the SPR_GAMC17 register with 12 bits per color in a 2.10 format with 2 integer and 10 fractional bits.

* For negative input values, gamma is mirrored along the X-axis, giving the same result as positive input values, except for a negative sign. When gamma input may be negative, the first gamma point should be programmed to a value of 0.0 in order to have a symmetric mirroring.

To program the gamma correction entries, calculate the desired gamma curve for inputs from 0 to 3.0. The curve must be flat or increasing, never decreasing. For inputs of 0 to 1.0, multiply the input value by 16 to find the gamma entry number, then store the desired gamma result in that entry. For inputs greater than 1.0 and less than or equal to 3.0, store the result for an input of 3.0 in the 18th gamma entry (SRP_GAMC17).

The gamma correction registers should only be updated when the sprite is off, otherwise screen artifacts may show temporarily.
Gamma correction can be enabled or disabled through the sprite control register.
See Pipe Gamma for an example gamma curve diagram.

DWord	Bit	Description	
0	31:30	**Reserved**	
		Format:	MBZ
	29:0	**GAMC0**	
		Default Value:	00000000h
		Format:	SPR_GAMC REFERENCE POINT FORMAT
1	31:30	**Reserved**	
		Format:	MBZ
	29:0	**GAMC1**	
		Default Value:	04010040h
		Format:	SPR_GAMC REFERENCE POINT FORMAT
2	31:30	**Reserved**	
		Format:	MBZ
	29:0	**GAMC2**	
		Default Value:	08020080h
		Format:	SPR_GAMC REFERENCE POINT FORMAT
3	31:30	**Reserved**	
		Format:	MBZ
	29:0	**GAMC3**	
		Default Value:	0C0300C0h
		Format:	SPR_GAMC REFERENCE POINT FORMAT
4	31:30	**Reserved**	

SPR_GAMC				
		Format:	MBZ	
	29:0	**GAMC4**		
		Default Value:	10040100h	
		Format:	SPR_GAMC REFERENCE POINT FORMAT	
5	31:30	**Reserved**		
		Format:	MBZ	
	29:0	**GAMC5**		
		Default Value:	14050140h	
		Format:	SPR_GAMC REFERENCE POINT FORMAT	
6	31:30	**Reserved**		
		Format:	MBZ	
	29:0	**GAMC6**		
		Default Value:	18060180h	
		Format:	SPR_GAMC REFERENCE POINT FORMAT	
7	31:30	**Reserved**		
		Format:	MBZ	
	29:0	**GAMC7**		
		Default Value:	1C0701C0h	
		Format:	SPR_GAMC REFERENCE POINT FORMAT	
8	31:30	**Reserved**		
		Format:	MBZ	
	29:0	**GAMC8**		
		Default Value:	20080200h	
		Format:	SPR_GAMC REFERENCE POINT FORMAT	
9	31:30	**Reserved**		
		Format:	MBZ	
	29:0	**GAMC9**		
		Default Value:	24090240h	
		Format:	SPR_GAMC REFERENCE POINT FORMAT	
10	31:30	**Reserved**		
		Format:	MBZ	
	29:0	**GAMC10**		
		Default Value:	280A0280h	
		Format:	SPR_GAMC REFERENCE POINT FORMAT	
11	31:30	**Reserved**		
		Format:	MBZ	
	29:0	**GAMC11**		
		Default Value:	2C0B02C0h	
		Format:	SPR_GAMC REFERENCE POINT FORMAT	
12	31:30	**Reserved**		
		Format:	MBZ	
	29:0	**GAMC12**		
		Default Value:	300C0300h	
		Format:	SPR_GAMC REFERENCE POINT FORMAT	
13	31:30	**Reserved**		
		Format:	MBZ	
	29:0	**GAMC13**		

SPR_GAMC			
		Default Value:	340D0340h
		Format:	SPR_GAMC REFERENCE POINT FORMAT
14	31:30	**Reserved**	
		Format:	MBZ
	29:0	**GAMC14**	
		Default Value:	380E0380h
		Format:	SPR_GAMC REFERENCE POINT FORMAT
15	31:30	**Reserved**	
		Format:	MBZ
	29:0	**GAMC15**	
		Default Value:	3C0F03C0h
		Format:	SPR_GAMC REFERENCE POINT FORMAT

5.4.13 SPR_GAMC16-Sprite Gamma Correction Point 16

SPR_GAMC16	
Register Space:	MMIO: 0/2/0
Project:	
Default Value:	0x00000400, 0x00000400, 0x00000400
Access:	R/W
Size (in bits):	3x32
Address:	70440h-7044Bh
Name:	Sprite A Gamma Correction Point 16
ShortName:	SPR_GAMC16_[0-2]_A
Address:	71440h-7144Bh
Name:	Sprite B Gamma Correction Point 16
ShortName:	SPR_GAMC16_[0-2]_B
Address:	72440h-7244Bh
Name:	Sprite C Gamma Correction Point 16
ShortName:	SPR_GAMC16_[0-2]_C

These registers are used to determine the 17th reference point (point 16 when counting from 0) for sprite gamma correction.
The values are represented in an unsigned 1.10 format with 1 integer and 10 fractional bits.
See SPR_GAMC for sprite gamma programming information.

Programming Notes
Restriction : The value should always be programmed to be less than or equal to 1.0.

DWord	Bit	Description	
0	31:11	**Reserved**	
		Format:	MBZ
	10:0	**GAMC16R**	
		Default Value:	00000400h
		Format:	U1.10

SPR_GAMC16		
		This value specifies the 17th reference point that is used for the red color channel sprite gamma correction.
1	31:11	**Reserved** Format: MBZ
	10:0	**GAMC16G** Default Value: 00000400h Format: U1.10 This value specifies the 17th reference point that is used for the green color channel sprite gamma correction.
2	31:11	**Reserved** Format: MBZ
	10:0	**GAMC16B** Default Value: 00000400h Format: U1.10 This value specifies the 17th reference point that is used for the blue color channel sprite gamma correction.

5.4.14 SPR_GAMC17-Sprite Gamma Correction Point 17

SPR_GAMC17	
Register Space:	MMIO: 0/2/0
Project:	
Default Value:	0x00000C00, 0x00000C00, 0x00000C00
Access:	R/W
Size (in bits):	3x32
Address:	7044Ch-70457h
Name:	Sprite A Gamma Correction Point 17
ShortName:	SPR_GAMC17_[0-2]_A
Address:	7144Ch-71457h
Name:	Sprite B Gamma Correction Point 17
ShortName:	SPR_GAMC17_[0-2]_B
Address:	7244Ch-72457h
Name:	Sprite C Gamma Correction Point 17
ShortName:	SPR_GAMC17_[0-2]_C

These registers are used to determine the 18th reference point (point 17 when counting from 0) for sprite gamma correction.
The values are represented in an unsigned 2.10 format with 2 integer and 10 fractional bits.
See SPR_GAMC for sprite gamma programming information.

Programming Notes

Restriction : The value should always be programmed to be less than or equal to 3.0.

DWord	Bit	Description	
0	31:12	**Reserved**	
		Format:	MBZ
	11:0	**GAMC17R**	
		Default Value:	00000C00h
		Format:	U2.10
		This value specifies the 18th reference point that is used for the red color channel sprite gamma correction.	
1	31:12	**Reserved**	
		Format:	MBZ
	11:0	**GAMC17G**	
		Default Value:	00000C00h
		Format:	U2.10
		This value specifies the 18th reference point that is used for the green color channel sprite gamma correction.	
2	31:12	**Reserved**	
		Format:	MBZ
	11:0	**GAMC17B**	
		Default Value:	00000C00h
		Format:	U2.10
		This value specifies the 18th reference point that is used for the blue color channel sprite gamma correction.	

Revision History

Revision Number	Description	Revision Date
1.0	First 2012 OpenSource edition	May 2012

§§