Intel Open Source Graphics Programmer's Reference Manual (PRM) for the 2013 Intel® Core™ Processor Family, including Intel HD Graphics, Intel Iris™ Graphics and Intel Iris Pro Graphics

Volume 12: PCIE Configuration Registers (Haswell)

12/18/2013

Copyright

PCIE Configuration Registers

Table of Contents

GFX PCI Registers

MSA Registers

MPGFXTRK_CR_FENCE0_0_2_0_GTTMMADR

B/D/F/Type: 0/2/0/GTTMMADR

Address Offset: 0x100000

Access: 64 bits

Size: RW

Fence Register

Bit	Type	Default Value	RST Type	Description
63:44	RW	0x0	default/uncore	**FENCEUP:** Bits 31:12 of the ending Graphics Address of the fence region. Fence regions must be aligned to a 4KB page. This address represents the last 4KB page of the fence region Upper Bound is included in the fence region. Graphics Address is the offset within GMADR space.
42:32	RW	0x0	default/uncore	**PITCH:** This field specifies the width pitch of the fence region in multiple of "tile widths". For Tile X this field must be programmed to a multiple of 512B "003" is the minimum value and for Tile Y this field must be programmed to a multiple of 128B "000" is the minimum value. 000h 128B 001h 256B 002h 384B 003h 512B 004h 640B 005h 768B 006h 896B 007h 1024B ... 3FFh 128KB 7FFh 256KB

Bit	Type	Default Value	RST Type	Description
31:12	RW	0x0	default/uncore	**FENCELOW:** Bits 31:12 of the starting Graphics Address of the fence region. Fence regions must be aligned to 4KB. This address represents the first 4KB page of the fence region Lower Bound is included in the fence region. Graphics Address is the offset within GMADR.
1:1	RW	0x0	default/uncore	**TILE:** This field specifies the spatial ordering of QW within tiles. 0b - Consectuvie SWords 32B sequenced in the X direction 1b - Consectuvie OWords 16B sequenced in the Y direction
0:0	RW	0x0	default/uncore	**FENCEVAL:** This field specifies whether or not this fence register defines a fence region. 0b - FENCE INVALID 1b - FENCE VALID

MPGFXTRK_CR_FENCE1_0_2_0_GTTMMADR

B/D/F/Type: 0/2/0/GTTMMADR
Address Offset: 0x100008
Access: 64 bits
Size: RW

Fence Register

Bit	Type	Default Value	RST Type	Description
63:44	RW	0x0	default/uncore	**FENCEUP:** Bits 31:12 of the ending Graphics Address of the fence region. Fence regions must be aligned to a 4KB page. This address represents the last 4KB page of the fence region Upper Bound is included in the fence region. Graphics Address is the offset within GMADR space.
42:32	RW	0x0	default/uncore	**PITCH:** This field specifies the width pitch of the fence region in multiple of "tile widths". For Tile X this field must be programmed to a multiple of 512B "003" is the minimum value and for Tile Y this field must be programmed to a multiple of 128B "000" is the minimum value. 000h 128B 001h 256B 002h 384B 003h 512B 004h 640B 005h 768B 006h 896B 007h 1024B ... 3FFh 128KB 7FFh 256KB
31:12	RW	0x0	default/uncore	**FENCELOW:** Bits 31:12 of the starting Graphics Address of the fence region. Fence regions must be aligned to 4KB. This address represents the first 4KB page of the fence region Lower Bound is included in the fence region. Graphics Address is the offset within GMADR.

Bit	Type	Default Value	RST Type	Description
1:1	RW	0x0	default/uncore	**TILE:** This field specifies the spatial ordering of QW within tiles. 0b - Consectuvie SWords 32B sequenced in the X direction 1b - Consectuvie OWords 16B sequenced in the Y direction
0:0	RW	0x0	default/uncore	**FENCEVAL:** This field specifies whether or not this fence register defines a fence region. 0b - FENCE INVALID 1b - FENCE VALID

MPGFXTRK_CR_FENCE2_0_2_0_GTTMMADR

B/D/F/Type: 0/2/0/GTTMMADR
Address Offset: 0x100010
Access: 64 bits
Size: RW

Fence Register

Bit	Type	Default Value	RST Type	Description
63:44	RW	0x0	default/uncore	**FENCEUP:** Bits 31:12 of the ending Graphics Address of the fence region. Fence regions must be aligned to a 4KB page. This address represents the last 4KB page of the fence region Upper Bound is included in the fence region. Graphics Address is the offset within GMADR space.
42:32	RW	0x0	default/uncore	**PITCH:** This field specifies the width pitch of the fence region in multiple of "tile widths". For Tile X this field must be programmed to a multiple of 512B "003" is the minimum value and for Tile Y this field must be programmed to a multiple of 128B "000" is the minimum value. 000h 128B 001h 256B 002h 384B 003h 512B 004h 640B 005h 768B 006h 896B 007h 1024B ... 3FFh 128KB 7FFh 256KB
31:12	RW	0x0	default/uncore	**FENCELOW:** Bits 31:12 of the starting Graphics Address of the fence region. Fence regions must be aligned to 4KB. This address represents the first 4KB page of the fence region Lower Bound is included in the fence region. Graphics Address is the offset within GMADR.

Bit	Type	Default Value	RST Type	Description
1:1	RW	0x0	default/uncore	**TILE:** This field specifies the spatial ordering of QW within tiles. 0b - Consectuvie SWords 32B sequenced in the X direction 1b - Consectuvie OWords 16B sequenced in the Y direction
0:0	RW	0x0	default/uncore	**FENCEVAL:** This field specifies whether or not this fence register defines a fence region. 0b - FENCE INVALID 1b - FENCE VALID

MPGFXTRK_CR_FENCE3_0_2_0_GTTMMADR

B/D/F/Type: 0/2/0/GTTMMADR
Address Offset: 0x100018
Access: 64 bits
Size: RW

Fence Register

Bit	Type	Default Value	RST Type	Description
63:44	RW	0x0	default/uncore	**FENCEUP:** Bits 31:12 of the ending Graphics Address of the fence region. Fence regions must be aligned to a 4KB page. This address represents the last 4KB page of the fence region Upper Bound is included in the fence region. Graphics Address is the offset within GMADR space.
42:32	RW	0x0	default/uncore	**PITCH:** This field specifies the width pitch of the fence region in multiple of "tile widths". For Tile X this field must be programmed to a multiple of 512B "003" is the minimum value and for Tile Y this field must be programmed to a multiple of 128B "000" is the minimum value. 000h 128B 001h 256B 002h 384B 003h 512B 004h 640B 005h 768B 006h 896B 007h 1024B ... 3FFh 128KB 7FFh 256KB
31:12	RW	0x0	default/uncore	**FENCELOW:** Bits 31:12 of the starting Graphics Address of the fence region. Fence regions must be aligned to 4KB. This address represents the first 4KB page of the fence region Lower Bound is included in the fence region. Graphics Address is the offset within GMADR.

Bit	Type	Default Value	RST Type	Description
1:1	RW	0x0	default/uncore	**TILE:** This field specifies the spatial ordering of QW within tiles. 0b - Consectuvie SWords 32B sequenced in the X direction 1b - Consectuvie OWords 16B sequenced in the Y direction
0:0	RW	0x0	default/uncore	**FENCEVAL:** This field specifies whether or not this fence register defines a fence region. 0b - FENCE INVALID 1b - FENCE VALID

MPGFXTRK_CR_FENCE4_0_2_0_GTTMMADR

B/D/F/Type: 0/2/0/GTTMMADR
Address Offset: 0x100020
Access: 64 bits
Size: RW

Fence Register

Bit	Type	Default Value	RST Type	Description
63:44	RW	0x0	default/uncore	**FENCEUP:** Bits 31:12 of the ending Graphics Address of the fence region. Fence regions must be aligned to a 4KB page. This address represents the last 4KB page of the fence region Upper Bound is included in the fence region. Graphics Address is the offset within GMADR space.
42:32	RW	0x0	default/uncore	**PITCH:** This field specifies the width pitch of the fence region in multiple of "tile widths". For Tile X this field must be programmed to a multiple of 512B "003" is the minimum value and for Tile Y this field must be programmed to a multiple of 128B "000" is the minimum value. 000h 128B 001h 256B 002h 384B 003h 512B 004h 640B 005h 768B 006h 896B 007h 1024B ... 3FFh 128KB 7FFh 256KB
31:12	RW	0x0	default/uncore	**FENCELOW:** Bits 31:12 of the starting Graphics Address of the fence region. Fence regions must be aligned to 4KB. This address represents the first 4KB page of the fence region Lower Bound is included in the fence region. Graphics Address is the offset within GMADR.

Bit	Type	Default Value	RST Type	Description
1:1	RW	0x0	default/uncore	**TILE:** This field specifies the spatial ordering of QW within tiles. 0b - Consectuvie SWords 32B sequenced in the X direction 1b - Consectuvie OWords 16B sequenced in the Y direction
0:0	RW	0x0	default/uncore	**FENCEVAL:** This field specifies whether or not this fence register defines a fence region. 0b - FENCE INVALID 1b - FENCE VALID

MPGFXTRK_CR_FENCE5_0_2_0_GTTMMADR

B/D/F/Type: 0/2/0/GTTMMADR
Address Offset: 0x100028
Access: 64 bits
Size: RW

Fence Register

Bit	Type	Default Value	RST Type	Description
63:44	RW	0x0	default/uncore	**FENCEUP:** Bits 31:12 of the ending Graphics Address of the fence region. Fence regions must be aligned to a 4KB page. This address represents the last 4KB page of the fence region Upper Bound is included in the fence region. Graphics Address is the offset within GMADR space.
42:32	RW	0x0	default/uncore	**PITCH:** This field specifies the width pitch of the fence region in multiple of "tile widths". For Tile X this field must be programmed to a multiple of 512B "003" is the minimum value and for Tile Y this field must be programmed to a multiple of 128B "000" is the minimum value. 000h 128B 001h 256B 002h 384B 003h 512B 004h 640B 005h 768B 006h 896B 007h 1024B ... 3FFh 128KB 7FFh 256KB
31:12	RW	0x0	default/uncore	**FENCELOW:** Bits 31:12 of the starting Graphics Address of the fence region. Fence regions must be aligned to 4KB. This address represents the first 4KB page of the fence region Lower Bound is included in the fence region. Graphics Address is the offset within GMADR.

Bit	Type	Default Value	RST Type	Description
1:1	RW	0x0	default/uncore	**TILE:** This field specifies the spatial ordering of QW within tiles. 0b - Consectuvie SWords 32B sequenced in the X direction 1b - Consectuvie OWords 16B sequenced in the Y direction
0:0	RW	0x0	default/uncore	**FENCEVAL:** This field specifies whether or not this fence register defines a fence region. 0b - FENCE INVALID 1b - FENCE VALID

MPGFXTRK_CR_FENCE6_0_2_0_GTTMMADR

B/D/F/Type: 0/2/0/GTTMMADR
Address Offset: 0x100030
Access: 64 bits
Size: RW

Fence Register

Bit	Type	Default Value	RST Type	Description
63:44	RW	0x0	default/uncore	**FENCEUP:** Bits 31:12 of the ending Graphics Address of the fence region. Fence regions must be aligned to a 4KB page. This address represents the last 4KB page of the fence region Upper Bound is included in the fence region. Graphics Address is the offset within GMADR space.
42:32	RW	0x0	default/uncore	**PITCH:** This field specifies the width pitch of the fence region in multiple of "tile widths". For Tile X this field must be programmed to a multiple of 512B "003" is the minimum value and for Tile Y this field must be programmed to a multiple of 128B "000" is the minimum value. 000h 128B 001h 256B 002h 384B 003h 512B 004h 640B 005h 768B 006h 896B 007h 1024B ... 3FFh 128KB 7FFh 256KB
31:12	RW	0x0	default/uncore	**FENCELOW:** Bits 31:12 of the starting Graphics Address of the fence region. Fence regions must be aligned to 4KB. This address represents the first 4KB page of the fence region Lower Bound is included in the fence region. Graphics Address is the offset within GMADR.

Bit	Type	Default Value	RST Type	Description
1:1	RW	0x0	default/uncore	**TILE:** This field specifies the spatial ordering of QW within tiles. 0b - Consectuvie SWords 32B sequenced in the X direction 1b - Consectuvie OWords 16B sequenced in the Y direction
0:0	RW	0x0	default/uncore	**FENCEVAL:** This field specifies whether or not this fence register defines a fence region. 0b - FENCE INVALID 1b - FENCE VALID

MPGFXTRK_CR_FENCE7_0_2_0_GTTMMADR

B/D/F/Type: 0/2/0/GTTMMADR
Address Offset: 0x100038
Access: 64 bits
Size: RW

Fence Register

Bit	Type	Default Value	RST Type	Description
63:44	RW	0x0	default/uncore	**FENCEUP:** Bits 31:12 of the ending Graphics Address of the fence region. Fence regions must be aligned to a 4KB page. This address represents the last 4KB page of the fence region Upper Bound is included in the fence region. Graphics Address is the offset within GMADR space.
42:32	RW	0x0	default/uncore	**PITCH:** This field specifies the width pitch of the fence region in multiple of "tile widths". For Tile X this field must be programmed to a multiple of 512B "003" is the minimum value and for Tile Y this field must be programmed to a multiple of 128B "000" is the minimum value. 000h 128B 001h 256B 002h 384B 003h 512B 004h 640B 005h 768B 006h 896B 007h 1024B ... 3FFh 128KB 7FFh 256KB
31:12	RW	0x0	default/uncore	**FENCELOW:** Bits 31:12 of the starting Graphics Address of the fence region. Fence regions must be aligned to 4KB. This address represents the first 4KB page of the fence region Lower Bound is included in the fence region. Graphics Address is the offset within GMADR.

Bit	Type	Default Value	RST Type	Description
1:1	RW	0x0	default/uncore	**TILE:** This field specifies the spatial ordering of QW within tiles. 0b - Consectuvie SWords 32B sequenced in the X direction 1b - Consectuvie OWords 16B sequenced in the Y direction
0:0	RW	0x0	default/uncore	**FENCEVAL:** This field specifies whether or not this fence register defines a fence region. 0b - FENCE INVALID 1b - FENCE VALID

MPGFXTRK_CR_FENCE8_0_2_0_GTTMMADR

B/D/F/Type: 0/2/0/GTTMMADR
Address Offset: 0x100040
Access: 64 bits
Size: RW

Fence Register

Bit	Type	Default Value	RST Type	Description
63:44	RW	0x0	default/uncore	**FENCEUP:** Bits 31:12 of the ending Graphics Address of the fence region. Fence regions must be aligned to a 4KB page. This address represents the last 4KB page of the fence region Upper Bound is included in the fence region. Graphics Address is the offset within GMADR space.
42:32	RW	0x0	default/uncore	**PITCH:** This field specifies the width pitch of the fence region in multiple of "tile widths". For Tile X this field must be programmed to a multiple of 512B "003" is the minimum value and for Tile Y this field must be programmed to a multiple of 128B "000" is the minimum value. 000h 128B 001h 256B 002h 384B 003h 512B 004h 640B 005h 768B 006h 896B 007h 1024B ... 3FFh 128KB 7FFh 256KB
31:12	RW	0x0	default/uncore	**FENCELOW:** Bits 31:12 of the starting Graphics Address of the fence region. Fence regions must be aligned to 4KB. This address represents the first 4KB page of the fence region Lower Bound is included in the fence region. Graphics Address is the offset within GMADR.

Bit	Type	Default Value	RST Type	Description
1:1	RW	0x0	default/uncore	**TILE:** This field specifies the spatial ordering of QW within tiles. 0b - Consectuvie SWords 32B sequenced in the X direction 1b - Consectuvie OWords 16B sequenced in the Y direction
0:0	RW	0x0	default/uncore	**FENCEVAL:** This field specifies whether or not this fence register defines a fence region. 0b - FENCE INVALID 1b - FENCE VALID

MPGFXTRK_CR_FENCE9_0_2_0_GTTMMADR

B/D/F/Type: 0/2/0/GTTMMADR
Address Offset: 0x100048
Access: 64 bits
Size: RW

Fence Register

Bit	Type	Default Value	RST Type	Description
63:44	RW	0x0	default/uncore	**FENCEUP:** Bits 31:12 of the ending Graphics Address of the fence region. Fence regions must be aligned to a 4KB page. This address represents the last 4KB page of the fence region Upper Bound is included in the fence region. Graphics Address is the offset within GMADR space.
42:32	RW	0x0	default/uncore	**PITCH:** This field specifies the width pitch of the fence region in multiple of "tile widths". For Tile X this field must be programmed to a multiple of 512B "003" is the minimum value and for Tile Y this field must be programmed to a multiple of 128B "000" is the minimum value. 000h 128B 001h 256B 002h 384B 003h 512B 004h 640B 005h 768B 006h 896B 007h 1024B ... 3FFh 128KB 7FFh 256KB
31:12	RW	0x0	default/uncore	**FENCELOW:** Bits 31:12 of the starting Graphics Address of the fence region. Fence regions must be aligned to 4KB. This address represents the first 4KB page of the fence region Lower Bound is included in the fence region. Graphics Address is the offset within GMADR.

Bit	Type	Default Value	RST Type	Description
1:1	RW	0x0	default/uncore	**TILE:** This field specifies the spatial ordering of QW within tiles. 0b - Consectuvie SWords 32B sequenced in the X direction 1b - Consectuvie OWords 16B sequenced in the Y direction
0:0	RW	0x0	default/uncore	**FENCEVAL:** This field specifies whether or not this fence register defines a fence region. 0b - FENCE INVALID 1b - FENCE VALID

MPGFXTRK_CR_FENCE10_0_2_0_GTTMMADR

B/D/F/Type: 0/2/0/GTTMMADR
Address Offset: 0x100050
Access: 64 bits
Size: RW

Fence Register

Bit	Type	Default Value	RST Type	Description
63:44	RW	0x0	default/uncore	**FENCEUP:** Bits 31:12 of the ending Graphics Address of the fence region. Fence regions must be aligned to a 4KB page. This address represents the last 4KB page of the fence region Upper Bound is included in the fence region. Graphics Address is the offset within GMADR space.
42:32	RW	0x0	default/uncore	**PITCH:** This field specifies the width pitch of the fence region in multiple of "tile widths". For Tile X this field must be programmed to a multiple of 512B "003" is the minimum value and for Tile Y this field must be programmed to a multiple of 128B "000" is the minimum value. 000h 128B 001h 256B 002h 384B 003h 512B 004h 640B 005h 768B 006h 896B 007h 1024B ... 3FFh 128KB 7FFh 256KB
31:12	RW	0x0	default/uncore	**FENCELOW:** Bits 31:12 of the starting Graphics Address of the fence region. Fence regions must be aligned to 4KB. This address represents the first 4KB page of the fence region Lower Bound is included in the fence region. Graphics Address is the offset within GMADR.

Bit	Type	Default Value	RST Type	Description
1:1	RW	0x0	default/uncore	**TILE:** This field specifies the spatial ordering of QW within tiles. 0b - Consectuvie SWords 32B sequenced in the X direction 1b - Consectuvie OWords 16B sequenced in the Y direction
0:0	RW	0x0	default/uncore	**FENCEVAL:** This field specifies whether or not this fence register defines a fence region. 0b - FENCE INVALID 1b - FENCE VALID

MPGFXTRK_CR_FENCE11_0_2_0_GTTMMADR

B/D/F/Type: 0/2/0/GTTMMADR
Address Offset: 0x100058
Access: 64 bits
Size: RW

Fence Register

Bit	Type	Default Value	RST Type	Description
63:44	RW	0x0	default/uncore	**FENCEUP:** Bits 31:12 of the ending Graphics Address of the fence region. Fence regions must be aligned to a 4KB page. This address represents the last 4KB page of the fence region Upper Bound is included in the fence region. Graphics Address is the offset within GMADR space.
42:32	RW	0x0	default/uncore	**PITCH:** This field specifies the width pitch of the fence region in multiple of "tile widths". For Tile X this field must be programmed to a multiple of 512B "003" is the minimum value and for Tile Y this field must be programmed to a multiple of 128B "000" is the minimum value. 000h 128B 001h 256B 002h 384B 003h 512B 004h 640B 005h 768B 006h 896B 007h 1024B ... 3FFh 128KB 7FFh 256KB
31:12	RW	0x0	default/uncore	**FENCELOW:** Bits 31:12 of the starting Graphics Address of the fence region. Fence regions must be aligned to 4KB. This address represents the first 4KB page of the fence region Lower Bound is included in the fence region. Graphics Address is the offset within GMADR.

Bit	Type	Default Value	RST Type	Description
1:1	RW	0x0	default/uncore	**TILE:** This field specifies the spatial ordering of QW within tiles. 0b - Consectuvie SWords 32B sequenced in the X direction 1b - Consectuvie OWords 16B sequenced in the Y direction
0:0	RW	0x0	default/uncore	**FENCEVAL:** This field specifies whether or not this fence register defines a fence region. 0b - FENCE INVALID 1b - FENCE VALID

MPGFXTRK_CR_FENCE12_0_2_0_GTTMMADR

B/D/F/Type: 0/2/0/GTTMMADR
Address Offset: 0x100060
Access: 64 bits
Size: RW

Fence Register

Bit	Type	Default Value	RST Type	Description
63:44	RW	0x0	default/uncore	**FENCEUP:** Bits 31:12 of the ending Graphics Address of the fence region. Fence regions must be aligned to a 4KB page. This address represents the last 4KB page of the fence region Upper Bound is included in the fence region. Graphics Address is the offset within GMADR space.
42:32	RW	0x0	default/uncore	**PITCH:** This field specifies the width pitch of the fence region in multiple of "tile widths". For Tile X this field must be programmed to a multiple of 512B "003" is the minimum value and for Tile Y this field must be programmed to a multiple of 128B "000" is the minimum value. 000h 128B 001h 256B 002h 384B 003h 512B 004h 640B 005h 768B 006h 896B 007h 1024B ... 3FFh 128KB 7FFh 256KB
31:12	RW	0x0	default/uncore	**FENCELOW:** Bits 31:12 of the starting Graphics Address of the fence region. Fence regions must be aligned to 4KB. This address represents the first 4KB page of the fence region Lower Bound is included in the fence region. Graphics Address is the offset within GMADR.

Bit	Type	Default Value	RST Type	Description
1:1	RW	0x0	default/uncore	**TILE:** This field specifies the spatial ordering of QW within tiles. 0b - Consectuvie SWords 32B sequenced in the X direction 1b - Consectuvie OWords 16B sequenced in the Y direction
0:0	RW	0x0	default/uncore	**FENCEVAL:** This field specifies whether or not this fence register defines a fence region. 0b - FENCE INVALID 1b - FENCE VALID

MPGFXTRK_CR_FENCE13_0_2_0_GTTMMADR

B/D/F/Type: 0/2/0/GTTMMADR
Address Offset: 0x100068
Access: 64 bits
Size: RW

Fence Register

Bit	Type	Default Value	RST Type	Description
63:44	RW	0x0	default/uncore	**FENCEUP:** Bits 31:12 of the ending Graphics Address of the fence region. Fence regions must be aligned to a 4KB page. This address represents the last 4KB page of the fence region Upper Bound is included in the fence region. Graphics Address is the offset within GMADR space.
42:32	RW	0x0	default/uncore	**PITCH:** This field specifies the width pitch of the fence region in multiple of "tile widths". For Tile X this field must be programmed to a multiple of 512B "003" is the minimum value and for Tile Y this field must be programmed to a multiple of 128B "000" is the minimum value. 000h 128B 001h 256B 002h 384B 003h 512B 004h 640B 005h 768B 006h 896B 007h 1024B ... 3FFh 128KB 7FFh 256KB
31:12	RW	0x0	default/uncore	**FENCELOW:** Bits 31:12 of the starting Graphics Address of the fence region. Fence regions must be aligned to 4KB. This address represents the first 4KB page of the fence region Lower Bound is included in the fence region. Graphics Address is the offset within GMADR.

Bit	Type	Default Value	RST Type	Description
1:1	RW	0x0	default/uncore	**TILE:** This field specifies the spatial ordering of QW within tiles. 0b - Consectuvie SWords 32B sequenced in the X direction 1b - Consectuvie OWords 16B sequenced in the Y direction
0:0	RW	0x0	default/uncore	**FENCEVAL:** This field specifies whether or not this fence register defines a fence region. 0b - FENCE INVALID 1b - FENCE VALID

MPGFXTRK_CR_FENCE14_0_2_0_GTTMMADR

B/D/F/Type: 0/2/0/GTTMMADR
Address Offset: 0x100070
Access: 64 bits
Size: RW

Fence Register

Bit	Type	Default Value	RST Type	Description
63:44	RW	0x0	default/uncore	**FENCEUP:** Bits 31:12 of the ending Graphics Address of the fence region. Fence regions must be aligned to a 4KB page. This address represents the last 4KB page of the fence region Upper Bound is included in the fence region. Graphics Address is the offset within GMADR space.
42:32	RW	0x0	default/uncore	**PITCH:** This field specifies the width pitch of the fence region in multiple of "tile widths". For Tile X this field must be programmed to a multiple of 512B "003" is the minimum value and for Tile Y this field must be programmed to a multiple of 128B "000" is the minimum value. 000h 128B 001h 256B 002h 384B 003h 512B 004h 640B 005h 768B 006h 896B 007h 1024B ... 3FFh 128KB 7FFh 256KB
31:12	RW	0x0	default/uncore	**FENCELOW:** Bits 31:12 of the starting Graphics Address of the fence region. Fence regions must be aligned to 4KB. This address represents the first 4KB page of the fence region Lower Bound is included in the fence region. Graphics Address is the offset within GMADR.

Bit	Type	Default Value	RST Type	Description
1:1	RW	0x0	default/uncore	**TILE:** This field specifies the spatial ordering of QW within tiles. 0b - Consectuvie SWords 32B sequenced in the X direction 1b - Consectuvie OWords 16B sequenced in the Y direction
0:0	RW	0x0	default/uncore	**FENCEVAL:** This field specifies whether or not this fence register defines a fence region. 0b - FENCE INVALID 1b - FENCE VALID

MPGFXTRK_CR_FENCE15_0_2_0_GTTMMADR

B/D/F/Type: 0/2/0/GTTMMADR
Address Offset: 0x100078
Access: 64 bits
Size: RW

Fence Register

Bit	Type	Default Value	RST Type	Description
63:44	RW	0x0	default/uncore	**FENCEUP:** Bits 31:12 of the ending Graphics Address of the fence region. Fence regions must be aligned to a 4KB page. This address represents the last 4KB page of the fence region Upper Bound is included in the fence region. Graphics Address is the offset within GMADR space.
42:32	RW	0x0	default/uncore	**PITCH:** This field specifies the width pitch of the fence region in multiple of "tile widths". For Tile X this field must be programmed to a multiple of 512B "003" is the minimum value and for Tile Y this field must be programmed to a multiple of 128B "000" is the minimum value. 000h 128B 001h 256B 002h 384B 003h 512B 004h 640B 005h 768B 006h 896B 007h 1024B ... 3FFh 128KB 7FFh 256KB
31:12	RW	0x0	default/uncore	**FENCELOW:** Bits 31:12 of the starting Graphics Address of the fence region. Fence regions must be aligned to 4KB. This address represents the first 4KB page of the fence region Lower Bound is included in the fence region. Graphics Address is the offset within GMADR.

Bit	Type	Default Value	RST Type	Description
1:1	RW	0x0	default/uncore	**TILE:** This field specifies the spatial ordering of QW within tiles. 0b - Consectuvie SWords 32B sequenced in the X direction 1b - Consectuvie OWords 16B sequenced in the Y direction
0:0	RW	0x0	default/uncore	**FENCEVAL:** This field specifies whether or not this fence register defines a fence region. 0b - FENCE INVALID 1b - FENCE VALID

MPGFXTRK_CR_FENCE16_0_2_0_GTTMMADR

B/D/F/Type: 0/2/0/GTTMMADR
Address Offset: 0x100080
Access: 64 bits
Size: RW

Fence Register

Bit	Type	Default Value	RST Type	Description
63:44	RW	0x0	default/uncore	**FENCEUP:** Bits 31:12 of the ending Graphics Address of the fence region. Fence regions must be aligned to a 4KB page. This address represents the last 4KB page of the fence region Upper Bound is included in the fence region. Graphics Address is the offset within GMADR space.
42:32	RW	0x0	default/uncore	**PITCH:** This field specifies the width pitch of the fence region in multiple of "tile widths". For Tile X this field must be programmed to a multiple of 512B "003" is the minimum value and for Tile Y this field must be programmed to a multiple of 128B "000" is the minimum value. 000h 128B 001h 256B 002h 384B 003h 512B 004h 640B 005h 768B 006h 896B 007h 1024B ... 3FFh 128KB 7FFh 256KB
31:12	RW	0x0	default/uncore	**FENCELOW:** Bits 31:12 of the starting Graphics Address of the fence region. Fence regions must be aligned to 4KB. This address represents the first 4KB page of the fence region Lower Bound is included in the fence region. Graphics Address is the offset within GMADR.

Bit	Type	Default Value	RST Type	Description
1:1	RW	0x0	default/uncore	**TILE:** This field specifies the spatial ordering of QW within tiles. 0b - Consectuvie SWords 32B sequenced in the X direction 1b - Consectuvie OWords 16B sequenced in the Y direction
0:0	RW	0x0	default/uncore	**FENCEVAL:** This field specifies whether or not this fence register defines a fence region. 0b - FENCE INVALID 1b - FENCE VALID

MPGFXTRK_CR_FENCE17_0_2_0_GTTMMADR

B/D/F/Type: 0/2/0/GTTMMADR
Address Offset: 0x100088
Access: 64 bits
Size: RW

Fence Register

Bit	Type	Default Value	RST Type	Description
63:44	RW	0x0	default/uncore	**FENCEUP:** Bits 31:12 of the ending Graphics Address of the fence region. Fence regions must be aligned to a 4KB page. This address represents the last 4KB page of the fence region Upper Bound is included in the fence region. Graphics Address is the offset within GMADR space.
42:32	RW	0x0	default/uncore	**PITCH:** This field specifies the width pitch of the fence region in multiple of "tile widths". For Tile X this field must be programmed to a multiple of 512B "003" is the minimum value and for Tile Y this field must be programmed to a multiple of 128B "000" is the minimum value. 000h 128B 001h 256B 002h 384B 003h 512B 004h 640B 005h 768B 006h 896B 007h 1024B ... 3FFh 128KB 7FFh 256KB
31:12	RW	0x0	default/uncore	**FENCELOW:** Bits 31:12 of the starting Graphics Address of the fence region. Fence regions must be aligned to 4KB. This address represents the first 4KB page of the fence region Lower Bound is included in the fence region. Graphics Address is the offset within GMADR.

Bit	Type	Default Value	RST Type	Description
1:1	RW	0x0	default/uncore	**TILE:** This field specifies the spatial ordering of QW within tiles. 0b - Consectuvie SWords 32B sequenced in the X direction 1b - Consectuvie OWords 16B sequenced in the Y direction
0:0	RW	0x0	default/uncore	**FENCEVAL:** This field specifies whether or not this fence register defines a fence region. 0b - FENCE INVALID 1b - FENCE VALID

MPGFXTRK_CR_FENCE18_0_2_0_GTTMMADR

B/D/F/Type: 0/2/0/GTTMMADR
Address Offset: 0x100090
Access: 64 bits
Size: RW

Fence Register

Bit	Type	Default Value	RST Type	Description
63:44	RW	0x0	default/uncore	**FENCEUP:** Bits 31:12 of the ending Graphics Address of the fence region. Fence regions must be aligned to a 4KB page. This address represents the last 4KB page of the fence region Upper Bound is included in the fence region. Graphics Address is the offset within GMADR space.
42:32	RW	0x0	default/uncore	**PITCH:** This field specifies the width pitch of the fence region in multiple of "tile widths". For Tile X this field must be programmed to a multiple of 512B "003" is the minimum value and for Tile Y this field must be programmed to a multiple of 128B "000" is the minimum value. 000h 128B 001h 256B 002h 384B 003h 512B 004h 640B 005h 768B 006h 896B 007h 1024B ... 3FFh 128KB 7FFh 256KB
31:12	RW	0x0	default/uncore	**FENCELOW:** Bits 31:12 of the starting Graphics Address of the fence region. Fence regions must be aligned to 4KB. This address represents the first 4KB page of the fence region Lower Bound is included in the fence region. Graphics Address is the offset within GMADR.

Bit	Type	Default Value	RST Type	Description
1:1	RW	0x0	default/uncore	**TILE:** This field specifies the spatial ordering of QW within tiles. 0b - Consectuvie SWords 32B sequenced in the X direction 1b - Consectuvie OWords 16B sequenced in the Y direction
0:0	RW	0x0	default/uncore	**FENCEVAL:** This field specifies whether or not this fence register defines a fence region. 0b - FENCE INVALID 1b - FENCE VALID

MPGFXTRK_CR_FENCE19_0_2_0_GTTMMADR

B/D/F/Type: 0/2/0/GTTMMADR
Address Offset: 0x100098
Access: 64 bits
Size: RW

Fence Register

Bit	Type	Default Value	RST Type	Description
63:44	RW	0x0	default/uncore	**FENCEUP:** Bits 31:12 of the ending Graphics Address of the fence region. Fence regions must be aligned to a 4KB page. This address represents the last 4KB page of the fence region Upper Bound is included in the fence region. Graphics Address is the offset within GMADR space.
42:32	RW	0x0	default/uncore	**PITCH:** This field specifies the width pitch of the fence region in multiple of "tile widths". For Tile X this field must be programmed to a multiple of 512B "003" is the minimum value and for Tile Y this field must be programmed to a multiple of 128B "000" is the minimum value. 000h 128B 001h 256B 002h 384B 003h 512B 004h 640B 005h 768B 006h 896B 007h 1024B ... 3FFh 128KB 7FFh 256KB
31:12	RW	0x0	default/uncore	**FENCELOW:** Bits 31:12 of the starting Graphics Address of the fence region. Fence regions must be aligned to 4KB. This address represents the first 4KB page of the fence region Lower Bound is included in the fence region. Graphics Address is the offset within GMADR.

Bit	Type	Default Value	RST Type	Description
1:1	RW	0x0	default/uncore	**TILE:** This field specifies the spatial ordering of QW within tiles. 0b - Consectuvie SWords 32B sequenced in the X direction 1b - Consectuvie OWords 16B sequenced in the Y direction
0:0	RW	0x0	default/uncore	**FENCEVAL:** This field specifies whether or not this fence register defines a fence region. 0b - FENCE INVALID 1b - FENCE VALID

MPGFXTRK_CR_FENCE20_0_2_0_GTTMMADR

B/D/F/Type: 0/2/0/GTTMMADR
Address Offset: 0x1000a0
Access: 64 bits
Size: RW

Fence Register

Bit	Type	Default Value	RST Type	Description
63:44	RW	0x0	default/uncore	**FENCEUP:** Bits 31:12 of the ending Graphics Address of the fence region. Fence regions must be aligned to a 4KB page. This address represents the last 4KB page of the fence region Upper Bound is included in the fence region. Graphics Address is the offset within GMADR space.
42:32	RW	0x0	default/uncore	**PITCH:** This field specifies the width pitch of the fence region in multiple of "tile widths". For Tile X this field must be programmed to a multiple of 512B "003" is the minimum value and for Tile Y this field must be programmed to a multiple of 128B "000" is the minimum value. 000h 128B 001h 256B 002h 384B 003h 512B 004h 640B 005h 768B 006h 896B 007h 1024B ... 3FFh 128KB 7FFh 256KB
31:12	RW	0x0	default/uncore	**FENCELOW:** Bits 31:12 of the starting Graphics Address of the fence region. Fence regions must be aligned to 4KB. This address represents the first 4KB page of the fence region Lower Bound is included in the fence region. Graphics Address is the offset within GMADR.

Bit	Type	Default Value	RST Type	Description
1:1	RW	0x0	default/uncore	**TILE:** This field specifies the spatial ordering of QW within tiles. 0b - Consectuvie SWords 32B sequenced in the X direction 1b - Consectuvie OWords 16B sequenced in the Y direction
0:0	RW	0x0	default/uncore	**FENCEVAL:** This field specifies whether or not this fence register defines a fence region. 0b - FENCE INVALID 1b - FENCE VALID

MPGFXTRK_CR_FENCE21_0_2_0_GTTMMADR

B/D/F/Type: 0/2/0/GTTMMADR
Address Offset: 0x1000a8
Access: 64 bits
Size: RW

Fence Register

Bit	Type	Default Value	RST Type	Description
63:44	RW	0x0	default/uncore	**FENCEUP:** Bits 31:12 of the ending Graphics Address of the fence region. Fence regions must be aligned to a 4KB page. This address represents the last 4KB page of the fence region Upper Bound is included in the fence region. Graphics Address is the offset within GMADR space.
42:32	RW	0x0	default/uncore	**PITCH:** This field specifies the width pitch of the fence region in multiple of "tile widths". For Tile X this field must be programmed to a multiple of 512B "003" is the minimum value and for Tile Y this field must be programmed to a multiple of 128B "000" is the minimum value. 000h 128B 001h 256B 002h 384B 003h 512B 004h 640B 005h 768B 006h 896B 007h 1024B ... 3FFh 128KB 7FFh 256KB
31:12	RW	0x0	default/uncore	**FENCELOW:** Bits 31:12 of the starting Graphics Address of the fence region. Fence regions must be aligned to 4KB. This address represents the first 4KB page of the fence region Lower Bound is included in the fence region. Graphics Address is the offset within GMADR.

Bit	Type	Default Value	RST Type	Description
1:1	RW	0x0	default/uncore	**TILE:** This field specifies the spatial ordering of QW within tiles. 0b - Consectuvie SWords 32B sequenced in the X direction 1b - Consectuvie OWords 16B sequenced in the Y direction
0:0	RW	0x0	default/uncore	**FENCEVAL:** This field specifies whether or not this fence register defines a fence region. 0b - FENCE INVALID 1b - FENCE VALID

MPGFXTRK_CR_FENCE22_0_2_0_GTTMMADR

B/D/F/Type: 0/2/0/GTTMMADR
Address Offset: 0x1000b0
Access: 64 bits
Size: RW

Fence Register

Bit	Type	Default Value	RST Type	Description
63:44	RW	0x0	default/uncore	**FENCEUP:** Bits 31:12 of the ending Graphics Address of the fence region. Fence regions must be aligned to a 4KB page. This address represents the last 4KB page of the fence region Upper Bound is included in the fence region. Graphics Address is the offset within GMADR space.
42:32	RW	0x0	default/uncore	**PITCH:** This field specifies the width pitch of the fence region in multiple of "tile widths". For Tile X this field must be programmed to a multiple of 512B "003" is the minimum value and for Tile Y this field must be programmed to a multiple of 128B "000" is the minimum value. 000h 128B 001h 256B 002h 384B 003h 512B 004h 640B 005h 768B 006h 896B 007h 1024B ... 3FFh 128KB 7FFh 256KB
31:12	RW	0x0	default/uncore	**FENCELOW:** Bits 31:12 of the starting Graphics Address of the fence region. Fence regions must be aligned to 4KB. This address represents the first 4KB page of the fence region Lower Bound is included in the fence region. Graphics Address is the offset within GMADR.

Bit	Type	Default Value	RST Type	Description
1:1	RW	0x0	default/uncore	**TILE:** This field specifies the spatial ordering of QW within tiles. 0b - Consectuvie SWords 32B sequenced in the X direction 1b - Consectuvie OWords 16B sequenced in the Y direction
0:0	RW	0x0	default/uncore	**FENCEVAL:** This field specifies whether or not this fence register defines a fence region. 0b - FENCE INVALID 1b - FENCE VALID

MPGFXTRK_CR_FENCE23_0_2_0_GTTMMADR

B/D/F/Type: 0/2/0/GTTMMADR
Address Offset: 0x1000b8
Access: 64 bits
Size: RW

Fence Register

Bit	Type	Default Value	RST Type	Description
63:44	RW	0x0	default/uncore	**FENCEUP:** Bits 31:12 of the ending Graphics Address of the fence region. Fence regions must be aligned to a 4KB page. This address represents the last 4KB page of the fence region Upper Bound is included in the fence region. Graphics Address is the offset within GMADR space.
42:32	RW	0x0	default/uncore	**PITCH:** This field specifies the width pitch of the fence region in multiple of "tile widths". For Tile X this field must be programmed to a multiple of 512B "003" is the minimum value and for Tile Y this field must be programmed to a multiple of 128B "000" is the minimum value. 000h 128B 001h 256B 002h 384B 003h 512B 004h 640B 005h 768B 006h 896B 007h 1024B ... 3FFh 128KB 7FFh 256KB
31:12	RW	0x0	default/uncore	**FENCELOW:** Bits 31:12 of the starting Graphics Address of the fence region. Fence regions must be aligned to 4KB. This address represents the first 4KB page of the fence region Lower Bound is included in the fence region. Graphics Address is the offset within GMADR.

Bit	Type	Default Value	RST Type	Description
1:1	RW	0x0	default/uncore	**TILE:** This field specifies the spatial ordering of QW within tiles. 0b - Consectuvie SWords 32B sequenced in the X direction 1b - Consectuvie OWords 16B sequenced in the Y direction
0:0	RW	0x0	default/uncore	**FENCEVAL:** This field specifies whether or not this fence register defines a fence region. 0b - FENCE INVALID 1b - FENCE VALID

MPGFXTRK_CR_FENCE24_0_2_0_GTTMMADR

B/D/F/Type: 0/2/0/GTTMMADR
Address Offset: 0x1000c0
Access: 64 bits
Size: RW

Fence Register

Bit	Type	Default Value	RST Type	Description
63:44	RW	0x0	default/uncore	**FENCEUP:** Bits 31:12 of the ending Graphics Address of the fence region. Fence regions must be aligned to a 4KB page. This address represents the last 4KB page of the fence region Upper Bound is included in the fence region. Graphics Address is the offset within GMADR space.
42:32	RW	0x0	default/uncore	**PITCH:** This field specifies the width pitch of the fence region in multiple of "tile widths". For Tile X this field must be programmed to a multiple of 512B "003" is the minimum value and for Tile Y this field must be programmed to a multiple of 128B "000" is the minimum value. 000h 128B 001h 256B 002h 384B 003h 512B 004h 640B 005h 768B 006h 896B 007h 1024B ... 3FFh 128KB 7FFh 256KB
31:12	RW	0x0	default/uncore	**FENCELOW:** Bits 31:12 of the starting Graphics Address of the fence region. Fence regions must be aligned to 4KB. This address represents the first 4KB page of the fence region Lower Bound is included in the fence region. Graphics Address is the offset within GMADR.

Bit	Type	Default Value	RST Type	Description
1:1	RW	0x0	default/uncore	**TILE:** This field specifies the spatial ordering of QW within tiles. 0b - Consectuvie SWords 32B sequenced in the X direction 1b - Consectuvie OWords 16B sequenced in the Y direction
0:0	RW	0x0	default/uncore	**FENCEVAL:** This field specifies whether or not this fence register defines a fence region. 0b - FENCE INVALID 1b - FENCE VALID

MPGFXTRK_CR_FENCE25_0_2_0_GTTMMADR

B/D/F/Type: 0/2/0/GTTMMADR
Address Offset: 0x1000c8
Access: 64 bits
Size: RW

Fence Register

Bit	Type	Default Value	RST Type	Description
63:44	RW	0x0	default/uncore	**FENCEUP:** Bits 31:12 of the ending Graphics Address of the fence region. Fence regions must be aligned to a 4KB page. This address represents the last 4KB page of the fence region Upper Bound is included in the fence region. Graphics Address is the offset within GMADR space.
42:32	RW	0x0	default/uncore	**PITCH:** This field specifies the width pitch of the fence region in multiple of "tile widths". For Tile X this field must be programmed to a multiple of 512B "003" is the minimum value and for Tile Y this field must be programmed to a multiple of 128B "000" is the minimum value. 000h 128B 001h 256B 002h 384B 003h 512B 004h 640B 005h 768B 006h 896B 007h 1024B ... 3FFh 128KB 7FFh 256KB
31:12	RW	0x0	default/uncore	**FENCELOW:** Bits 31:12 of the starting Graphics Address of the fence region. Fence regions must be aligned to 4KB. This address represents the first 4KB page of the fence region Lower Bound is included in the fence region. Graphics Address is the offset within GMADR.

Bit	Type	Default Value	RST Type	Description
1:1	RW	0x0	default/uncore	**TILE:** This field specifies the spatial ordering of QW within tiles. 0b - Consectuvie SWords 32B sequenced in the X direction 1b - Consectuvie OWords 16B sequenced in the Y direction
0:0	RW	0x0	default/uncore	**FENCEVAL:** This field specifies whether or not this fence register defines a fence region. 0b - FENCE INVALID 1b - FENCE VALID

MPGFXTRK_CR_FENCE26_0_2_0_GTTMMADR

B/D/F/Type: 0/2/0/GTTMMADR
Address Offset: 0x1000d0
Access: 64 bits
Size: RW

Fence Register

Bit	Type	Default Value	RST Type	Description
63:44	RW	0x0	default/uncore	**FENCEUP:** Bits 31:12 of the ending Graphics Address of the fence region. Fence regions must be aligned to a 4KB page. This address represents the last 4KB page of the fence region Upper Bound is included in the fence region. Graphics Address is the offset within GMADR space.
42:32	RW	0x0	default/uncore	**PITCH:** This field specifies the width pitch of the fence region in multiple of "tile widths". For Tile X this field must be programmed to a multiple of 512B "003" is the minimum value and for Tile Y this field must be programmed to a multiple of 128B "000" is the minimum value. 000h 128B 001h 256B 002h 384B 003h 512B 004h 640B 005h 768B 006h 896B 007h 1024B ... 3FFh 128KB 7FFh 256KB
31:12	RW	0x0	default/uncore	**FENCELOW:** Bits 31:12 of the starting Graphics Address of the fence region. Fence regions must be aligned to 4KB. This address represents the first 4KB page of the fence region Lower Bound is included in the fence region. Graphics Address is the offset within GMADR.

Bit	Type	Default Value	RST Type	Description
1:1	RW	0x0	default/uncore	**TILE:** This field specifies the spatial ordering of QW within tiles. 0b - Consectuvie SWords 32B sequenced in the X direction 1b - Consectuvie OWords 16B sequenced in the Y direction
0:0	RW	0x0	default/uncore	**FENCEVAL:** This field specifies whether or not this fence register defines a fence region. 0b - FENCE INVALID 1b - FENCE VALID

MPGFXTRK_CR_FENCE27_0_2_0_GTTMMADR

B/D/F/Type: 0/2/0/GTTMMADR
Address Offset: 0x1000d8
Access: 64 bits
Size: RW

Fence Register

Bit	Type	Default Value	RST Type	Description
63:44	RW	0x0	default/uncore	**FENCEUP:** Bits 31:12 of the ending Graphics Address of the fence region. Fence regions must be aligned to a 4KB page. This address represents the last 4KB page of the fence region Upper Bound is included in the fence region. Graphics Address is the offset within GMADR space.
42:32	RW	0x0	default/uncore	**PITCH:** This field specifies the width pitch of the fence region in multiple of "tile widths". For Tile X this field must be programmed to a multiple of 512B "003" is the minimum value and for Tile Y this field must be programmed to a multiple of 128B "000" is the minimum value. 000h 128B 001h 256B 002h 384B 003h 512B 004h 640B 005h 768B 006h 896B 007h 1024B ... 3FFh 128KB 7FFh 256KB
31:12	RW	0x0	default/uncore	**FENCELOW:** Bits 31:12 of the starting Graphics Address of the fence region. Fence regions must be aligned to 4KB. This address represents the first 4KB page of the fence region Lower Bound is included in the fence region. Graphics Address is the offset within GMADR.

Bit	Type	Default Value	RST Type	Description
1:1	RW	0x0	default/uncore	**TILE:** This field specifies the spatial ordering of QW within tiles. 0b - Consectuvie SWords 32B sequenced in the X direction 1b - Consectuvie OWords 16B sequenced in the Y direction
0:0	RW	0x0	default/uncore	**FENCEVAL:** This field specifies whether or not this fence register defines a fence region. 0b - FENCE INVALID 1b - FENCE VALID

MPGFXTRK_CR_FENCE28_0_2_0_GTTMMADR

B/D/F/Type: 0/2/0/GTTMMADR
Address Offset: 0x1000e0
Access: 64 bits
Size: RW

Fence Register

Bit	Type	Default Value	RST Type	Description
63:44	RW	0x0	default/uncore	**FENCEUP:** Bits 31:12 of the ending Graphics Address of the fence region. Fence regions must be aligned to a 4KB page. This address represents the last 4KB page of the fence region Upper Bound is included in the fence region. Graphics Address is the offset within GMADR space.
42:32	RW	0x0	default/uncore	**PITCH:** This field specifies the width pitch of the fence region in multiple of "tile widths". For Tile X this field must be programmed to a multiple of 512B "003" is the minimum value and for Tile Y this field must be programmed to a multiple of 128B "000" is the minimum value. 000h 128B 001h 256B 002h 384B 003h 512B 004h 640B 005h 768B 006h 896B 007h 1024B ... 3FFh 128KB 7FFh 256KB
31:12	RW	0x0	default/uncore	**FENCELOW:** Bits 31:12 of the starting Graphics Address of the fence region. Fence regions must be aligned to 4KB. This address represents the first 4KB page of the fence region Lower Bound is included in the fence region. Graphics Address is the offset within GMADR.

Bit	Type	Default Value	RST Type	Description
1:1	RW	0x0	default/uncore	**TILE:** This field specifies the spatial ordering of QW within tiles. 0b - Consectuvie SWords 32B sequenced in the X direction 1b - Consectuvie OWords 16B sequenced in the Y direction
0:0	RW	0x0	default/uncore	**FENCEVAL:** This field specifies whether or not this fence register defines a fence region. 0b - FENCE INVALID 1b - FENCE VALID

MPGFXTRK_CR_FENCE29_0_2_0_GTTMMADR

B/D/F/Type: 0/2/0/GTTMMADR
Address Offset: 0x1000e8
Access: 64 bits
Size: RW

Fence Register

Bit	Type	Default Value	RST Type	Description
63:44	RW	0x0	default/uncore	**FENCEUP:** Bits 31:12 of the ending Graphics Address of the fence region. Fence regions must be aligned to a 4KB page. This address represents the last 4KB page of the fence region Upper Bound is included in the fence region. Graphics Address is the offset within GMADR space.
42:32	RW	0x0	default/uncore	**PITCH:** This field specifies the width pitch of the fence region in multiple of "tile widths". For Tile X this field must be programmed to a multiple of 512B "003" is the minimum value and for Tile Y this field must be programmed to a multiple of 128B "000" is the minimum value. 000h 128B 001h 256B 002h 384B 003h 512B 004h 640B 005h 768B 006h 896B 007h 1024B ... 3FFh 128KB 7FFh 256KB
31:12	RW	0x0	default/uncore	**FENCELOW:** Bits 31:12 of the starting Graphics Address of the fence region. Fence regions must be aligned to 4KB. This address represents the first 4KB page of the fence region Lower Bound is included in the fence region. Graphics Address is the offset within GMADR.

Bit	Type	Default Value	RST Type	Description
1:1	RW	0x0	default/uncore	**TILE:** This field specifies the spatial ordering of QW within tiles. 0b - Consectuvie SWords 32B sequenced in the X direction 1b - Consectuvie OWords 16B sequenced in the Y direction
0:0	RW	0x0	default/uncore	**FENCEVAL:** This field specifies whether or not this fence register defines a fence region. 0b - FENCE INVALID 1b - FENCE VALID

MPGFXTRK_CR_FENCE30_0_2_0_GTTMMADR

B/D/F/Type: 0/2/0/GTTMMADR
Address Offset: 0x1000f0
Access: 64 bits
Size: RW

Fence Register

Bit	Type	Default Value	RST Type	Description
63:44	RW	0x0	default/uncore	**FENCEUP:** Bits 31:12 of the ending Graphics Address of the fence region. Fence regions must be aligned to a 4KB page. This address represents the last 4KB page of the fence region Upper Bound is included in the fence region. Graphics Address is the offset within GMADR space.
42:32	RW	0x0	default/uncore	**PITCH:** This field specifies the width pitch of the fence region in multiple of "tile widths". For Tile X this field must be programmed to a multiple of 512B "003" is the minimum value and for Tile Y this field must be programmed to a multiple of 128B "000" is the minimum value. 000h 128B 001h 256B 002h 384B 003h 512B 004h 640B 005h 768B 006h 896B 007h 1024B ... 3FFh 128KB 7FFh 256KB
31:12	RW	0x0	default/uncore	**FENCELOW:** Bits 31:12 of the starting Graphics Address of the fence region. Fence regions must be aligned to 4KB. This address represents the first 4KB page of the fence region Lower Bound is included in the fence region. Graphics Address is the offset within GMADR.

Bit	Type	Default Value	RST Type	Description
1:1	RW	0x0	default/uncore	**TILE:** This field specifies the spatial ordering of QW within tiles. 0b - Consectuvie SWords 32B sequenced in the X direction 1b - Consectuvie OWords 16B sequenced in the Y direction
0:0	RW	0x0	default/uncore	**FENCEVAL:** This field specifies whether or not this fence register defines a fence region. 0b - FENCE INVALID 1b - FENCE VALID

MPGFXTRK_CR_FENCE31_0_2_0_GTTMMADR

B/D/F/Type: 0/2/0/GTTMMADR
Address Offset: 0x1000f8
Access: 64 bits
Size: RW

Fence Register

Bit	Type	Default Value	RST Type	Description
63:44	RW	0x0	default/uncore	**FENCEUP:** Bits 31:12 of the ending Graphics Address of the fence region. Fence regions must be aligned to a 4KB page. This address represents the last 4KB page of the fence region Upper Bound is included in the fence region. Graphics Address is the offset within GMADR space.
42:32	RW	0x0	default/uncore	**PITCH:** This field specifies the width pitch of the fence region in multiple of "tile widths". For Tile X this field must be programmed to a multiple of 512B "003" is the minimum value and for Tile Y this field must be programmed to a multiple of 128B "000" is the minimum value. 000h 128B 001h 256B 002h 384B 003h 512B 004h 640B 005h 768B 006h 896B 007h 1024B ... 3FFh 128KB 7FFh 256KB
31:12	RW	0x0	default/uncore	**FENCELOW:** Bits 31:12 of the starting Graphics Address of the fence region. Fence regions must be aligned to 4KB. This address represents the first 4KB page of the fence region Lower Bound is included in the fence region. Graphics Address is the offset within GMADR.

Bit	Type	Default Value	RST Type	Description
1:1	RW	0x0	default/uncore	**TILE:** This field specifies the spatial ordering of QW within tiles. 0b - Consectuvie SWords 32B sequenced in the X direction 1b - Consectuvie OWords 16B sequenced in the Y direction
0:0	RW	0x0	default/uncore	**FENCEVAL:** This field specifies whether or not this fence register defines a fence region. 0b - FENCE INVALID 1b - FENCE VALID

MPGFXTRK_CR_DPFC_CONTROL_SA_0_2_0_GTTMMADR

B/D/F/Type: 0/2/0/GTTMMADR

Address Offset: 0x100100

Access: 32 bits

Size: RW

This register contains control bits related to Display Frame Buffer Compression Host Invalidation in System Agent.

Bit	Type	Default Value	RST Type	Description
29:29	RW	0x0	default/uncore/flr	**CPUFNCEN:** 0: Display Buffer is not in a CPU fence. No modifications are allowed from CPU to the Display Buffer. 1: Display Buffer exists in a CPU fence.
4:0	RW	0x0	default/uncore/flr	**CPUFNCNUM:** This field specifies the CPU visible FENCE number corresponding to the placement of the uncompressed frame buffer.

MPGFXTRK_CR_DPFC_CPU_FENCE_OFFSET_0_2_0_GTTMMADR

B/D/F/Type: 0/2/0/GTTMMADR

Address Offset: 0x100104

Access: 32 bits

Size: RW

This register contains control bits related to Display Frame Buffer Compression Host Invalidation in System Agent.

Bit	Type	Default Value	RST Type	Description
21:0	RW	0x0	default/uncore/flr	**YFNCDISP:** Y offset from the CPU fence to the Display Buffer base

MPGFXTRK_CR_TILECTL_0_2_0_GTTMMADR

B/D/F/Type: 0/2/0/GTTMMADR

Address Offset: 0x101000

Access: 32 bits

Size: RW

This register contains control functionality related to GFX Aperture Tiling.

Bit	Type	Default Value	RST Type	Description
3:3	RW	0x0	default/uncore	**BKSNPDIS:** This bit allows to disable backsnoop requests as a result of IA requests to the Aperture. 0: Snoops are sent for IA requests that hit the Aperture 1: Snoops are never sent for IA requests that hit the Aperture
2:2	RW	0x0	default/uncore	**DISTLBPF:** On Tile Y GFX TLB miss, the cacheline read from the GTT contains 16 PTEs. This bit indicates whether all 16 PTEs are required to be cached or only the PTE that was requested. 0 - Prefetch 15 entries into the GFX TLB in addition to the demand-based fetch for Tile Y 1 - Disable TLB prefetch for Tile Y
1:0	RW	0x0	default/uncore	**SWZCTL:** This register location is updated via GFX Driver prior to enabling DRAM accesses. The Driver needs to obtain the need for memory address swizzling via DRAM configuration registers and set the following bits. 00b - No Address Swizzling 01b - Address bit 6 needs to be swizzled for tiled surfaces 10b - Reserved 11b - Reserved

MPGFXTRK_CR_GFX_FLSH_CNTL_0_2_0_GTTMMADR

B/D/F/Type: 0/2/0/GTTMMADR
Address Offset: 0x101008
Access: 32 bits
Size: WO

This register is used to flush GFX TLBs in the System Agent.

Bit	Type	Default Value	RST Type	Description
0:0	WO	0x0	default/uncore	**GFX_FLSH_CNTL:** A CPU write to this bit flushes the GFX TLBs in the System Agent. The data associated with the write is discarded and a read returns all 0s.

MPGFXTRK_CR_MTOLUD_0_2_0_GTTMMADR

B/D/F/Type: 0/2/0/GTTMMADR

Address Offset: 0x108000

Access: 32 bits

Size: RO_V

This 32 bit register defines the Top of Low Usable DRAM. TSEG, GTT Graphics memory and Graphics Stolen Memory are within the DRAM space defined. From the top, the Host optionally claims 1 to 64MBs of DRAM for internal graphics if enabled, 1or 2MB of DRAM for GTT Graphics Stolen Memory if enabled and 1, 2, or 8 MB of DRAM for TSEG if enabled.

Programming Example:

C1DRB3 is set to 4GB

TSEG is enabled and TSEG size is set to 1MB

Internal Graphics is enabled, and Graphics Mode Select is set to 32MB

GTT Graphics Stolen Memory Size set to 2MB

BIOS knows the OS requires 1G of PCI space.

BIOS also knows the range from 0FEC00000h to 0FFFFFFFFh is not usable by the system. This 20MB range at the very top of addressable memory is lost to APIC and LT.

According to the above equation, TOLUD is originally calculated to: 4GB 100000000h

The system memory requirements are: 4GB max addressable space - 1GB pci space - 35MB lost memory 3GB - 35MB minimum granularity 0ECB00000h

Since 0ECB00000h PCI and other system requirements is less than 100000000h, TOLUD should be programmed to ECBh.

These bits are LT lockable.

Bit	Type	Default Value	RST Type	Description
31:20	RO_V	0x1	default/uncore	**TOLUD:** This register contains bits 31 to 20 of an address one byte above the maximum DRAM memory below 4G that is usable by the operating system. Address bits 31 down to 20 programmed to 01h implies a minimum memory size of 1MB. Configuration software must set this value to the smaller of the following 2 choices: maximum amount memory in the system minus ME stolen memory plus one byte or the minimum address allocated for PCI memory. Address bits 19:0 are assumed to be 00000h for the purposes of address comparison. The Host interface positively decodes an address towards DRAM if the incoming address is less than the value programmed in this register. The Top of Low Usable DRAM is the lowest address above both Graphics Stolen memory and Tseg. BIOS determines the base of Graphics Stolen Memory by subtracting the Graphics Stolen Memory Size from TOLUD and further decrements by Tseg size to determine base of Tseg. This register must be 1MB aligned when reclaim is enabled.
0:0	RO_V	0x0	default/uncore	**LOCK:** This bit will lock all writeable settings in this register, including itself.

MPGFXTRK_CR_MGGC_0_2_0_GTTMMADR

B/D/F/Type: 0/2/0/GTTMMADR
Address Offset: 0x108040
Access: 16 bits
Size: RO_V

All the bits in this register are LT lockable.

Bit	Type	Default Value	RST Type	Description
14:14	RO_V	0x0	default/uncore	**VAMEN:** Enables the use of the iGFX enbines for Versatile Acceleration. 1 - iGFX engines are in Versatile Acceleration Mode. Device 2 Class Code is 048000h. 0 - iGFX engines are in iGFX Mode. Device 2 Class Code is 030000h.
9:8	RO_V	0x0	default/uncore	**GGMS:** This field is used to select the amount of Main Memory that is pre-allocated to support the Internal Graphics Translation Table. The BIOS ensures that memory is pre-allocated only when Internal graphics is enabled. GSM is assumed to be a contiguous physical DRAM space with DSM, and BIOS needs to allocate a contiguous memory chunk. Hardware will derive the base of GSM from DSM only using the GSM size programmed in the register. Hardware functionality in case of programming this value to Reserved is not guaranteed. 0h:No Preallocated Memory 1h:1MB of Preallocated Memory 2h:2MB of Preallocated Memory 3h:Reserved

Bit	Type	Default Value	RST Type	Description
7:3	RO_V	0x5	default/uncore	**GMS:** This field is used to select the amount of Main Memory that is pre-allocated to support the Internal Graphics device in VGA non-linear and Native linear modes. The BIOS ensures that memory is pre-allocated only when Internal graphics is enabled. Hardware does not clear or set any of these bits automatically based on IGD being disabledenabled. BIOS Requirement: BIOS must not set this field to 0h if IVD bit 1 of this register is 0. 00h:0MB 01h:32MB 02h:64MB 03h:96MB 04h:128MB 05h:160MB 06h:192MB 07h:224MB 08h:256MB 09h:288MB 0Ah:320MB 0Bh:352MB 0Ch:384MB 0Dh:416MB 0Eh:448MB 0Fh:480MB 10h:512MB Other:Reserved

Bit	Type	Default Value	RST Type	Description
1:1	RO_V	0x0	default/uncore	**IVD:** 0: Enable. Device 2 IGD claims VGA memory and IO cycles, the Sub-Class Code within Device 2 Class Code register is 00. 1: Disable. Device 2 IGD does not claim VGA cycles Mem and IO, and the Sub- Class Code field within Device 2 function 0 Class Code register is 80. BIOS Requirement: BIOS must not set this bit to 0 if the GMS field bits 7:3 of this register pre-allocates no memory. This bit MUST be set to 1 if Device 2 is disabled either via a fuse or fuse override CAPID0AIGD 1 or via a register DEVEN3 0. 0:Enable 1:Disable
0:0	RO_V	0x0	default/uncore	**GGCLCK:** When set to 1b, this bit will lock all bits in this register.

MPGFXTRK_CR_MTOUUD_0_2_0_GTTMMADR

B/D/F/Type: 0/2/0/GTTMMADR
Address Offset: 0x108080
Access: 64 bits
Size: RO_V

This 64 bit register defines the Top of Upper Usable DRAM.

Configuration software must set this value to TOM minus all ME stolen memory if reclaim is disabled. If reclaim is enabled, this value must be set to reclaim limit 1byte, 1MB aligned, since reclaim limit is 1MB aligned. Address bits 19:0 are assumed to be 0000000h for the purposes of address comparison. The Host interface positively decodes an address towards DRAM if the incoming address is less than the value programmed in this register and greater than or equal to 4GB.

BIOS Restriction: Minimum value for TOUUD is 4GB.

These bits are LT lockable.

Bit	Type	Default Value	RST Type	Description
38:20	RO_V	0x0	default/uncore	**TOUUD:** This register contains bits 38 to 20 of an address one byte above the maximum DRAM memory above 4G that is usable by the operating system. Configuration software must set this value to TOM minus all ME stolen memory if reclaim is disabled. If reclaim is enabled, this value must be set to reclaim limit 1MB aligned since reclaim limit 1byte is 1MB aligned. Address bits 19:0 are assumed to be 0000000h for the purposes of address comparison. The Host interface positively decodes an address towards DRAM if the incoming address is less than the value programmed in this register and greater than 4GB.
0:0	RO_V	0x0	default/uncore	**LOCK:** This bit will lock all writeable settings in this register, including itself.

MPGFXTRK_CR_MBDSM_0_2_0_GTTMMADR

B/D/F/Type: 0/2/0/GTTMMADR

Address Offset: 0x1080c0

Access: 32 bits

Size: RO_V

This register contains the base address of graphics data stolen DRAM memory. BIOS determines the base of graphics data stolen memory by subtracting the graphics data stolen memory size PCI Device 0 offset 52 bits 7:4 from TOLUD PCI Device 0 offset BC bits 31:20.

Bit	Type	Default Value	RST Type	Description
31:20	RO_V	0x0	default/uncore	**BDSM:** This register contains bits 31 to 20 of the base address of stolen DRAM memory. BIOS determines the base of graphics stolen memory by subtracting the graphics stolen memory size PCI Device 0 offset 52 bits 6:4 from TOLUD PCI Device 0 offset BC bits 31:20.
0:0	RO_V	0x0	default/uncore	**LOCK:** This bit will lock all writeable settings in this register, including itself.

MPGFXTRK_CR_MBGSM_0_2_0_GTTMMADR

B/D/F/Type: 0/2/0/GTTMMADR
Address Offset: 0x108100
Access: 32 bits
Size: RO_V

This register contains the base address of stolen DRAM memory for the GTT. BIOS determines the base of GTT stolen memory by subtracting the GTT graphics stolen memory size PCI Device 0 offset 52 bits 9:8 from the Graphics Base of Data Stolen Memory PCI Device 0 offset B0 bits 31:20.

Bit	Type	Default Value	RST Type	Description
31:20	RO_V	0x1	default/uncore	**BGSM:** This register contains the base address of stolen DRAM memory for the GTT. BIOS determines the base of GTT stolen memory by subtracting the GTT graphics stolen memory size PCI Device 0 offset 52 bits 11:8 from the Graphics Base of Data Stolen Memory PCI Device 0 offset B0 bits 31:20.
0:0	RO_V	0x0	default/uncore	**LOCK:** This bit will lock all writeable settings in this register, including itself.

MPGFXTRK_CR_MPLMBASE_REG_0_2_0_GTTMMADR

B/D/F/Type: 0/2/0/GTTMMADR

Address Offset: 0x1081c0

Access: 32 bits

Size: RO_V

This register is always treated as RO for implementations not supporting protected low memory region PLMR field reported as Clear in the Capability register.

The alignment of the protected low memory region base depends on the number of reserved bits N:0 of this register. Software may determine N by writing all 1s to this register, and finding the most significant zero bit position with 0 in the value read back from the register. Bits N:0 of this register is decoded by hardware as all 0s.

Software must setup the protected low memory region below 4GB.

Bit	Type	Default Value	RST Type	Description
31:20	RO_V	0x0	default/uncore	**PLMB:** This register specifies the base of protected low-memory region in system memory.

MPGFXTRK_CR_MPHMBASE_REG_0_2_0_GTTMMADR

B/D/F/Type: 0/2/0/GTTMMADR

Address Offset: 0x108240

Access: 64 bits

Size: RO_V

This register is always treated as RO for implementations not supporting protected high memory region PHMR field reported as Clear in the Capability register.

The alignment of the protected high memory region base depends on the number of reserved bits N:0 of this register. Software may determine N by writing all 1's to this register, and finding most significant zero bit position below host address width HAW in the value read back from the register. Bits N:0 of this register are decoded by hardware as all 0s.

Software may setup the protected high memory region either above or below 4GB.

Bit	Type	Default Value	RST Type	Description
38:20	RO_V	0x0	default/uncore	**PHMB:** This register specifies the base of protected high memory region in system memory. Hardware ignores, and does not implement, bits 63:HAW, where HAW is the host address width.

MPGFXTRK_CR_MPHMLIMIT_REG_0_2_0_GTTMMADR

B/D/F/Type: 0/2/0/GTTMMADR

Address Offset: 0x108280

Access: 64 bits

Size: RO_V

Register to set up the limit address of DMA-protected high-memory region. This register must be set up before enabling protected memory through PMENREG, and must not be updated when protected memory regions are enabled.

This register is always treated as RO for implementations not supporting protected high memory region PHMR field reported as Clear in the Capability register.

The alignment of the protected high memory region limit depends on the number of reserved bits N:0 of this register. Software may determine the value of N by writing all 1's to this register, and finding most significant zero bit position below host address width HAW in the value read back from the register. Bits N:0 of the limit register is decoded by hardware as all 1s.

The protected high-memory base limit registers functions as follows.

- Programming the protected low-memory base and limit registers with the same value in bits HAW:N1 specifies a protected low-memory region of size 2^N1 bytes.
- Programming the protected high-memory limit register with a value less than the protected high-memory base register disables the protected high-memory region.

Software must not modify this register when protected memory regions are enabled PRS field Set in PMENREG.

Bit	Type	Default Value	RST Type	Description
38:20	RO_V	0x0	default/uncore	**PHML:** This register specifies the last host physical address of the DMA-protected high-memory region in system memory. Hardware ignores and does not implement bits 63:HAW, where HAW is the host address width.

MPGFXTRK_CR_MGCMD_REG_0_2_0_GTTMMADR

B/D/F/Type: 0/2/0/GTTMMADR

Address Offset: 0x108300

Access: 32 bits

Size: RO_V

Register to control remapping hardware. If multiple control fields in this register need to be modified, software must serialize the modifications through multiple writes to this register.

Bit	Type	Default Value	RST Type	Description
31:31	RO_V	0x0	default/uncore	**TE:** Software writes to this field to request hardware to enabledisable DMA-remapping: 0: Disable DMA remapping 1: Enable DMA remapping Hardware reports the status of the translation enable operation through the TES field in the Global Status register. There may be active DMA requests in the platform when software updates this field. Hardware must enable or disable remapping logic only at determinstic transaction boundaries, so that any in-flight transaction is either subject to remapping or not at all. Hardware implementations supporting DMA draining must drain any in-flight DMA readwrite requests queued within the Root-Complex before completing the translation enable command and reflecting the status of the command through the TES field in the Global Status register. The value returned on a read of this field is undefined.

Bit	Type	Default Value	RST Type	Description
30:30	RO_V	0x0	default/uncore	**SRTP:** Software sets this field to setupdate the root-entry table pointer used by hardware. The root-entry table pointer is specified through the Root-entry Table Address RTAREG register. Hardware reports the status of the "Set Root Table Pointer" operation through the RTPS field in the Global Status register. The "Set Root Table Pointer" operation must be performed before enabling or re-enabling after disabling DMA remapping through the TE field. After a "Set Root Table Pointer" operation, software must globally invalidate the context cache and then globally invalidate of IOTLB. This is required to ensure hardware uses only the remapping structures referenced by the new root table pointer, and not stale cached entries. While DMA remapping hardware is active, software may update the root table pointer through this field. However, to ensure valid in-flight DMA requests are deterministically remapped, software must ensure that the structures referenced by the new root table pointer are programmed to provide the same remapping results as the structures referenced by the previous root-table pointer. Clearing this bit has no effect. The value returned on read of this field is undefined.
29:29	RO_V	0x0	default/uncore	**SFL:** This field is valid only for implementations supporting advanced fault logging. Software sets this field to request hardware to setupdate the fault-log pointer used by hardware. The fault-log pointer is specified through Advanced Fault Log register. Hardware reports the status of the 'Set Fault Log' operation through the FLS field in the Global Status register. The fault log pointer must be set before enabling advanced fault logging through EAFL field. Once advanced fault logging is enabled, the fault log pointer may be updated through this field while DMA remapping is active. Clearing this bit has no effect. The value returned on read of this field is undefined.

Bit	Type	Default Value	RST Type	Description
28:28	RO_V	0x0	default/uncore	**EAFL:** This field is valid only for implementations supporting advanced fault logging. Software writes to this field to request hardware to enable or disable advanced fault logging: 0: Disable advanced fault logging. In this case, translation faults are reported through the Fault Recording registers. 1: Enable use of memory-resident fault log. When enabled, translation faults are recorded in the memory-resident log. The fault log pointer must be set in hardware through the SFL field before enabling advaned fault logging. Hardware reports the status of the advaned fault logging enable operation through the AFLS field in the Global Status register. The value returned on read of this field is undefined.
27:27	RO_V	0x0	default/uncore	**WBF:** This bit is valid only for implementations requiring write buffer flushing. Software sets this field to request that hardware flush the Root-Complex internal write buffers. This is done to ensure any updates to the memory-resident remapping structures are not held in any internal write posting buffers. Hardware reports the status of the write buffer flushing operation through the WBFS field in the Global Status register. Clearing this bit has no effect. The value returned on a read of this field is undefined.
26:26	RO_V	0x0	default/uncore	**QIE:** This field is valid only for implementations supporting queued invalidations. Software writes to this field to enable or disable queued invalidations. 0: Disable queued invalidations. 1: Enable use of queued invalidations. Hardware reports the status of queued invalidation enable operation through QIES field in the Global Status register. The value returned on a read of this field is undefined.

Bit	Type	Default Value	RST Type	Description
25:25	RO_V	0x0	default/uncore	**IRE:** This field is valid only for implementations supporting interrupt remapping. 0: Disable interrupt-remapping hardware 1: Enable interrupt-remapping hardware Hardware reports the status of the interrupt remapping enable operation through the IRES field in the Global Status register. There may be active interrupt requests in the platform when software updates this field. Hardware must enable or disable interrupt-remapping logic only at deterministic transaction boundaries, so that any in-flight interrupts are either subject to remapping or not at all. Hardware implementations must drain any in-flight interrupts requests queued in the Root-Complex before completing the interrupt-remapping enable command and reflecting the status of the command through the IRES field in the Global Status register. The value returned on a read of this field is undefined.
24:24	RO_V	0x0	default/uncore	**SIRTP:** This field is valid only for implementations supporting interrupt-remapping. Software sets this field to setupdate the interrupt remapping table pointer used by hardware. The interrupt remapping table pointer is specified through the Interrupt Remapping Table Address IRTAREG register. Hardware reports the status of the 'Set Interrupt Remap Table Pointer' operation through the IRTPS field in the Global Status register. The 'Set Interrupt Remap Table Pointer' operation must be performed before enabling or re-enabling after disabling interrupt-remapping hardware through the IRE field. After a 'Set Interrupt Remap Table Pointer' operation, software must globally invalidate the interrupt entry cache. This is required to ensure hardware uses only the interrupt-remapping entries referenced by the new interrupt remap table pointer, and not any stale cached entries. While interrupt remapping is active, software may update the interrupt remapping table pointer through this field. However, to ensure valid in-flight interrupt requests are deterministically remapped, software must ensure that the structures referenced by the new interrupt remap table pointer are programmed to provide the same remapping results as the structures referenced by the previous interrupt remap table pointer. Clearing this bit has no effect. The value returned on a read of this field is undefined.

Bit	Type	Default Value	RST Type	Description
23:23	RO_V	0x0	default/uncore	**CFI:** This field is valid only for Intel64 implementations supporting interrupt-remapping. Software writes to this field to enable or disable Compatibility Format interrupts on Intel64 platforms. The value in this field is effective only when interrupt-remapping is enabled and Extended Interrupt Mode x2APIC mode is not enabled. 0: Block Compatibility format interrupts. 1: Process Compatibility format interrupts as pass-through bypass interrupt remapping. Hardware reports the status of updating this field through the CFIS field in the Global Status register. The value returned on a read of this field is undefined.

MPGFXTRK_CR_MEMRR_BASE_0_2_0_GTTMMADR

B/D/F/Type: 0/2/0/GTTMMADR
Address Offset: 0x108340
Access: 64 bits
Size: RW_L

The EMRR range is used to protect Xucode memory from unauthorized reads and writes. Any IO access to this range is aborted. This register controls the location of the EMRR range by indicating its starting address.

It functions in tandem with the EMRR mask register.

Bit	Type	Default Value	RST Type	Description
38:12	RW_L	0x0	default/uncore	**RANGE_BASE:** This field corresponds to bits 38:12 of the base address memory range which is allocated to EMRR memory.

MPGFXTRK_CR_MEMRR_MASK_0_2_0_GTTMMADR

B/D/F/Type: 0/2/0/GTTMMADR
Address Offset: 0x108380
Access: 64 bits
Size: RW_KL/RW_L

This register controls the size of the EMRR range by indicating which address bits must match the EMRR base register value.

Bit	Type	Default Value	RST Type	Description
38:12	RW_L	0x0	default/uncore	**RANGE_MASK:** This field indicates which address bits must match EMRR base in order to qualify as an EMRR access.
11:11	RW_L	0x0	default/uncore	**RANGE_EN:** Indicates whether the EMRR range is enabled and valid.
10:10	RW_KL	0x0	default/uncore	**LOCK:** Setting this bit locks all writeable settings in this register, including itself.

PCU Registers

PCU_CR_GT_THREAD_STATUS_0_2_0_GTTMMADR

B/D/F/Type: 0/2/0/GTTMMADR

Address Offset: 0x13805c

Access: 32 bits

Size: RO_V

Per-thread status register.

Bit	Type	Default Value	RST Type	Description
31:31	RO_V	0x0	default/uncore	**Thread Active(THREAD_ACTIVE):** Virtual signal from the ROB signaling that the thread is active.
30:30	RO_V	0x0	default/uncore	**Vote Request(VOTE_REQUEST):** The PCU HW will compute this field which indicates that the thread requests a P-State voting right. This bit will be set on TC0-TC1 and cleared on TC1E-TC7.
25:25	RO_V	0x0	default/uncore	Write to Both Threads(WRITE_TO_BOTH_THREADS): Command bit that causes the data for all RW fields in this register to be written to both thread copies for that core.
18:16	RO_V	0x7	default/uncore	**Thread Wish C-State Result(THREAD_WISH_RESULT):** This field contains the PCU's response to the Thread's C-State Wish. It is equal to the minimum of THREADWISHSTATE in this register and MAXALLOWEDCSTATE IO Register.

Bit	Type	Default Value	RST Type	Description
15:12	RO_V	0xf	default/uncore	**Thread Wish Sub-State(THREAD_WISH_SUB_STATE):** This field specifies the Thread C-State Sub-State that the thread wants to be in. It is updated by the thread during the WISH phase. Sub-States for TC1 0000b TC1KEEPVR 0001b TC1LOOSEVR C1E Sub-States for TC7 0000b TC7GRADUALLLCSHUTDOWN 0001b TC7CLOSELLCATONCE C7S Sub-States for RC6RC7 0000b RC6
11:8	RO_V	0xf	default/uncore	**Thread Wish C-State(THREAD_WISH_STATE):** This field specifies the Thread C-State that the thread wants to be in. It is updated by the thread during the WISH phase. UCODE will update this field with the parameters of the MWAIT instruction.
6:4	RO_V	0x0	default/uncore	**Thread Power Down State(THREAD_TPD_STATE):** The default value for the Thread Power Down TPD State is Normal 000b.
2:0	RO_V	0x7	default/uncore	**Thread C- State(THREAD_STATE):** The default value of the Thread C-State is is TRST 111b. NOTE: For unsupported C-states, PCODE will demote the request to the next higher power C-state.

PCU_CR_GT_CORE_STATUS_0_2_0_GTTMMADR

B/D/F/Type: 0/2/0/GTTMMADR
Address Offset: 0x138060
Access: 32 bits
Size: RO_V

Per-core status register.

Bit	Type	Default Value	RST Type	Description
31:31	RO_V	0x0	default/uncore	**Core Active(CORE_ACTIVE):** Virtual signal from the ROB, signaling that the core is active. During CORERESET, ROB will indicate that the core is active but this indication will be filtered by the HW i.e. COREACTIVE will remain 0b.
30:30	RO_V	0x0	default/uncore	**Wakeup Request(WAKEUP_REQUEST):** Virtual signal from the NCUGTFIFO specifying that there is an event pending for the IAGT Core. This can only happen if the core blocked events. PCODE can set this bit by writing to the PCODEWAKEUPREQUEST IO Register.
28:28	RO_V	0x1	default/uncore	**Core in C3/C6(CORE_IN_C3_C6):** This bit indicates that the dispatcher has put a core in C3C6. NOTE: On wakeup, this bit must de-asserted after at least COREACTIVE is asserted or CORESTATE was cleared.
27:27	RO_V	0x0	default/uncore	**Probe Mode Done Indication(PROBE_MODE_DONE):** This bit is used by UCODE to inform the PCU that the Probe Mode sequence that it was running is done.
26:26	RO_V	0x0	default/uncore	**Disable Wakeup Request(DISABLE_WAKEUP_REQ):** This bit is only applicable to the GT Core. When this bit is set, the GTFIFO should be blocked and should not send any wakeup request. This is done by writing the value of this bit to IMPHCRFIFOCTLPSMIBLK.

Bit	Type	Default Value	RST Type	Description
25:25	RO_V	0x0	default/uncore	**S1 acknowledge from PMA(S1_ACK):** This bit is set by an UpS virtual signal from the PMA when the core has acknowledged the S1 CPD request from pcode.
24:24	RO_V	0x1	default/uncore	**Block Request(PM_BLOCK_REQ):** When this bit is set to 1b, the GTFIFO should be blocked. This is done by writing to the lower byte only of IMPHCRFIFOCTL a value of 01h. When this bit is set to 0b, the GTFIFO should be unblocked. This is done by writing to the lower byte only of IMPHCRFIFOCTL a value of 00h.
23:23	RO_V	0x1	default/uncore	**Render C6 Entry(RC6_ENTRY):** This field only has meaning for the GT register instance. It is a don't care for the IA register instances. For GT: This bit indicates that the PMBLOCKREQ is due to RC6 entry. If it is clear, then it indicates that the PMBLOCKREQ is due to CPD flow.
15:12	RO_V	0xf	default/uncore	**Core Wish Sub C-State(CORE_WISH_SUB_STATE):** This field specifies the coordinated Sub C-State that the core wants to be in. In case the two threads have different wish states, the field should contain sub C-state of the thread with the smaller wish state. In case the two threads have identical wish states, this field should contain a bit-wise AND of each thread's wish sub C-state. The thread wish sub C-state is given in PCUCRTHREADSTATUS.
11:8	RO_V	0xf	default/uncore	**Core Wish C-State(CORE_WISH_STATE):** This field specifies the coordinated Core C-State that the core wants to be in. It is defined as the minimum of the thread wish CST in case two threads are active. The thread wish state is given in PCUCRTHREADSTATUS.
6:4	RO_V	0x0	default/uncore	**Core Power Down State(CORE_CPD_STATE):** The default value for the Core Power Down CPD State is Normal 000b.

Bit	Type	Default Value	RST Type	Description
2:0	RO_V	0x7	default/uncore	**Core C-State(CORE_STATE):** The default value of the Core C-State is is CRST 111b. HW will clear this field to 000b on C-State exit when Core clock is ungated. NOTE: For unsupported C-states, PCODE will demote the request to the next higher power C-state.

PCU_CR_GT_SLICE_INFO_0_2_0_GTTMMADR

B/D/F/Type: 0/2/0/GTTMMADR
Address Offset: 0x138064
Access: 32 bits
Size: RO_V/RW/RW_V

Control and Status register for GT half slice mode.

Bit	Type	Default Value	RST Type	Description
3:3	RW	0x1	default/uncore	**Slice Selection(LSLICESEL):** Number of GT slices to power at the next C6 exit. 0: Power only the main GT slice 1: Power both GT slices
2:2	RW_V	0x0	default/uncore	**Auto wake(AUTOWAKE):** Control automatic wake of GT 0: Normal mode; wake when iMPH indicates FIFO not empty 1: Wake immediately after C6 entry
1:0	RO_V	0x0	default/uncore	**Slice Status(LSLICESTAT):** Status of GT power planes 00: GT is powered off C6 or not yet booted 10: GT has main slice powered 11: GT has both slices powered

PCU_CR_GT_THREAD_P_REQ_0_2_0_GTTMMADR

B/D/F/Type: 0/2/0/GTTMMADR
Address Offset: 0x13807c
Access: 32 bits
Size: RW

Thread Target P-state Value. All values in this register are updated by Ucode as a result of WRMSR requests.

Bit	Type	Default Value	RST Type	Description
31:31	RW	0x0	default/uncore	**Turbo Disable(TURBO_DISABLE):** The Turbo Disable bit is determined by SW. SW access to IA32PERFCTLMSRTURBODIS is routed to this field by Ucode. NOTE: If Turbo is disabled for ANY thread, it will prevent turbo for ALL threads.
30:24	RW	0x0	default/uncore	**Maximum P-state Request(P_STATE_REQ):** This field indicates the maximum P-State request in units of 100MHz. It is determined by SW. SW access to IA32PERFCTLMSRPREQ is routed to this field by Ucode.
23:18	RW	0x0	default/uncore	**P-State Offset(P_STATE_OFFSET):** This field defines the number of steps that Energy Efficient P-State is allowed to fall in units of 100 MHz. It is determined by Ucode as follows: IF PSTCONFIGCONTROLMSRLEGACYSUPPORTENABLE 1b PSTATEOFST PSTCONFIGCONTROLMSRPSTATEOFST ELSE PSTATEOFST IA32PERFCTLMSRPSTATEOFST
17:14	RW	0x0	default/uncore	**Energy Efficiency Policy(ENERGY_EFFICIENCY_POLICY):** The energy efficiency policy is determined by SW. SW access to IA32ENERGYPERFORMANCEBIASMSRPERFPOLICY is routed to this field by Ucode.

PCU_CR_GT_GFX_RC6_0_2_0_GTTMMADR

B/D/F/Type: 0/2/0/GTTMMADR
Address Offset: 0x138108
Access: 32 bits
Size: RO_FW

This register contains the total RC6 residency time that GT was in since boot. The counter will wrap around. The time is given in units of 1.28 uSec.

Bit	Type	Default Value	RST Type	Description
31:0	RO_FW	0x0	default/uncore	**Residency Time(RC6):** Value

PCU_CR_GTDRIVER_MAILBOX_INTERFACE_0_2_0_GTTMMADR

B/D/F/Type: 0/2/0/GTTMMADR
Address Offset: 0x138124
Access: 32 bits
Size: RW1S/RW_V

Control and Status register for the GFX-DRIVER-to-PCODE mailbox. This mailbox is implemented as a means for tuning parameters for specific GFX workloads.

This register is used in conjunction with GTDRIVERMAILBOXDATA.

Bit	Type	Default Value	RST Type	Description
31:31	RW1S	0x0	default/uncore	**Run/Busy Indicator(RUN_BUSY):** SW may write to the mailbox registers only when RUNBUSY is cleared 0b. Setting RUNBUSY to 1b will create a Fast Path event. After setting this bit, SW will poll this bit until it is cleared. Alternatively, PCODE can generate an interrupt to SW via GTDRIVERP2GEVENTS. PCODE will clear RUNBUSY after updating the mailbox registers with the result and error code.
28:8	RW_V	0x0	default/uncore	**Address Control(ADDR_CNTL):** This field contains the address associated with specific commands.
7:0	RW_V	0x0	default/uncore	**Command Code(COMMAND):** This field contains the SW request command or the PCODE response code, depending on the setting of RUNBUSY. Command Encodings: 00h ZERO 01h CMD_CONFIG 02h WRITE_PCS 03h READ_PCS 04h Unavailable 05h Unavailable 06h Unavailable 07h Unavailable

Bit	Type	Default Value	RST Type	Description
				08h WRITE_MIN_FREQUENCY_TABLE 09h READ_MIN_FREQUENCY_TABLE 0Ah CLEAR_RCX_RESIDENCE_COUNTERS 0Bh READ_RING_RATIOS 0Ch READ_OVERCLOCK_PARAMS 0Dh READ_PCU_MISC_CONFIG 0Eh WRITE_PCU_MISC_CONFIG 0Fh READ_PKGC_PMREQ_FORMAT 10h READ_PCU_CR_D_COMP (ULT only) 11h WRITE_PCU_CR_D_COMP (ULT only) 12h WRITE_MCHBAR 13h Unavailable 14h Unavailable 15h Unavailable 16h READ_REQUESTED_DUTY_CYCLE 17h DE_WRITE_FREQ_REQ 18h Unavailable 19h Unavailable 1Ah DYNAMIC_DUTY_CYCLE_CONTROL 1Bh Unavailable 1Ch Unavailable 1Dh Unavailable 1Eh Unavailable 1Fh Unavailable Completion Encodings: 00h SUCCESS 01h ILLEGAL_CMD 02h TIMEOUT 03h ILLEGAL_DATA 10h MIN_FREQUENCY_TABLE_GT_RATIO_OUT_OF_RANGE

PCU_CR_GTDRIVER_MAILBOX_DATA0_0_2_0_GTTMMADR

B/D/F/Type: 0/2/0/GTTMMADR

Address Offset: 0x138128

Access: 32 bits

Size: RW_V

Data register for the GFX-DRIVER-to-PCODE mailbox. This mailbox is implemented as a means for tuning parameters for specific GFX workloads.

This register is used in conjunction with GTDRIVERMAILBOXINTERFACE.

Bit	Type	Default Value	RST Type	Description
31:0	RW_V	0x0	default/uncore	**Data(DATA):** This field contains the data associated with specific commands.

PCU_CR_GTDRIVER_MAILBOX_DATA1_0_2_0_GTTMMADR

B/D/F/Type: 0/2/0/GTTMMADR

Address Offset: 0x13812c

Access: 32 bits

Size: RW_V

Data register for the GFX-DRIVER-to-PCODE mailbox. This mailbox is implemented as a means for tuning parameters for specific GFX workloads.

This register is used in conjunction with GTDRIVERMAILBOXINTERFACE.

Bit	Type	Default Value	RST Type	Description
31:0	RW_V	0x0	default/uncore	**Data(DATA):** This field contains the data associated with specific commands.

PCU_CR_GTDRIVER_G2P_EVENTS_0_2_0_GTTMMADR

B/D/F/Type: 0/2/0/GTTMMADR
Address Offset: 0x138164
Access: 32 bits
Size: RW1S

This extended capability allows the GFX Driver to send a request to PCODE.

The GFX Driver will set the appropriate bit in this register to 1b when it wants to generate an event to PCODE. This will generate a Fast Path event.

PCODE will clear the appropriate bit in this register after servicing the request.

Bit	Type	Default Value	RST Type	Description
7:7	RW1S	0x0	default/uncore	**Event 7(EVENT7):** Placeholder for Event
6:6	RW1S	0x0	default/uncore	**Event 6(EVENT6):** Placeholder for Event
5:5	RW1S	0x0	default/uncore	**Event 5(EVENT5):**
4:4	RW1S	0x0	default/uncore	**Event 4(EVENT4):** Placeholder for Event
3:3	RW1S	0x0	default/uncore	**Event 3(EVENT3):** Placeholder for Event
2:2	RW1S	0x0	default/uncore	**Event 2(EVENT2):** Placeholder for Event
1:1	RW1S	0x0	default/uncore	**Event 1(EVENT1):** Placeholder for Event
0:0	RW1S	0x0	default/uncore	**Event 0(EVENT0):** Placeholder for Event

SA Registers

GSA_CR_GTSP0_0_2_0_GTTMMADR

B/D/F/Type: 0/2/0/GTTMMADR

Address Offset: 0x130040

Access: 32 bits

Size: RO_V/RW/RO

This is an alternate name for the LCPLL Control Register (LCPLL_CTL). See the LCPLL_CTL definition for bit field definitions.

Bit	Type	Default Value	RST Type	Description
31:0	RW	0x0	default/uncore/flr	LCPLL Control bits. See the LCPLL_CTL definition for more details.

GSA_CR_GTSP1_0_2_0_GTTMMADR

B/D/F/Type: 0/2/0/GTTMMADR

Address Offset: 0x130044

Access: 32 bits

Size: RW

Bit	Type	Default Value	RST Type	Description
31:16	RW	0x0	default/uncore/flr	**GT scratch pad(GTSP)**
15:0	RW	0x0	default/uncore/flr	**Multiple Force Wake** GT programs this field with the multiple force wake status. Software reads this field to find the status. Refer to MULTIFORCEWAKE 0xA188 register description for the usage.

GSA_CR_GTSP2_0_2_0_GTTMMADR

B/D/F/Type: 0/2/0/GTTMMADR
Address Offset: 0x130048
Access: 32 bits
Size: RW

Bit	Type	Default Value	RST Type	Description
31:0	RW	0x0	default/uncore/flr	GT scratch pad(GTSP):

GSA_CR_GTSP3_0_2_0_GTTMMADR

B/D/F/Type: 0/2/0/GTTMMADR

Address Offset: 0x13004c

Access: 32 bits

Size: RW

Bit	Type	Default Value	RST Type	Description
31:0	RW	0x0	default/uncore/flr	GT scratch pad(GTSP):

GSA_CR_GTSP4_0_2_0_GTTMMADR

B/D/F/Type: 0/2/0/GTTMMADR
Address Offset: 0x130050
Access: 32 bits
Size: RW

Bit	Type	Default Value	RST Type	Description
31:0	RW	0x0	default/uncore/flr	GT scratch pad(GTSP):

GSA_CR_GTSP5_0_2_0_GTTMMADR

B/D/F/Type: 0/2/0/GTTMMADR

Address Offset: 0x130054

Access: 32 bits

Size: RW

Bit	Type	Default Value	RST Type	Description
31:0	RW	0x0	default/uncore/flr	GT scratch pad(GTSP):

GSA_CR_GTSP6_0_2_0_GTTMMADR

B/D/F/Type: 0/2/0/GTTMMADR
Address Offset: 0x130058
Access: 32 bits
Size: RW

Bit	Type	Default Value	RST Type	Description
31:0	RW	0x0	default/uncore/flr	GT scratch pad(GTSP):

GSA_CR_GTSP7_0_2_0_GTTMMADR

B/D/F/Type: 0/2/0/GTTMMADR

Address Offset: 0x13005c

Access: 32 bits

Size: RW

Bit	Type	Default Value	RST Type	Description
31:0	RW	0x0	default/uncore/flr	GT scratch pad(GTSP):

Bit 0 is used as a graphics software to PSMI handler semaphore. Software programs bit 0 = 1 when it does not want the PSMI handler to run.

GSA_CR_GTFORCEAWAKE_0_2_0_GTTMMADR

B/D/F/Type: 0/2/0/GTTMMADR

Address Offset: 0x130090

Access: 32 bits

Size: RW

This register is used as a way for the driver to make sure GT does not initiate any power-down sequence when required.

SW including the driver is assumed not to write to this register under normal operation.

Bit	Type	Default Value	RST Type	Description
0:0	RW	0x0	default/uncore/flr	**Force Awake(FORCEAWAKE):** This bit is used as a way for the driver to make sure GT does not initiate any power-down sequence. The driver will send GT a message that causes GT to go ahead and set this bit. When the driver reads this bit set, it knows that GT will not initiate any power-down sequence. At a later time, the driver will send a message to GT informing it that it may be powered down. This will cause GT to reset this bit.

IGFX Registers

Register Name	Symbol	Type	Addr	Size	Register Info
GSA_CR_VID2_0_2_0_PCI	VID2	CFG	0x0	16	RO
GSA_CR_DID2_0_2_0_PCI	DID2	CFG	0x2	16	RO_V/RO_FW
UNCORE_CR_PCICMD_0_2_0_PCI	PCICMD	CFG	0x4	16	RW/RO
GSA_CR_PCISTS2_0_2_0_PCI	PCISTS2	CFG	0x6	16	RO_V/RO
GSA_CR_RID2_0_2_0_PCI	RID2	CFG	0x8	8	RO_FW
GSA_CR_CC_0_2_0_PCI	CC	CFG	0x9	24	RO_V/RO
GSA_CR_CLS_0_2_0_PCI	CLS	CFG	0xc	8	RO
GSA_CR_MLT2_0_2_0_PCI	MLT2	CFG	0xd	8	RO
GSA_CR_HDR2_0_2_0_PCI	HDR2	CFG	0xe	8	RO
GSA_CR_BIST_0_2_0_PCI	BIST	CFG	0xf	8	RO
UNCORE_CR_GTTMMADR_0_2_0_PCI	GTTMMADR	CFG	0x10	64	RW/RO
UNCORE_CR_GMADR_0_2_0_PCI	GMADR	CFG	0x18	64	RW/RW_L/RO
UNCORE_CR_IOBAR_0_2_0_PCI	IOBAR	CFG	0x20	32	RW/RO
GSA_CR_SVID2_0_2_0_PCI	SVID2	CFG	0x2c	16	RW_O
GSA_CR_SID2_0_2_0_PCI	SID2	CFG	0x2e	16	RW_O
GSA_CR_ROMADR_0_2_0_PCI	ROMADR	CFG	0x30	32	RO
GSA_CR_CAPPOINT_0_2_0_PCI	CAPPOINT	CFG	0x34	8	RO_V
GSA_CR_INTRLINE_0_2_0_PCI	INTRLINE	CFG	0x3c	8	RW
GSA_CR_INTRPIN_0_2_0_PCI	INTRPIN	CFG	0x3d	8	RO
GSA_CR_MINGNT_0_2_0_PCI	MINGNT	CFG	0x3e	8	RO
GSA_CR_MAXLAT_0_2_0_PCI	MAXLAT	CFG	0x3f	8	RO
GSA_CR_CAPID0_0_2_0_PCI	CAPID0	CFG	0x40	16	RO
GSA_CR_CAPCTRL0_0_2_0_PCI	CAPCTRL0	CFG	0x42	16	RO
GSA_CR_CAPID0_A_0_2_0_PCI	CAPID0_A	CFG	0x44	32	RO_V
GSA_CR_CAPID0_B_0_2_0_PCI	CAPID0_B	CFG	0x48	32	RO_V
GSA_CR_MGGC0_0_2_0_PCI	MGGC0	CFG	0x50	16	RO_V
GSA_CR_DEVEN0_0_2_0_PCI	DEVEN0	CFG	0x54	32	RO_V/RO
GSA_CR_BDSM_0_2_0_PCI	BDSM	CFG	0x5c	32	RO_V
GSA_CR_HSRW_0_2_0_PCI	HSRW	CFG	0x60	16	RW
UNCORE_CR_MSAC_0_2_0_PCI	MSAC	CFG	0x62	8	RW/RW_K
GSA_CR_MSI_CAPID_0_2_0_PCI	MSI_CAPID	CFG	0x90	16	RO
GSA_CR_MC_0_2_0_PCI	MC	CFG	0x92	16	RW/RO
GSA_CR_MA_0_2_0_PCI	MA	CFG	0x94	32	RW/RO

Register Name	Symbol	Type	Addr	Size	Register Info
GSA_CR_MD_0_2_0_PCI	MD	CFG	0x98	16	RW
GSA_CR_AFCIDNP_0_2_0_PCI	AFCIDNP	CFG	0xa4	16	RO
GSA_CR_AFLC_0_2_0_PCI	AFLC	CFG	0xa6	16	RO
UNCORE_CR_AFCTL_0_2_0_PCI	AFCTL	CFG	0xa8	8	RW1S
GSA_CR_AFSTS_0_2_0_PCI	AFSTS	CFG	0xa9	8	RO
GSA_CR_PMCAPID_0_2_0_PCI	PMCAPID	CFG	0xd0	16	RO
GSA_CR_PMCAP_0_2_0_PCI	PMCAP	CFG	0xd2	16	RO
UNCORE_CR_PMCS_0_2_0_PCI	PMCS	CFG	0xd4	16	RO_V/RO
GSA_CR_SWSMI_0_2_0_PCI	SWSMI	CFG	0xe0	16	RW
GSA_CR_GSE_0_2_0_PCI	GSE	CFG	0xe4	32	RW
GSA_CR_SWSCI_0_2_0_PCI	SWSCI	CFG	0xe8	16	RW/RW_O
GSA_CR_SRID_0_2_0_PCI	SRID	CFG	0xf8	32	RO_FW
GSA_CR_ASLS_0_2_0_PCI	ASLS	CFG	0xfc	32	RW

GSA_CR_VID2_0_2_0_PCI

B/D/F/Type: 0/2/0/pci
Address Offset: 0x0
Access: 16 bits
Size: RO

This register combined with the Device Identification register uniquely identifies any PCI device.

Bit	Type	Default Value	RST Type	Description
15:0	RO	0x8086	default/uncore	**Vendor Identification Number(VID):** PCI standard identification for Intel.

GSA_CR_DID2_0_2_0_PCI

B/D/F/Type: 0/2/0/pci
Address Offset: 0x2
Access: 16 bits
Size: RO_V/RO_FW

This register combined with the Vendor Identification register uniquely identifies any PCI device.

This is a 16 bit value assigned to Graphics device.

Bit	Type	Default Value	RST Type	Description
15:4	RO_FW	0xc0	default/uncore	**Device Identification Number MSB(DID_MSB):** This is the upper part of a 16 bit value assigned to the Graphics device.
3:2	RO_V	0x0	default/uncore	**Device Identification Number - SKU(DID_SKU):** These are bits 3:2 of the 16 bit value assigned to the Graphics device. SKU 3:2 Desktop 00 Mobile 01 Server 10 Marketing spare 11 Versatile Acceleration 00 If MGGC0VAMEN then DID23:2 lt; 10b Else If CAPID0ADIDOE 1 then DID23:2 lt; DIDOVR1:0 Else DID23:2 lt; CAPID0ACDID
1:0	RO_V	0x2	default/uncore	**Device Identification Number LSB(DID_LSB):** This is the lower part of a 16 bit value assigned to the Graphics device.

UNCORE_CR_PCICMD_0_2_0_PCI

B/D/F/Type: 0/2/0/pci
Address Offset: 0x4
Size: 16 bits
Access: RW/RO
Global: YES

This 16-bit register provides basic control over the IGD's ability to respond to PCI cycles. The PCICMD Register in the IGD disables the IGD PCI compliant master accesses to main memory.

Bit	Type	Default Value	RST Type	Description
10:10	RW	0x0	default/uncore/flr	**Interrupt Disable(INTDIS):** This bit disables the device from asserting INTx#. 0: Enable the assertion of this device's INTx# signal. 1: Disable the assertion of this device's INTx# signal. DOINTx messages will not be sent to DMI.
9:9	RO	0x0	default/uncore	**Fast Back-to-Back(FB2B):** Not Implemented. Hardwired to 0.
8:8	RO	0x0	default/uncore	**SERR Enable(SEN):** Not Implemented. Hardwired to 0.
7:7	RO	0x0	default/uncore	**Wait Cycle Control(WCC):** Not Implemented. Hardwired to 0.
6:6	RO	0x0	default/uncore	**Parity Error Enable(PER):** Not Implemented. Hardwired to 0. Since the IGD belongs to the category of devices that does not corrupt programs or data in system memory or hard drives, the IGD ignores any parity error that it detects and continues with normal operation.
5:5	RO	0x0	default/uncore	**Video Palette Snooping(VPS):** This bit is hardwired to 0 to disable snooping.
4:4	RO	0x0	default/uncore	**Memory Write and Invalidate Enable(MWIE):** Hardwired to 0. The IGD does not support memory write and invalidate commands.
3:3	RO	0x0	default/uncore	**Special Cycle Enable(SCE):** This bit is hardwired to 0. The IGD ignores Special cycles.

Bit	Type	Default Value	RST Type	Description
2:2	RW	0x0	default/uncore/flr	**Bus Master Enable(BME):** 0: Disable IGD bus mastering. 1: Enable the IGD to function as a PCI compliant master.
0:0	RW	0x0	default/uncore/flr	**I/O Access Enable(IOAE):** This bit controls the IGD's response to IO space accesses. 0: Disable. 1: Enable.

GSA_CR_PCISTS2_0_2_0_PCI

B/D/F/Type: 0/2/0/pci
Address Offset: 0x6
Access: 16 bits
Size: RO_V/RO

PCISTS is a 16-bit status register that reports the occurrence of a PCI compliant master abort and PCI compliant target abort.

PCISTS also indicates the DEVSEL# timing that has been set by the IGD.

Bit	Type	Default Value	RST Type	Description
15:15	RO	0x0	default/uncore	**Detected Parity Error(DPE):** Since the IGD does not detect parity, this bit is always hardwired to 0.
14:14	RO	0x0	default/uncore	**Signaled System Error(SSE):** The IGD never asserts SERR#, therefore this bit is hardwired to 0.
13:13	RO	0x0	default/uncore	**Received Master Abort Status(RMAS):** The IGD never gets a Master Abort, therefore this bit is hardwired to 0.
12:12	RO	0x0	default/uncore	**Received Target Abort Status(RTAS):** The IGD never gets a Target Abort, therefore this bit is hardwired to 0.
11:11	RO	0x0	default/uncore	**Signaled Target Abort Status(STAS):** Hardwired to 0. The IGD does not use target abort semantics.
10:9	RO	0x0	default/uncore	**DEVSEL Timing(DEVT):** NA. These bits are hardwired to "00".
8:8	RO	0x0	default/uncore	**Master Data Parity Error Detected(DPD):** Since Parity Error Response is hardwired to disabled and the IGD does not do any parity detection, this bit is hardwired to 0.
7:7	RO	0x1	default/uncore	**Fast Back-to-Back(FB2B):** Hardwired to 1. The IGD accepts fast back-to-back when the transactions are not to the same agent.
6:6	RO	0x0	default/uncore	**User Defined Format(UDF):** Hardwired to 0.
5:5	RO	0x0	default/uncore	**66 MHz PCI Capable(C66):** NA - Hardwired to 0.
4:4	RO	0x1	default/uncore	**Capability List(CLIST):** This bit is set to 1 to indicate that the register at 34h provides an offset into the function's PCI Configuration Space containing a pointer to the location of the first item in the list.

Bit	Type	Default Value	RST Type	Description
3:3	RO_V	0x0	default/uncore	**Interrupt Status(INTSTS):** This bit reflects the state of the interrupt in the device. Only when the Interrupt Disable bit in the command register is a 0 and this Interrupt Status bit is a 1, will the devices INTx# signal be asserted.

GSA_CR_RID2_0_2_0_PCI

B/D/F/Type: 0/2/0/pci
Address Offset: 0x8
Access: 8 bits
Size: RO_FW

This register contains the revision number for Device #2 Functions 0.

These bits are read only and writes to this register have no effect.

For the A-0 Stepping, this value is 00h

Bit	Type	Default Value	RST Type	Description
7:4	RO_FW	0x0	default/uncore	**Revision Identification Number MSB(RID_MSB):** Four MSB of RID
3:0	RO_FW	0x0	default/uncore	**Revision Identification Number(RID):** Four LSB of RID

GSA_CR_CC_0_2_0_PCI

B/D/F/Type: 0/2/0/pci
Address Offset: 0x9
Access: 24 bits
Size: RO_V/RO

This register contains the device programming interface information related to the Sub-Class Code and Base Class Code definition for the IGD. This register also contains the Base Class Code and the function sub-class in relation to the Base Class Code.

Bit	Type	Default Value	RST Type	Description
23:16	RO_V	0x3	default/uncore	**Base Class Code(BCC):** This is an 8-bit value that indicates the base class code. When MGGC0VAMEN is 0 this code has the value 03h, indicating a Display Controller. When MGGC0VAMEN is 1 this code has the value 04h, indicating a Multimedia Device.
15:8	RO_V	0x0	default/uncore	**Sub-Class Code(SUBCC):** When MGGC0VAMEN is 0 this value will be determined based on Device 0 GGC register, GMS and IVD fields. 00h: VGA compatible 80h: Non VGA GMS "00h" or IVD "1b" When MGGC0VAMEN is 1, this value is 80h, indicating other multimedia device.
7:0	RO	0x0	default/uncore	**Programming Interface(PI):** When MGGC0VAMEN is 0 this value is 00h, indicating a Display Controller. When MGGC0VAMEN is 1 this value is 00h, indicating a NOP.

GSA_CR_CLS_0_2_0_PCI

B/D/F/Type: 0/2/0/pci
Address Offset: 0xc
Access: 8 bits
Size: RO

The IGD does not support this register as a PCI slave.

Bit	Type	Default Value	RST Type	Description
7:0	RO	0x0	default/uncore	**Cache Line Size(CLS):** This field is hardwired to 0s. The IGD as a PCI compliant master does not use the Memory Write and Invalidate command and, in general, does not perform operations based on cache line size.

GSA_CR_MLT2_0_2_0_PCI

B/D/F/Type: 0/2/0/pci
Address Offset: 0xd
Access: 8 bits
Size: RO

The IGD does not support the programmability of the master latency timer because it does not perform bursts.

Bit	Type	Default Value	RST Type	Description
7:0	RO	0x0	default/uncore	**Master Latency Timer Count Value(MLTCV):** Hardwired to 0s.

GSA_CR_HDR2_0_2_0_PCI

B/D/F/Type: 0/2/0/pci
Address Offset: 0xe
Access: 8 bits
Size: RO

This register contains the Header Type of the IGD.

Bit	Type	Default Value	RST Type	Description
7:7	RO	0x0	default/uncore	**Multi Function Status(MFUNC):** Indicates if the device is a Multi-Function Device. The Value of this register is hardwired to 0.
6:0	RO	0x0	default/uncore	**Header Code(H):** This is a 7-bit value that indicates the Header Code for the IGD. This code has the value 00h, indicating a type 0 configuration space format.

GSA_CR_BIST_0_2_0_PCI

B/D/F/Type: 0/2/0/pci
Address Offset: 0xf
Access: 8 bits
Size: RO

This register is used for control and status of Built In Self Test BIST.

Bit	Type	Default Value	RST Type	Description
7:7	RO	0x0	default/uncore	**BIST Supported(BISTS):** BIST is not supported. This bit is hardwired to 0.

UNCORE_CR_GTTMMADR_0_2_0_PCI

B/D/F/Type: 0/2/0/pci
Address Offset: 0x10
Size: 64 bits
Access: RW/RO
Global: YES

This register requests allocation for the combined Graphics Translation Table Modification Range and Memory Mapped Range. The range requires 4 MB combined for MMIO and Global GTT aperture, with 2MB of that used by MMIO and 2MB used by GTT. GTTADR will begin at GTTMMADR 2 MB while the MMIO base address will be the same as GTTMMADR.

For the Global GTT, this range is defined as a memory BAR in graphics device config space. It is an alias into which software is required to write Page Table Entry values PTEs. Software may read PTE values from the global Graphics Translation Table GTT. PTEs cannot be written directly into the global GTT memory area.

The device snoops writes to this region in order to invalidate any cached translations within the various TLB's implemented on-chip.

The allocation is for 4MB and the base address is defined by bits 38:22.

Bit	Type	Default Value	RST Type	Description
63:39	RW	0x0	default/uncore/flr	**Reserved for Memory Base Address(RSVDRW):** Must be set to 0 since addressing above 512GB is not supported.
38:22	RW	0x0	default/uncore/flr	**Memory Base Address(MBA):** Set by the OS, these bits correspond to address signals 38:22. 4MB combined for MMIO and Global GTT table aperture 2MB for MMIO and 2 MB for GTT.
21:4	RO	0x0	default/uncore	**Address Mask(ADM):** Hardwired to 0s to indicate at least 4MB address range.
3:3	RO	0x0	default/uncore	**Prefetchable Memory(PREFMEM):** Hardwired to 0 to prevent prefetching.
2:1	RO	0x2	default/uncore	**Memory Type(MEMTYP):** 00 : To indicate 32 bit base address 01: Reserved 10 : To indicate 64 bit base address 11: Reserved

UNCORE_CR_GMADR_0_2_0_PCI

B/D/F/Type: 0/2/0/pci
Address Offset: 0x18
Size: 64 bits
Access: RW/RW_L/RO
Global: YES

GMADR is the PCI aperture used by SW to access tiled GFX surfaces in a linear fashion.

Bit	Type	Default Value	RST Type	Description
63:39	RW	0x0	default/uncore/flr	**Reserved for Memory Base Address** Must be set to 0 since addressing above 512GB is not supported.
38:29	RW	0x0	default/uncore/flr	**Memory Base Address(MBA):** Memory Base Address MBA: Set by the OS, these bits correspond to address signals 38:29.
28:28	RW_L	0x0	default/uncore/flr	**512MB Address Mask(ADMSK512):** This Bit is either part of the Memory Base Address RW or part of the Address Mask RO, depending on the value of MSAC2:1. See MSAC Dev2, Func 0, offset 62h for details.
27:27	RW_L	0x0	default/uncore/flr	**256 MB Address Mask(ADMSK256):** This bit is either part of the Memory Base Address RW or part of the Address Mask RO, depending on the value of MSAC2:1. See MSAC Dev 2, Func 0, offset 62h for details.
26:4	RO	0x0	default/uncore	**Address Mask(ADM):** Hardwired to 0s to indicate at least 128MB address range.
3:3	RO	0x1	default/uncore	**Prefetchable Memory(PREFMEM):** Hardwired to 1 to enable prefetching.
2:1	RO	0x2	default/uncore	**Memory Type(MEMTYP):** 00: indicate 32-bit address. 10: Indicate 64-bit address

UNCORE_CR_IOBAR_0_2_0_PCI

B/D/F/Type: 0/2/0/pci
Address Offset: 0x20
Size: 32 bits
Access: RW/RO
Global: YES

This register provides the Base offset of the IO registers within Device #2. Bits 15:6 are programmable allowing the IO Base to be located anywhere in 16bit IO Address Space. Bits 2:1 are fixed and return zero; bit 0 is hardwired to a one indicating that 8 bytes of IO space are decoded. Access to the 8Bs of IO space is allowed in PM state D0 when IO Enable PCICMD bit 0 set. Access is disallowed in PM states D1-D3 or if IO Enable is clear or if Device #2 is turned off or if Internal graphics is disabled through the fuse or fuse override mechanisms.

Note that access to this IO BAR is independent of VGA functionality within Device #2.

If accesses to this IO bar is allowed then all 8, 16 or 32 bit IO cycles from IA cores that falls within the 8B are claimed.

Bit	Type	Default Value	RST Type	Description
15:6	RW	0x0	default/uncore/flr	**IO Base Address(IOBASE):** Set by the OS, these bits correspond to address signals 15:6.
2:1	RO	0x0	default/uncore	**Memory Type(MEMTYPE):** Hardwired to 0s to indicate 32-bit address.
0:0	RO	0x1	default/uncore	**Memory/IO Space(MIOS):** Hardwired to "1" to indicate IO space.

GSA_CR_SVID2_0_2_0_PCI

B/D/F/Type: 0/2/0/pci
Address Offset: 0x2c
Access: 16 bits
Size: RW_O

This register is used to uniquely identify the subsystem where the PCI device resides.

Bit	Type	Default Value	RST Type	Description
15:0	RW_O	0x0	default/uncore	**Subsystem Vendor ID(SUBVID):** This value is used to identify the vendor of the subsystem. This register should be programmed by BIOS during boot-up. Once written, this register becomes ReadOnly. This register can only be cleared by a Reset.

GSA_CR_SID2_0_2_0_PCI

B/D/F/Type: 0/2/0/pci
Address Offset: 0x2e
Access: 16 bits
Size: RW_O

This register is used to uniquely identify the subsystem where the PCI device resides.

Bit	Type	Default Value	RST Type	Description
15:0	RW_O	0x0	default/uncore	**Subsystem Identification(SUBID):** This value is used to identify a particular subsystem. This field should be programmed by BIOS during boot-up. Once written, this register becomes ReadOnly. This register can only be cleared by a Reset.

GSA_CR_ROMADR_0_2_0_PCI

B/D/F/Type: 0/2/0/pci
Address Offset: 0x30
Access: 32 bits
Size: RO

The IGD does not use a separate BIOS ROM, therefore this register is hardwired to 0s.

Bit	Type	Default Value	RST Type	Description
31:18	RO	0x0	default/uncore	**ROM Base Address(RBA):** Hardwired to 0's.
17:11	RO	0x0	default/uncore	**Address Mask(ADMSK):** Hardwired to 0s to indicate 256 KB address range.
0:0	RO	0x0	default/uncore	**ROM BIOS Enable(RBE):** 0: ROM not accessible.

GSA_CR_CAPPOINT_0_2_0_PCI

B/D/F/Type: 0/2/0/pci
Address Offset: 0x34
Access: 8 bits
Size: RO_V

This register points to a linked list of capabilities implemented by this device.

Bit	Type	Default Value	RST Type	Description
7:0	RO_V	0x90	default/uncore	**Capabilities Pointer Value(CPV):** This field contains an offset into the function's PCI Configuration Space for the first item in the New Capabilities Linked List, the MSI Capabilities ID registers at address 90h or the Power Management capability at D0h. This value is determined by the configuration in CAPL0.

GSA_CR_INTRLINE_0_2_0_PCI

B/D/F/Type: 0/2/0/pci
Address Offset: 0x3c
Access: 8 bits
Size: RW

This 8-bit register is used to communicate interrupt line routing information. It is readwrite and must be implemented by the device. POST software will write the routing information into this register as it initializes and configures the system.

The value in this register tells which input of the system interrupt controllers the device's interrupt pin is connected to. The device itself does not use this value, rather it is used by device drivers and operating systems to determine priority and vector information.

Bit	Type	Default Value	RST Type	Description
7:0	RW	0x0	default/uncore	**Interrupt Connection(INTCON):** Used to communicate interrupt line routing information. POST software writes the routing information into this register as it initializes and configures the system. The value in this register indicates to which input of the system interrupt controller the device's interrupt pin is connected.

GSA_CR_INTRPIN_0_2_0_PCI

B/D/F/Type: 0/2/0/pci
Address Offset: 0x3d
Access: 8 bits
Size: RO

This register tells which interrupt pin the device uses. The Integrated Graphics Device uses INTA#.

Bit	Type	Default Value	RST Type	Description
7:0	RO	0x1	default/uncore	**Interrupt Pin(INTPIN):** As a single function device, the IGD specifies INTA# as its interrupt pin. 01h: INTA#.

GSA_CR_MINGNT_0_2_0_PCI

B/D/F/Type: 0/2/0/pci
Address Offset: 0x3e
Access: 8 bits
Size: RO

The Integrated Graphics Device has no requirement for the settings of Latency Timers.

Bit	Type	Default Value	RST Type	Description
7:0	RO	0x0	default/uncore	**Minimum Grant Value(MGV):** The IGD does not burst as a PCI compliant master.

GSA_CR_MAXLAT_0_2_0_PCI

B/D/F/Type: 0/2/0/pci
Address Offset: 0x3f
Access: 8 bits
Size: RO

The Integrated Graphics Device has no requirement for the settings of Latency Timers.

Bit	Type	Default Value	RST Type	Description
7:0	RO	0x0	default/uncore	**Maximum Latency Value(MLV):** The IGD has no specific requirements for how often it needs to access the PCI bus.

GSA_CR_CAPID0_0_2_0_PCI

B/D/F/Type: 0/2/0/pci
Address Offset: 0x40
Access: 16 bits
Size: RO

Control of bits in this register are only required for customer visible SKU differentiation.

Bit	Type	Default Value	RST Type	Description
15:8	RO	0x0	default/uncore	**Next Capability Pointer(NEXT_CAP):** This field is hardwired to 00h indicating the end of the capabilities linked list.
7:0	RO	0x9	default/uncore	**Capability Identifier(CAP_ID):** This field has the value 1001b to identify the CAPID assigned by the PCI SIG for vendor dependent capability pointers.

GSA_CR_CAPCTRL0_0_2_0_PCI

B/D/F/Type: 0/2/0/pci
Address Offset: 0x42
Access: 16 bits
Size: RO

Control of bits in this register are only required for customer visible SKU differentiation.

Bit	Type	Default Value	RST Type	Description
11:8	RO	0x1	default/uncore	**CAPID Version(CAPID_VER):** This field has the value 0001b to identify the first revision of the CAPID register definition.
7:0	RO	0xc	default/uncore	**CAPID Length(CAPIDLEN):** This field has the value 0Ch to indicate the structure length 12 bytes.

GSA_CR_CAPID0_A_0_2_0_PCI

B/D/F/Type: 0/2/0/pci
Address Offset: 0x44
Size: 32 bits
Access: RO_V
Global: YES

Control of bits in this register are only required for customer visible SKU differentiation.

Bit	Type	Default Value	RST Type	Description
31:31	RO_V	0x0	default/uncore	**DHDAD:** 0: Display HD Audio Enabled 1: Display HD Audio Disabled
30:30	RO_V	0x0	default/uncore	**PEG12D:** 0: Device 1 Function 2 and associated memory spaces are accessible. 1: Device 1 Function 2 and associated memory and IO spaces are disabled by hardwiring the D1F2EN field, bit 1 of the Device Enable register, DEVEN Dev 0 Offset 54h to '0'.
29:29	RO_V	0x0	default/uncore	**PEG11D:** 0: Device 1 Function 1 and associated memory spaces are accessible. 1: Device 1 Function 1 and associated memory and IO spaces are disabled by hardwiring the D1F1EN field, bit 2 of the Device Enable register, DEVEN Dev 0 Offset 54h to '0'.
28:28	RO_V	0x0	default/uncore	**PEG10D:** 0: Device 1 Function 0 and associated memory spaces are accessible. 1: Device 1 Function 0 and associated memory and IO spaces are disabled by hardwiring the D1F0EN field, bit 3 of the Device Enable register, DEVEN Dev 0 Offset 54h to '0'.

Bit	Type	Default Value	RST Type	Description
27:27	RO_V	0x0	default/uncore	**PELWUD:** 0: Link width upconfig is supported. The CPU advertises upconfig capability using the data rate symbol in its TS2 training ordered sets during Configuration.Complete. The CPU responds to link width upconfigs initiated by the downstream device. 1: Link width upconfig is NOT supported. The CPU does not advertise upconfig capability using the data rate field in TS2 training ordered sets during Configuration.Complete. The CPU does not respond to link width upconfigs initiated by the downstream device.
26:26	RO_V	0x0	default/uncore	**DW:** 0: DMI x4 1: DMI x2
25:25	RO_V	0x0	default/uncore	**ECCDIS:** 0b ECC capable 1b Not ECC capable
24:24	RO_V	0x0	default/uncore	**FDEE:** 0: DRAM ECC optional via software. 1: DRAM ECC enabled. MCHBAR C0MISCCTL bit 0 and C1MISCCTL bit 0 are forced to 1 and Read-Only. Note that FDEE and ECCDIS must not both be set to 1.
23:23	RO_V	0x0	default/uncore	**VTDD:** 0: Enable VTd 1: Disable VTd
22:22	RO_V	0x0	default/uncore	**DMIG2DIS:** 0: Capable of running DMI in Gen 2 mode 1: Not capable of running DMI in Gen 2 mode
21:21	RO_V	0x0	default/uncore	**PEGG2DIS:** 0: Capable of running any of the PEG controllers in Gen 2 mode 1: Not capable of running any of the PEG controllers in Gen 2 mode

Bit	Type	Default Value	RST Type	Description
20:19	RO_V	0x0	default/uncore	**DDRSZ:** This field defines the maximum allowed memory size per channel. 00b Unlimited 16GB per channel 01b Maximum 8GB per channel 10b Maximum 2GB per channel 11b Maximum 0.5GB per channel
18:18	RO_V	0x0	default/uncore	**PCIE_RATIO_DIS:** 0 - Capable of changing PCIe ratio to support Bclk overclocking 1 - Not capable of changing PCIe ratio to support Bclk overclocking
17:17	RO_V	0x0	default/uncore	**D1NM:** 0: Part is capable of supporting 1n mode timings on the DDR interface. 1: Part is not capable of supporting 1n mode. Only supported timings are 2n or greater.
16:16	RO_V	0x0	default/uncore	**FUFRD:** Controls how much ULT information is available to debugging software via configuration transactions. 0: Full ULT information is available, including Wafer, X, and Y location. 1: ULT information is hidden. All ULT information will read out as zeros.
15:15	RO_V	0x0	default/uncore	**CDD:** 0: Camarillo Device enabled. 1: Camarillo Device disabled.
14:14	RO_V	0x0	default/uncore	**DDPCD:** Allows Dual Channel operation but only supports 1 DIMM per channel. 0: 2 DIMMs per channel enabled 1: 2 DIMMs per channel disabled. This setting hardwires bits 2 and 3 of the rank population field for each channel to zero. MCHBAR offset 260h, bits 22-23 for channel 0 and MCHBAR offset 660h, bits 22-23 for channel 1

Bit	Type	Default Value	RST Type	Description
13:13	RO_V	0x0	default/uncore	**X2APIC_EN:** Extended Interrupt Mode. 0b: Hardware does not support Extended APIC mode. 1b: Hardware supports Extended APIC mode.
12:12	RO_V	0x0	default/uncore	**PDCD:** 0: Capable of Dual Channels 1: Not Capable of Dual Channel - only single channel capable.
11:11	RO_V	0x0	default/uncore	**IGD:** 0: There is a graphics engine within this CPU. Internal Graphics Device Device 2 is enabled and all of its memory and IO spaces are accessible. Configuration cycles to Device 2 will be completed within the CPU. All non-SMM memory and IO accesses to VGA will be handled based on Memory and IO enables of Device 2 and IO registers within Device 2 and VGA Enable of the PCI to PCI bridge control register in Devices 1 and 6 If PCI Express GFX attach is supported. Graphics Memory is pre-allocated above TSEG Memory. 1: There is no graphics engine within this CPU. Internal Graphics Device Device 2 and all of its memory and IO functions are disabled. Configuration cycle targeted to Device 2 will be passed on to DMI. In addition, all clocks to internal graphics logic are turned off. All non-SMM memory and IO accesses to VGA will be handled based on VGA Enable of the PCI to PCI bridge control register in Devices 1 and 6. DEVEN 4:3 Device 0, offset 54h have no meaning. Device 2 Functions 0 and 1 are disabled and hidden.
10:10	RO_V	0x0	default/uncore	**DIDOE:** 0b Disable ability to override DID - For production 1b Enable ability to override DID - For samples only
9:8	RO_V	0x0	default/uncore	**CDID:** Identifier assigned to the core primary PCI device.The corresponding two bit capability ID programming is: 00: Desktop 01: Server 10: Mobile 11: Marketing Spare

Bit	Type	Default Value	RST Type	Description
7:4	RO_V	0x0	default/uncore	**CRID:** This is an 8-bit value that indicates the revision identification number for the Host Device 0. For the A-0 Stepping, this value is 00h.
3:3	RO_V	0x0	default/uncore	**DDR_OVERCLOCK:** Indicates support for DDR Overclocking.
2:2	RO_V	0x0	default/uncore	**OC_ENABLED_DSKU:** The default constant non-fuse value is zero. When the VDM sets this bit, we will apply OC if OCCTLSSKU points to DSKU.
1:1	RO_V	0x0	default/uncore	**DDR_WRTVREF:** Allow on-die DDR write Vref generation. PCODE will update this field with the value of FUSEDDRWRTVREF.
0:0	RO_V	0x0	default/uncore	**DDR3L_EN:** Allow DDR3L 1.35V DDR operation. PCODE will update this field with the value of FUSEDDR3LEN.

GSA_CR_CAPID0_B_0_2_0_PCI

B/D/F/Type: 0/2/0/pci
Address Offset: 0x48
Size: 32 bits
Access: RO_V
Global: YES

Control of bits in this register are only required for customer visible SKU differentiation.

Bit	Type	Default Value	RST Type	Description
31:31	RO_V	0x0	default/uncore	**SPARE31:** Reserved for future capabilities
30:30	RO_V	0x0	default/uncore	**OC_CTL_DSKU_DIS:** PCODE will update this field with the value of FUSEOCCTLSSKU, and then apply SSKU overrides. IA Overclocking controlled by SSKU rather than DSKU
29:29	RO_V	0x0	default/uncore	**OC_ENABLED:** PCODE will update this field with the value of FUSEOCENABLEDSSKU. 0b Over-clocking is Disabled 1b Over-clocking is Enabled If over-clocking is enabled, FUSEOCBINS contains how many bits of over-clocking are supported. The encoding is as follows: 0h Overclocking is Disabled 1h Max 1 bin of overclocking is supported 2h Max 2 bin of overclocking is supported 3h Max 3 bin of overclocking is supported 4h Max 4 bin of overclocking is supported 5h Max 5 bin of overclocking is supported 6h Max 6 bin of overclocking is supported 7h Unlimited If overclocking is not enabled, FUSEOCBINS is meaningless, and should be 0.

Bit	Type	Default Value	RST Type	Description
28:28	RO_V	0x0	default/uncore	**SMT:** This setting indicates whether or not the CPU is SMT capable.
27:25	RO_V	0x0	default/uncore	**CACHESZ:** This setting indicates the supporting cache sizes.
24:24	RO_V	0x0	default/uncore	**SOFTBIN:** CPU is Soft Bin capable via FUSESSKUSOFTBINEN and it was enabled by the PCH. Both the Mask and Value are needed to enable this. If enabled, pcode will use the final resolved values in CAPID.D-G except CAPID0E31:12 for frequencybranding instead of the baseline non-SSKU fused values.
23:21	RO_V	0x0	default/uncore	**PLL_REF100_CFG:** DDR3 Maximum Frequency Capability with 100 Memory. PCODE will update this field with the value of FUSEPLLREF100CFG and then apply SSKU overrides. Maximum allowed memory frequency with 100 MHz ref clk. Also serves as defeature. Unlike 133 MHz ref fuses, these are normal 3 bit field 0 - 100 MHz ref disabled 1 - upto DDR-1400 7 x 200 2 - upto DDR-1600 8 x 200 3 - upto DDR-1800 8 x 200 4 - upto DDR-2000 10 x 200 5 - upto DDR-2200 11 x 200 6 - upto DDR-2400 12 x 200 7 - no limit but still limited by DDRFREQ200 to 2600
20:20	RO_V	0x0	default/uncore	**PEGG3_DIS:** PCIe Gen 3 Disable fuse. This fuse will be strap selectablemodifiable to enable SSKU capabilities. This is a defeature fuse -- an un-programmed device should have PCIe Gen 3 capabilities enabled. 0: Capable of running any of the Gen 3-compliant PEG controllers in Gen 3 mode Devices 010, 011, 012 1: Not capable of running any of the PEG controllers in Gen 3 mode
19:19	RO_V	0x0	default/uncore	**PKGTYP:** This setting indicates the CPU Package Type.
18:18	RO_V	0x0	default/uncore	**ADDGFXEN:** 0 - Additive Graphics Disabled 1- Additive Graphics Enabled

Bit	Type	Default Value	RST Type	Description
17:17	RO_V	0x0	default/uncore	**ADDGFXCAP:** 0 - Capable of Additive Graphics 1 - Not capable of Additive Graphics
16:16	RO_V	0x0	default/uncore	**PEGX16D:** 0: Capable of x16 PEG Port 1: Not Capable of x16 PEG port, instead PEG limited to x8 and below. Causes PEG port to enable and train logical lanes 7:0 only. Logical lanes 15:8 are powered down unless in use by the other PEG port or the embedded Display Port, and the Max Link Width field of the Link Capability register reports x8 instead of x16. In the case of lane reversal, lanes 15:8 are active and lanes 7:0 are powered down.
15:12	RO_V	0x0	default/uncore	**SPARE15_12:** Reserved for future capabilities
10:8	RO_V	0x0	default/uncore	**SPARE10_8:** Reserved for future capabilities. Was HGKS.
7:7	RO_V	0x0	default/uncore	**DDD:** 1 - Production mode
6:4	RO_V	0x0	default/uncore	**DMFC:** This field controls which values may be written to the Memory Frequency Select field 6:4 of the Clocking Configuration registers MCHBAR Offset C00h. Any attempt to write an unsupported value will be ignored. 000: MC capable of DDR3 2667 2667 is the upper limit 001: MC capable of up to DDR3 2667 010: MC capable of up to DDR3 2400 011: MC capable of up to DDR3 2133 100: MC capable of up to DDR3 1867 101: MC capable of up to DDR3 1600 110: MC capable of up to DDR3 1333 111: MC capable of up to DDR3 1067
3:3	RO_V	0x0	default/uncore	**SPARE3:** Reserved for Future Capabilities
2:2	RO_V	0x0	default/uncore	**SPARE2:** Reserved for Future Capabilities

Bit	Type	Default Value	RST Type	Description
1:1	RO_V	0x0	default/uncore	**DPEGFX1:** This bit has no effect on Device 1 unless Device 1 is configured for at least two ports via PEG0CFGSEL strap. 0b All PEG port widths do not depend on their respective BCTRLVGAEN. 1b Each PEG port width is limited to x1 operation when its respective BCTRLVGAEN is set to 1b.
0:0	RO_V	0x0	default/uncore	**SPEGFX1:** This bit has no effect on Device 1 unless Device 1 is configured for a single port via PEG0CFGSEL strap. 0b PEG10 width does not depend on its BCTRLVGAEN. 1b PEG10 width is limited to x1 operation when its respective BCTRLVGAEN is set to 1b.

GSA_CR_MGGC0_0_2_0_PCI

B/D/F/Type: 0/2/0/pci
Address Offset: 0x50
Access: 16 bits
Size: RO_V

All the bits in this register are LT lockable.

Bit	Type	Default Value	RST Type	Description
14:14	RO_V	0x0	default/uncore	**Versatile Acceleration Mode Enable(VAMEN):** Enables the use of the iGFX enbines for Versatile Acceleration. 1 - iGFX engines are in Versatile Acceleration Mode. Device 2 Class Code is 048000h. 0 - iGFX engines are in iGFX Mode. Device 2 Class Code is 030000h.
9:8	RO_V	0x0	default/uncore	**GTT Graphics Memory Size(GGMS):** This field is used to select the amount of Main Memory that is pre-allocated to support the Internal Graphics Translation Table. The BIOS ensures that memory is pre-allocated only when Internal graphics is enabled. GSM is assumed to be a contiguous physical DRAM space with DSM, and BIOS needs to allocate a contiguous memory chunk. Hardware will derive the base of GSM from DSM only using the GSM size programmed in the register. Hardware functionality in case of programming this value to Reserved is not guaranteed.
7:3	RO_V	0x5	default/uncore	**Graphics Mode Select(GMS):** This field is used to select the amount of Main Memory that is pre-allocated to support the Internal Graphics device in VGA non-linear and Native linear modes. The BIOS ensures that memory is pre-allocated only when Internal graphics is enabled. This register is also LT lockable. Hardware does not clear or set any of these bits automatically based on IGD being disabledenabled. BIOS Requirement: BIOS must not set this field to 0h if IVD bit 1 of this

Bit	Type	Default Value	RST Type	Description
				register is 0.
1:1	RO_V	0x0	default/uncore	**IGD VGA Disable(IVD):** 0: Enable. Device 2 IGD claims VGA memory and IO cycles, the Sub-Class Code within Device 2 Class Code register is 00. 1: Disable. Device 2 IGD does not claim VGA cycles Mem and IO, and the Sub- Class Code field within Device 2 function 0 Class Code register is 80. BIOS Requirement: BIOS must not set this bit to 0 if the GMS field bits 7:3 of this register pre-allocates no memory. This bit MUST be set to 1 if Device 2 is disabled either via a fuse or fuse override CAPID0AIGD 1 or via a register DEVEN3 0. This register is locked by LT lock.
0:0	RO_V	0x0	default/uncore	**GGC Lock(GGCLCK):** When set to 1b, this bit will lock all bits in this register.

GSA_CR_DEVEN0_0_2_0_PCI

B/D/F/Type: 0/2/0/pci
Address Offset: 0x54
Size: 32 bits
Access: RO_V/RO
Global: YES

Allows for enablingdisabling of PCI devices and functions that are within the CPU package. The table below the bit definitions describes the behavior of all combinations of transactions to devices controlled by this register.

All the bits in this register are LT Lockable.

Bit	Type	Default Value	RST Type	Description
14:14	RO_V	0x0	default/uncore	**D7EN:** 0: Bus 0 Device 7 is disabled and not visible. 1: Bus 0 Device 7 is enabled and visible. Non-production BIOS code should provide a setup option to enable Bus 0 Device 7. When enabled, Bus 0 Device 7 must be initialized in accordance to standard PCI device initialization procedures.
7:7	RO_V	0x1	default/uncore	**D4EN:** 0: Bus 0 Device 4 is disabled and not visible. 1: Bus 0 Device 4 is enabled and visible. This bit will be set to 0b and remain 0b if Device 4 capability is disabled.
5:5	RO_V	0x1	default/uncore	**D3EN:** 0: Bus 0 Device 3 is disabled and hidden 1: Bus 0 Device 3 is enabled and visible This bit will be set to 0b and remain 0b if Device 3 capability is disabled.
4:4	RO_V	0x1	default/uncore	**D2EN:** 0: Bus 0 Device 2 is disabled and hidden 1: Bus 0 Device 2 is enabled and visible This bit will be set to 0b and remain 0b if Device 2 capability is disabled.

Bit	Type	Default Value	RST Type	Description
3:3	RO_V	0x1	default/uncore	**D1F0EN:** 0: Bus 0 Device 1 Function 0 is disabled and hidden. 1: Bus 0 Device 1 Function 0 is enabled and visible. This bit will be set to 0b and remain 0b if PEG10 capability is disabled.
2:2	RO_V	0x1	default/uncore	**D1F1EN:** 0: Bus 0 Device 1 Function 1 is disabled and hidden. 1: Bus 0 Device 1 Function 1 is enabled and visible. This bit will be set to 0b and remain 0b if: - PEG11 capability is disabled by fuses, OR - PEG11 is disabled by strap PEG0CFGSEL
1:1	RO_V	0x1	default/uncore	**D1F2EN:** 0: Bus 0 Device 1 Function 2 is disabled and hidden. 1: Bus 0 Device 1 Function 2 is enabled and visible. This bit will be set to 0b and remain 0b if: - PEG12 capability is disabled by fuses, OR - PEG12 is disabled by strap PEG0CFGSEL
0:0	RO	0x1	default/uncore	**D0EN:** Bus 0 Device 0 Function 0 may not be disabled and is therefore hardwired to 1.

GSA_CR_BDSM_0_2_0_PCI

B/D/F/Type: 0/2/0/pci
Address Offset: 0x5c
Access: 32 bits
Size: RO_V

This register contains the base address of graphics data stolen DRAM memory. BIOS determines the base of graphics data stolen memory by subtracting the graphics data stolen memory size PCI Device 0 offset 52 bits 7:4 from TOLUD PCI Device 0 offset BC bits 31:20.

Bit	Type	Default Value	RST Type	Description
31:20	RO_V	0x0	default/uncore	**Graphics Base of Stolen Memory (BDSM):** This register contains bits 31 to 20 of the base address of stolen DRAM memory. BIOS determines the base of graphics stolen memory by subtracting the graphics stolen memory size PCI Device 0 offset 52 bits 6:4 from TOLUD PCI Device 0 offset BC bits 31:20.
0:0	RO_V	0x0	default/uncore	**Lock(LOCK):** This bit will lock all writeable settings in this register, including itself.

GSA_CR_HSRW_0_2_0_PCI

B/D/F/Type: 0/2/0/pci
Address Offset: 0x60
Access: 16 bits
Size: RW

This register is reserved as a HW scratchpad.

Bit	Type	Default Value	RST Type	Description
15:0	RW	0x0	default/uncore/flr	**Reserved R/W(ReservedRW):** Reserved for future usage.

UNCORE_CR_MSAC_0_2_0_PCI

B/D/F/Type: 0/2/0/pci
Address Offset: 0x62
Size: 8 bits
Access: RW/RW_K
Global: YES

This register determines the size of the graphics memory aperture in function 0 and in the trusted space. Only the system BIOS will write this register based on pre- boot address allocation efforts, but the graphics may read this register to determine the correct aperture size. System BIOS needs to save this value on boot so that it can reset it correctly during S3 resume.

This register is LT locked, becomes read-only when trusted environment is launched.

Bit	Type	Default Value	RST Type	Description
7:4	RW	0x0	default/uncore	**Reserved R/W(RSVDRW):** Scratch Bits Only -- Have no physical effect on hardware
2:2	RW_K	0x0	default/uncore	**Untrusted Aperture Size High(LHSASH):** This field is used in conjunction with LHSASL. The description below is for both fields LHSASH LHSASL. 11b Bits 28:27 of GMADR are RO, allowing 512MB of GMADR 10b Illegal Programming 01b Bit 28 of GMADR is RW but bit 27 of GMADR is RO, allowing 256MB of GMADR 00b Bits 28:27 of GMADR are RW, allowing 128MB of GMADR
1:1	RW_K	0x1	default/uncore	**Untrusted Aperture Size Low(LHSASL):** This field is used in conjunction with LHSASH. The description below is for both fields LHSASH LHSASL. 11b Bits 28:27 of GMADR are RO, allowing 512MB of GMADR 10b Illegal Programming 01b Bit 28 of GMADR is RW but bit 27 of GMADR is RO, allowing 256MB of GMADR 00b Bits 28:27 of GMADR are RW, allowing 128MB of GMADR

GSA_CR_MSI_CAPID_0_2_0_PCI

B/D/F/Type: 0/2/0/pci
Address Offset: 0x90
Size: 16 bits
Access: RO

When a device supports MSI it can generate an interrupt request to the processor by writing a predefined data item a message to a predefined memory address.

Bit	Type	Default Value	RST Type	Description
15:8	RO	0xd0	default/uncore	**Pointer to Next Capability(POINTNEXT):** This contains a pointer to the next item in the capabilities list which is the Power Management capability.
7:0	RO	0x5	default/uncore	**Capability ID(CAPID):** Value of 05h identifies this linked list item capability structure as being for MSI registers.

GSA_CR_MC_0_2_0_PCI

B/D/F/Type: 0/2/0/pci
Address Offset: 0x92
Access: 16 bits
Size: RW/RO

System software can modify bits in this register, but the device is prohibited from doing so. If the device writes the same message multiple times, only one of those messages is guaranteed to be serviced. If all of them must be serviced, the device must not generate the same message again until the driver services the earlier one.

Bit	Type	Default Value	RST Type	Description
7:7	RO	0x0	default/uncore	**64 Bit Capable(CAP64B):** Hardwired to 0 to indicate that the function does not implement the upper 32 bits of the Message address register and is incapable of generating a 64-bit memory address. This may need to change in future implementations when addressable system memory exceeds the 32b4GB limit.
6:4	RW	0x0	default/uncore/flr	**Multiple Message Enable(MME):** System software programs this field to indicate the actual number of messages allocated to this device. This number will be equal to or less than the number actually requested. The encoding is the same as for the MMC field below.

Bit	Type	Default Value	RST Type	Description
3:1	RO	0x0	default/uncore	**Multiple Message Capable(MMC):** System Software reads this field to determine the number of messages being requested by this device. Value: Number of requests 000: 1 All of the following are reserved in this implementation 001: 2 010: 4 011: 8 100: 16 101: 32 110: Reserved 111: Reserved
0:0	RW	0x0	default/uncore/flr	**MSI Enable(MSIEN):** Controls the ability of this device to generate MSIs.

GSA_CR_MA_0_2_0_PCI

B/D/F/Type: 0/2/0/pci
Address Offset: 0x94
Access: 32 bits
Size: RW/RO

This register contains the Message Address for MSIs sent by the device.

Bit	Type	Default Value	RST Type	Description
31:2	RW	0x0	default/uncore/flr	**Message Address(MESSADD):** Used by system software to assign an MSI address to the device. The device handles an MSI by writing the padded contents of the MD register to this address.
1:0	RO	0x0	default/uncore	**Force Dword Align(FDWORD):** Hardwired to 0 so that addresses assigned by system software are always aligned on a DWORD address boundary.

GSA_CR_MD_0_2_0_PCI

B/D/F/Type: 0/2/0/pci
Address Offset: 0x98
Access: 16 bits
Size: RW

This register contains the Message Data for MSIs sent by the device.

Bit	Type	Default Value	RST Type	Description
15:0	RW	0x0	default/uncore/flr	**Message Data(MESSDATA):** Base message data pattern assigned by system software and used to handle an MSI from the device. When the device must generate an interrupt request, it writes a 32-bit value to the memory address specified in the MA register. The upper 16 bits are always set to 0. The lower 16 bits are supplied by this register.

GSA_CR_AFCIDNP_0_2_0_PCI

B/D/F/Type: 0/2/0/pci
Address Offset: 0xa4
Access: 16 bits
Size: RO

When this capability is linked into the list, the second function of the Internal Graphics Device can be reset independently of the first function.

Bit	Type	Default Value	RST Type	Description
15:8	RO	0x0	default/uncore	**NEXT_PTR:** This contains a pointer to next item in capabilities list. This is the final capability in the list and must be set to 00h.
7:0	RO	0x13	default/uncore	**CAP_ID:** A value of 13h identifies that this PCI Function is capable of Advanced Features.

GSA_CR_AFLC_0_2_0_PCI

B/D/F/Type: 0/2/0/pci
Address Offset: 0xa6
Access: 16 bits
Size: RO

See the July 27th 2006 Engineering Change Notice for Conventional PCI Advanced Capabilities.

Bit	Type	Default Value	RST Type	Description
9:9	RO	0x1	default/uncore	**FLR Capability(FLR_CAP):** Indicates support for Function Level Reset FLR.
8:8	RO	0x1	default/uncore	**TXP Capability(TXP_CAP):** Indicates support for the Transactions Pending bit.
7:0	RO	0x6	default/uncore	**Capability Length(CAP_LEN):** The Advanced Features capability structure requires 6 bytes of configuration space.

UNCORE_CR_AFCTL_0_2_0_PCI

B/D/F/Type: 0/2/0/pci
Address Offset: 0xa8
Size: 8 bits
Access: RW1S
Global: YES

See the July 27th 2006 Engineering Change Notice for Conventional PCI Advanced Capabilities.

Bit	Type	Default Value	RST Type	Description
0:0	RW1S	0x0	default/uncore/flr	**Initiate Function Level Reset(INIT_FLR):** A write of 1b initiates Function Level Reset FLR. FLR requirements are defined in the PCI Express Base Specification. Registers and state information that do not apply to conventional PCI are exempt from the FLR requirements given there. Once written 1, FLR will be initiated. During FLR, a read will return 1s since device 2 reads abort. Once FLR completes, hardware will clear the bit to 0.

GSA_CR_AFSTS_0_2_0_PCI

B/D/F/Type: 0/2/0/pci
Address Offset: 0xa9
Access: 8 bits
Size: RO

See the July 27th 2006 Engineering Change Notice for Conventional PCI Advanced Capabilities.

Bit	Type	Default Value	RST Type	Description
0:0	RO	0x0	default/uncore	**Transactions Pending(TP):** 1: The Function has issued one or more non-posted transactions which have not been completed, including non-posted transactions that a target has terminated with Retry. 0: All non-posted transactions have been completed.

GSA_CR_PMCAPID_0_2_0_PCI

B/D/F/Type: 0/2/0/pci
Address Offset: 0xd0
Access: 16 bits
Size: RO

This register contains the PCI Power Management Capability ID and the next capability pointer.

Bit	Type	Default Value	RST Type	Description
15:8	RO	0xa4	default/uncore	**Next Capability Pointer(NEXT_PTR):** This contains a pointer to the next item in the capabilities list.
7:0	RO	0x1	default/uncore	**Capability Identifier(CAP_ID):** SIG defines this ID is 01h for power management.

GSA_CR_PMCAP_0_2_0_PCI

B/D/F/Type: 0/2/0/pci
Address Offset: 0xd2
Access: 16 bits
Size: RO

This register provides information on the capabilities of the function related to powermanagement.

Bit	Type	Default Value	RST Type	Description
15:11	RO	0x0	default/uncore	**PME Support(PMES):** This field indicates the power states in which the IGD may assert PME#. Hardwired to 0 to indicate that the IGD does not assert the PME# signal.
10:10	RO	0x0	default/uncore	**D2 Support(D2):** The D2 power management state is not supported. This bit is hardwired to 0.
9:9	RO	0x0	default/uncore	**D1 Support(D1):** Hardwired to 0 to indicate that the D1 power management state is not supported.
5:5	RO	0x1	default/uncore	**Device Specific Initialization(DSI):** Hardwired to 1 to indicate that special initialization of the IGD is required before generic class device driver is to use it.
3:3	RO	0x0	default/uncore	**PME Clock(PMECLK):** Hardwired to 0 to indicate IGD does not support PME# generation.
2:0	RO	0x2	default/uncore	**Version(VER):** Hardwired to 010b to indicate that there are 4 bytes of power management registers implemented and that this device complies with revision 1.1 of the PCI Power Management Interface Specification.

UNCORE_CR_PMCS_0_2_0_PCI

B/D/F/Type: 0/2/0/pci
Address Offset: 0xd4
Size: 16 bits
Access: RO_V/RO
Global: YES

Bit	Type	Default Value	RST Type	Description
15:15	RO	0x0	default/uncore	**PMESTS:** This bit is 0 to indicate that IGD does not support PME# generation from D3 cold.
14:13	RO	0x0	default/uncore	**DSCALE:** The IGD does not support data register. This bit always returns 00 when read, write operations have no effect.
12:9	RO	0x0	default/uncore	**DSEL:** The IGD does not support data register. This bit always returns 0h when read, write operations have no effect.
8:8	RO	0x0	default/uncore	**PMEEN:** This bit is 0 to indicate that PME# assertion from D3 cold is disabled.
1:0	RO_V	0x0	default/uncore/flr	**PWRSTAT:** This field indicates the current power state of the IGD and can be used to set the IGD into a new power state. If software attempts to write an unsupported state to this field, write operation must complete normally on the bus, but the data is discarded and no state change occurs. On a transition from D3 to D0 the graphics controller is optionally reset to initial values. Behavior of the graphics controller in supported states is detailed in the power management section of the Bspec. Bits1:0 Power state 00: D0 Default 01: D1 Not Supported 10: D2 Not Supported 11: D3

GSA_CR_SWSMI_0_2_0_PCI

B/D/F/Type: 0/2/0/pci
Address Offset: 0xe0
Access: 16 bits
Size: RW

As long as there is the potential that DVO port legacy drivers exist which expect this register at this address, Dev#2F0address E0h-E1h must be reserved for this register.

Bit	Type	Default Value	RST Type	Description
15:8	RW	0x0	default/uncore	**Software Scratch Bits(SWSB):**
7:1	RW	0x0	default/uncore	**Software Flag(SWF):** Used to indicate caller and SMI function desired, as well as return result.
0:0	RW	0x0	default/uncore	**GMCH Software SMI Event(GSSMIE):** When Set this bit will trigger an SMI. Software must write a "0" to clear this bit. SMI will be triggered only if SWSCISMISCISEL is set to select SMI.

GSA_CR_GSE_0_2_0_PCI

B/D/F/Type: 0/2/0/pci
Address Offset: 0xe4
Access: 32 bits
Size: RW

This register can be accessed by either Byte, Word, or Dword PCI config cycles. A write to this register will cause the Graphics System Event Display Interrupt if enabled.

Bit	Type	Default Value	RST Type	Description
31:24	RW	0x0	default/uncore	**GSE Scratch Trigger 3(GSE3):** When written, this scratch byte triggers an interrupt when the GSE interrupt is unmasked and enabled in the display interrupt registers. If written as part of a 16-bit or 32-bit write, only one interrupt is generated in common.
23:16	RW	0x0	default/uncore	**GSE Scratch Trigger 2(GSE2):** When written, this scratch byte triggers an interrupt when the GSE interrupt is unmasked and enabled in the display interrupt registers. If written as part of a 16-bit or 32-bit write, only one interrupt is generated in common.
15:8	RW	0x0	default/uncore	**GSE Scratch Trigger 1(GSE1):** When written, this scratch byte triggers an interrupt when the GSE interrupt is unmasked and enabled in the display interrupt registers. If written as part of a 16-bit or 32-bit write, only one interrupt is generated in common.
7:0	RW	0x0	default/uncore	**GSE Scratch Trigger 0(GSE0):** When written, this scratch byte triggers an interrupt when the GSE interrupt is unmasked and enabled in the display interrupt registers. If written as part of a 16-bit or 32-bit write, only one interrupt is generated in common.

GSA_CR_SWSCI_0_2_0_PCI

B/D/F/Type: 0/2/0/pci
Address Offset: 0xe8
Access: 16 bits
Size: RW/RW_O

This register serves 2 purposes:

1. Support selection of SMI or SCI event source SMISCISEL - bit15
2. SCI Event trigger GSSCIE - bit 0.

To generate a SW SCI event, software System BIOS/Graphics driver should program bit 15 SMISCISEL to 1. This is typically programmed once assuming SMIs are never triggered. On a "0" to "1" subsequent transition in bit 0 of this register caused by a software write operation, CPU sends a single SCI message down the DMI link to PCH. PCH will set the DMISCI bit in its TCO1STS register and TCOSCISTS bit in its GPE0 register upon receiving this message from DMI. The corresponding SCI event handler in BIOS is to be defined as a Lxx method, indicating level trigger to the operating system.

Once written as 1, software must write a "0" to this bit to clear it, and all other write transitions 1-0, 0-0, 1-1 or if bit 15 is "0" will not cause CPU to send SCI message to DMI link.

To generate a SW SMI event, software should program bit 15 to 0 and trigger SMI via writes to SWSMI register See SWSMI register for programming details.

Bit	Type	Default Value	RST Type	Description
15:15	RW_O	0x0	default/uncore	**SMI or SCI event select(SMISCISEL):** 0 SMI default 1 SCI If selected event source is SMI, SMI trigger and associated scratch bits accesses are performed via SWSMI register at offset E0h. If SCI event source is selected, the rest of the bits in this register provide SCI trigger capability and associated SW scratch pad area.
14:1	RW	0x0	default/uncore	**Software scratch bits(SCISB):** Read/write bits not used by hardware SCISB

Bit	Type	Default Value	RST Type	Description
0:0	RW	0x0	default/uncore	**Software SCI Event(GSSCIE):** If SCI event is selected SMISCISEL 1, on a 0 to 1 transition of GSSCIE bit, a SCI message will be sent via DMI link to ICH to cause the TCOSCISTS bit in its GPE0 register to be set to 1. Software must write a 0 to clear this bit.

GSA_CR_ASLS_0_2_0_PCI

B/D/F/Type: 0/2/0/pci
Address Offset: 0xfc
Access: 32 bits
Size: RW

This software scratch register only needs to be readwrite accessible. The exact bit register usage must be worked out in common between System BIOS and driver software, but storage for switchingindicating up to 6 devices is possible with this amount.

For each device, the ASL control method with require two bits for DOD BIOS detectable yes or no, VGA/NonVGA, one bit for DGS enable/disable requested, and two bits for DCS enabled now/disabled now, connected or not.

Bit	Type	Default Value	RST Type	Description
31:0	RW	0x0	default/uncore	**Device Switching Storage(DSS):** Software controlled usage to support device switching.