AMD 780G Family
Register Programming Requirements

For the RS780, RS780C, RS780D, RS780M,
RS780E, RS780MC, and RX781

Technical Reference Manual
Rev. 1.01

P/N: 43291_rs780_rpr_pub_1.01

Table of Contents

Chapter 8: HTIU Settings

Chapter 9: CLMC Programming

Appendix A: Revision History

Chapter 1
Introduction

1.1 About This Manual

This document is intended for BIOS engineers designing BIOSes for systems based on AMD's 780G family of northbridges. It describes the register programming requirements needed to ensure the proper functioning of the 780G ASIC. Use this document in conjunction with the related *AMD 780G Family Register Reference Guide* and *AMD 780G Family BIOS Developer's Guide*.

Unless indicated otherwise, the programming information in this document applies to the following 780G variants (note that *Chapter 9* only applies to 780G mobile variants):

- RS780 (AMD 780G)
- RS780C (AMD 780V)
- RS780D (AMD 790GX)
- RS780E (AMD 780E)
- RS780M (AMD M780G)
- RS780MC (AMD M780V)
- RX781 (AMD M770) (*Chapter 6* does not apply to the RX781 variant)

Some of the settings indicated in this document are workarounds for items that are expected to be solved in subsequent ASIC revisions. This document will therefore be updated as frequently as required.

Changes and additions to the previous release of this document are highlighted in red. Refer to *Appendix A: Revision History* at the end of this document for a detailed revision history.

This page intentionally left blank.

Chapter 2
I/O Control (IOC)

2.1 RS780 Device Mapping

The RS780 has the following devices:

- Bus0Dev0Fun0: Host bridge
- Bus0Dev0Fun1: Clock control
- Bus0Dev1: Internal graphics P2P bridge
- Bus0Dev2: PCIE P2P bridge (external graphics)
- Bus0Dev3: PCIE P2P bridge (external graphics)
- Bus0Dev4: PCIE P2P bridge
- Bus0Dev5: PCIE P2P bridge
- Bus0Dev6: PCIE P2P bridge
- Bus0Dev7: PCIE P2P bridge
- Bus0Dev9: PCIE P2P bridge
- Bus0Dev10: PCIE P2P bridge
- Bus0Dev8: NB/SB Link P2P bridge (hidden by default)

Note: Each device has a P2P bridge header, except Dev0, which has a PCI device header.

2.2 RS780 Device IDs

Table 2-1 RS780 Device IDs

Register	RS780	RS780C	RS780M	RS780MC	Description
NB_DEVICE_ID<nbcfg:0X02>DEVICE_ID [15:0]	0x9600				Northbridge configuration space ID
APC_DEVICE_ID<APCCFG:0x02>DEVICE_ID [15:0]	0x9602				Internal PCI-PCI bridge ID
	0x9603				External GFX - port 0
	0x960B				External GFX - port 1
	0x9610	0x9611	0x9612	0x9613	Internal graphics
	0x9604				PCI-PCI bridge - Port 0
	0x9605				PCI-PCI bridge - Port 1
	0x9606				PCI-PCI bridge - Port 2
	0x9607				PCI-PCI bridge - Port 3
	0x9608				PCI-PCI bridge - Port 4
	0x9609				PCI-PCI bridge - Port 5
	0x960A				PCI-PCI bridge (SB)
	0x960F				HD Audio controller
	0x791A				HDMI Audio codec

2.3 Configuration Access to RS780 Device Registers

Configuration access to the RS780 can be accomplished through one of the following two methods described in sections *2.3.1* and *2.3.2* below.

2.3.1 Using CF8/CFC I/O Pair

This method works for all registers of Dev0 and Dev1, and all PCI registers of Dev2 to Dev10. This method DOES NOT work for PCIE extended registers of Dev2 to Dev10. The address mapping follows the standard PCI specification:

- Addr[11:8] = FunNum
- Addr[15:12] = DevNum
- Addr[23:16] = BusNum
- Addr[7:2] = RegNum

Note: For conventional CF8/CFC IO pair configuration access, the first IO write to CF8 (which is a register index access), has to set Data[31] to indicate that this is a configuration access. Otherwise, it will be treated as a regular IO cycle.

2.3.2 Using BAR3 Memory Mapped Register Access

This method works for all PCI registers of Dev0, all PCI registers, and PCIE extended registers of Dev2 to Dev8. The address mapping follows the PCIE specification:

- Addr[14:12] = FunNum
- Addr[19:15] = DevNum
- Addr[11:2] = RegNum (Addr[11:8] is an extended register field)
- Addr[20 + n-1:20] = BusNum *
- Addr[33:20 + n] = Reserved for BAR3 match *

* **Note:** 'n' indicates how many bits are allocated for the bus number. This value is decided by nbcfg0x84[18:16]. These relations are listed in *Table 2-2* below:

Table 2-2 nbcfg0x84[18:16] Relations

nbcfg0x84[18:16]	n
3'b001	1
3'b010	2
3'b011	3
3'b100	4
3'b101	5
3'b110	6
3'b111	7
3'b000	8

The programming procedure to enable BAR3 is as follows:

- Step 1: Enable BAR3 register access (set nbcfg0x7C[30])
- Step 2: Program BAR3 bus range (nbcfg0x84[18:16]).
- Step 3: Program the BAR3 register (nbcfg0x1C[31:21] and nbcfg0x20[1:0])
- Step 4: Enable BAR3 decoding (set htiunbind 0x32[28]).

Note: nbcfg0x20 is the BAR3 memory upper address register (above 4G). The RS780 could support memory up to 16G, so this register must be set correctly.

2.4 General RS780 IOC Programming After Boot-Up

After system boot-up, all registers should keep the default values.

The BIOS starts the bus enumeration, and detects the following: Bus0Dev0Fun0, Dev0Fun1, Dev1Fun0, Dev1Fun1, Dev2Fun0, Dev3Fun0, Dev4Fun0, Dev5Fun0, Dev6Fun0, Dev7Fun0. Then, for all of these PCI device headers or P2P device headers, the BIOS enables IOSpace (0x04[0]) and MemSpaceEn (0x04[1]). It also defines the primary bus number, the secondary bus number, and the subordinate bus number.

The following registers in *Table 2-3* need to be programmed after boot-up. Note: After boot-up to Windows occurs, the IOC register default values follow the values in this table.

Table 2-3 Expected Register Values

Register	Offset	Expected Value
NB_BAR1_RCRB	nbcfg0x14	32'hxxxx_xxxx
NB_BAR2_PM2	nbcfg0x18	32'hxxxx_xxxx
NB_BAR3_PCIEXP_MMCFG	nbcfg0x1C	32'hxxxx_xxxx
NB_BAR3_UPPER_PCIEXP_MMCFG	nbcfg0x20	32'h0000_000x
NB_PCI_CTRL	nbcfg0x4C	32'h0000_0000_0x00_01x1_0010_0000_1100_00xx
NB_IO_CFG_CNTL	nbcfg0x7C	32'h4000_0000
NB_PCI_ARB	nbcfg0x84	32'b0000_0000_0000_0xxx_0000_00xx_1001_0101
IOC_DMA_ARBITER	nbmisc0x09	32'hxxxx_xxxx
IOC_PCIE_CSR_COUNT	nbmisc0x0A	32'hxxxx_xxxx
IOC_PCIE_CNTL	nbmisc0x0B	32'h0000_0180
IOC_P2P_CNTL	nbmisc0x0C	32'b0000_0000_0000_0000_0xx1_0111_xxxx_xx00
CMP_MSK_EOB	nbmisc0x0D	32'hxxxx_xxxx
IOC_DMA_ARBITER	nbmisc0x0E	32'hxxxx_xxxx
IOC_DMA_ARBITER	nbmisc0x0F	32'hxxxx_xxxx
IOC_DMA_ARBITER	nbmisc0x11	32'hxxxx_xxxx
NB_TOM_PCI	nbmisc0x16	32'hxxxx_000x
NB_MMIOBASE	nbmisc0x17	32'h0000_0000
NB_MMIOLIMIT	nbmisc0x18	32'h0000_0000
NB_BROADCAST_BASE_LO	nbmisc0x3A	32'hxxx0_0000
NB_BROADCAST_BASE_HI	nbmisc0x3B	32'h0000_000x
NB_BROADCAST_CNTL	nbmisc0x3C	32'hxxxx_xxxx
IOC_PCIE_D2_CNTL	nbmisc0x51	32'h0010_0100
IOC_PCIE_D3_CNTL	nbmisc0x53	32'h0010_0100
IOC_PCIE_D4_CNTL	nbmisc0x55	32'h0010_0100
IOC_PCIE_D5_CNTL	nbmisc0x57	32'h0010_0100
IOC_PCIE_D6_CNTL	nbmisc0x59	32'h0010_0100
IOC_PCIE_D7_CNTL	nbmisc0x5B	32'h0010_0100
IOC_PCIE_D9_CNTL	nbmisc0x5D	32'h0010_0100
IOC_PCIE_D10_CNTL	nbmisc0x5F	32'h0010_0100
NB_IOC_DEBUG	nbmisc0x1	32'h0000_0048

2.5 Miscellaneous IOC Features Programming

2.5.1 Power Management Register Access Setup

BAR2 is used to access the Power Management registers. The programming procedure to setup BAR2 is as follows:

- Step 1: Enable BAR2 register access (set nbcfg0x4C[17]).
- Step 2: Program the BAR2 register (assign values to nbcfg0x18[31:5]). A 32 bytes IO space is reserved for BAR2(ACPI PM) registers.
- Step 3: Enable BAR2 decoding (set nbcfg0x84[7]).

Note: The above programming procedure is necessary before enabling ACPI. BAR2 is a memory mapped IO base register that could be used to reserve some space for the ACPI registers. After BAR2 is setup, IO access which address matches BAR[31:5] should be treated as ACPI register access, and Addr[4:0] is used as the register offset. The current offset 0x00 and 0x04 are used, as PM2_CNTL and PM1_Status, respectively.

2.5.2 S3 PME_Turn_Off/PME_To_Ack Sequence

No programming is required in the RS780. However, a backup sequence is required in case there is a mis-communication between the northbridge and the southbridge.

2.5.3 Disabling Internal Graphics

Internal graphics disabling is controlled by an efuse bit, but may also be disabled by writing 1 to register nbcfg0x7C[0] (NBCFG.NB_IOC_CFG_CNTL[0])

2.5.4 GFX MSI Enable

The SBIOS must enable internal graphics MSI capability in GCCFG by setting the following:

- NBCFG.NB_CNTL.STRAP_MSI_ENABLE=’1’

The OS will determine if MSI’s are supported by the system, and if so, the OS will set the following:

- GCCFG.MSI_MSG_CNTL.MSI_EN=’1’

Note: At the time of this writing, to enable MSI in Vista, set the registry key as follows:

- MSIsupported=1

2.5.5 Disabling Bus0 Device 3 PCI Bridge (Secondary External PCIE Graphics)

Set nbmiscind0x0C[3] to disable Bus0 Device3 register access and decoding. Note: An efuse called CrossFireDisable is also used that could disable Device 3. Either bit as 1 would disable device 3.

2.5.6 Disabling Bus0 Device 3 PCI Bridges (Dev2, Dev4 to Dev7, Dev9-Dev10)

Set any bit according to the information in *Table 2-4*:

Table 2-4 Disabling Bus0 Device3 PCI Bridges Settings

Devices	Bit Settings
Bus0 Device2	nbmiscind 0x0C[2]
Bus0 Device4	nbmiscind 0x0C[4]
Bus0 Device5	nbmiscind 0x0C[5]
Bus0 Device6	nbmiscind 0x0C[6]
Bus0 Device7	nbmiscind 0x0C[7]
Bus0 Device9	nbmiscind 0x0C[16]
Bus0 Device10	nbmiscind 0x0C[17]

2.6 Broadcast CPU Requests to Dual External Graphics PCIE Devices

Figure 2-1 describes the algorithm for address translation and broadcast:

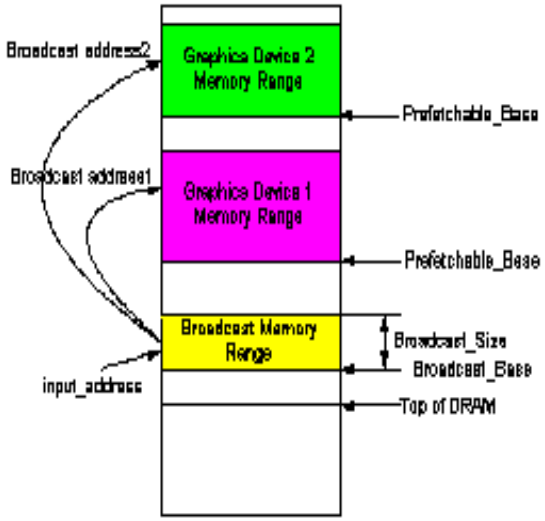

Figure 2-1 Address Translation And Broadcast

The Address Translation and Broadcast Algorithm is as follows:

- Broadcast_Address = input_address[63:0] – {BROADCAST_BASE[63:20], 20'b0} + {Bridge_Prefetchable_BASE[63:20], 20'h} + {32'h0, BROADCAST_OFFSET[31:12], 12'h0}.
- Input_address refers the request address IOC received from CPU.
- BROADCAST_BASE refers to the broadcast memory range start address.
- Bridge_Prefetchable_BASE refers to external graphics device memory range start address.
- BROADCAST_OFFSET is the offset between translated broadcast base address and bridge Prefetchable_BASE address.

Two broadcast addresses are obtained by applying two Bridge_Prefetchable_BASE addresses from the two PCI configuration space. Therefore, a single CPU memory write request could be translated and redirected to two external graphics devices by IOC. Note that this address translation and broadcast algorithm is only applicable to CPU memory write requests. For CPU memory read requests, the address translation to the primary graphics device is performed using the above equation, and the request is only forwarded to the primary graphics devices since only one response is expected by the CPU.

- The 32-bit registers are defined as follows:
  - [63:20] GPU_FB_BROADCAST_BASE // 1M aligned broadcast address
  - [31:12] GPU_FB_BROADCAST_OFFSET // 4K aligned broadcast offset address
  - [11:11] GPU_FB_BROADCAST_EN // Enable broadcast feature
  - [10:10] GPU_FB_BROADCAST_PRIMARY // Primary GPU
    - 0 = Lower device/port#
    - 1 = Higher device/port#
  - [07:00] GPU_FB_BROADCAST_SIZE // Size (8MB)
- If enabled, and the address is in range (address -> address + size), then the broadcast memory writes to both ports.
- Broadcast only works for applicable memory writes, and applicable reads will be sent to the primary device only.
- Broadcast is enabled when the enable bit is set, when both device 2 and device 3 bridges are enabled, and when the memory space is enabled.
- The broadcast memory range should not conflict with any existing P2P memory range, or any BAR memory range.
- Broadcast address = input_address[63:0] - {GPU_FB_BROADCASE_BASE[63:20], 20'h0} + {bridge_prefetchable_bar[63:20], 20'h0}
- + { 32'h0, GPU_FB_BROADCAST_OFFSET[31:12], 12'h0}
- Address [63:34] does not need to be checked since they are not used (the RS780 supports up to 16G memory space)

2.7 Enabling/Disabling Peer-To-Peer Traffic Access

The P2P master could be any device from the southbridge, devices connected behind P2P bridge 2, 3, 4, 5, 6, 7,9 and 10. The P2P targets could be devices connected behind P2P bridge 1, 2, 3, 4, 5, 6, 7,9, 10. The southbridge cannot be a target for trusted-PC purposes. The P2P traffic could be only memory writes. After bootup, by default all P2P traffic listed above should be enabled. In order to disable a P2P target at a specific device, the following register bits in *Table 2-5* need to be set as follows:

Table 2-5 Enabling/Disabling Peer-To-Peer Traffic Settings

Devices	Bit Settings
Bus0 Device1	nbmisc0x4C[2]
Bus0 Device2	nbmisc0x51[3]
Bus0 Device3	nbmisc0x53[3]
Bus0 Device4	nbmisc0x55[3]
Bus0 Device5	nbmisc0x57[3]
Bus0 Device6	nbmisc0x59[3]
Bus0 Device7	nbmisc0x5B[3]
Bus0 Device9	nbmisc0x5D[3]
Bus0 Device10	nbmisc0x5F[3]

2.8 Enabling/Disabling MVPU

MVPU is a feature that enables P2P traffic between external graphics devices (the devices behind P2P bridge 2 and 3) and the internal graphics device (the device behind P2P bridge 1). The corresponding P2P traffic access enable bits are described in section *Table 2-5* above.

2.9 IOC Dynamic Clock Setup

The following clocks are in IOC:

- LCLK (free running)
- LCLK_MST (master branch)
- LCLK_SLV (slave branch - Note: This dynamic branch should not be used)

Note: Only LCLK_MST (master branch) and LCLK_SLV (slave branch) can be dynamically turned on and off.

The two bits that control IOC dynamic clocks are as follows:

- clkcfg0x8C[13] CLKGATE_DIS_IOC_LCLK_MST
- clkcfg0x8C[14] CLKGATE_DIS_IOC_LCLK_SLV (Note: Ensure that this bit is programmed to 1 in order to avoid system instability)

Note: Clkconfig:0x94[27] CLKGATE_IOC_SLV_GFX - BIOS should program to 1 to disable clock gating on this branch.

For both of these bits:

- 1=Dynamic clock is disabled
- 0=Dynamic clock is enabled

2.10 Interrupt Mapping

Table 2-6 Interrupt Mapping Settings

Devices	Bit Settings
1 (internal graphics)	INTA -> INTC, INTB->INTD
2	INTA -> INTC
3	INTA -> INTD
4	INTA -> INTA
5	INTA -> INTB
6	INTA -> INTC
7	INTA -> INTD
9	INTA -> INTB
10	INTA -> INTC

2.11 GSM Enable

Set nbmisind0x0C[13]=1 to enable GSM in the RS780.

This page intentionally left blank.

Chapter 3
Clock Settings

3.1 SBIOS Memory Clock Initialization

3.1.1 UMA Mode

RS780 boots up in synchronous UMA clock mode. The memory clock and HT clock are driven by the same HT PLL. In UMA sync mode the memory PLL is not used and should be powered down.

- Program <NBMCIND:0x6> Bit[31] MC_MPLL_CONTROL.MPLL_POWERDOWN = '1' to power down memory PLL in sync mode.

3.1.2 Side-Port Async Mode

- Step 1: Program IO 1XCLK skew delay by setting
  - <NBMCIND:0x6> Bit[10:8] MC_MPLL_CONTROL.MPLL_SKEW1=0x1
- Step 2: Program current control for SCL for PLL to 0%
  - < NBMCIND:0x8 > Bit[20:18] MC_MPLL_CONTROL3.MPLL_SCLBIAS = 0x1
- Step 3: Select memory PLL reference clock. Default is 100MHz HT reference and it should be used in normal operation.
  - <NBMCIND : 0x8> Bit[10] MC_MPLL_CONTROL3.MPLL_REFCLK_SEL ('0'=100 MHz HT reference clock; '1'=PCIE reference clock)
- Step 4: Program memory PLL settings for different operating frequencies.
  - Feedback divider : < NBMCIND:0x7 > Bit[8:0] MC_MPLL_CONTROL2.MPLL_FBDIV
  - Reference divider : < NBMCIND:0x7 > Bit[13:9] MC_MPLL_CONTROL2.MPLL_REFDIV
  - Post divider (postdiv) : < NBMCIND:0x7 > Bit[15:14] MC_MPLL_CONTROL2.MPLL_POSTDIV
  - Charge pump : < NBMCIND:0x7 > Bit[19:16] MC_MPLL_CONTROL2.MPLL_CP
  - VCO mode : < NBMCIND:0x7 > Bit[21:20] MC_MPLL_CONTROL2.MPLL_VCO_MODE
  - Loop filter mode: < NBMCIND:0x7 > Bit[31:28] MC_MPLL_CONTROL2.MPLL_LF_MODE
  - Consider the following equation to calculate MCLK:
    - P1 = 100MHz / (MPLL_REFDIV + 1)
    - P2 = (MPLL_FBDIV[2:0] + 1) * (MPLL_FBDIV[8:3] + 1) * 2
    - P3 = 2 * (MPLL_POSTDIV+1)
    - MCLK = P1 * P2 / P3
  - The divider settings should use the frequency plan in *Table 3-1* from memory PLL specification.
    - 200MHz settings:
      - <NBMCIND:0x7> MC_MPLL_CONTROL2=0x00004018;
    - 266MHz settings:
      - <NBMCIND:0x7> MC_MPLL_CONTRO2=0x00114478;
    - 333MHz settings:
      - <NBMCIND:0x7> MC_MPLL_CONTROL2=0x00224498;
    - 400MHz settings:
      - <NBMCIND:0x7> MC_MPLL_CONTROL2=0x00000018;
    - 533MHz settings:
      - <NBMCIND:0x7> MC_MPLL_CONTROL2=0x00110478;

        - 667MHz settings:
            - <NBMCIND:0x7> MC_MPLL_CONTROL2=0x00220498
    - Note: For other frequencies, refer to TABLE X below.
- Step 5: Program PM mode PLL setting the same as nominal mode setting
    - 200MHz:
        - <NBMCIND:0xB> MC_MPLL_DIV_CONTROL=0x00004018
        - <NBMCIND:0x9> Bit[11:8] MC_MPLL_FREQ_CONTROL.PM_MPLL_CP=0x0
        - <NBMCIND:0x9> Bit[13:12] MC_MPLL_FREQ_CONTROL.PM_MPLL_VCO_MODE=0x0
        - <NBMCIND:0x9> Bit[19:16] MC_MPLL_FREQ_CONTROL.PM_MPLL_LF_MODE=0x6
    - 266MHz:
        - <NBMCIND:0xB> MC_MPLL_DIV_CONTROL=0x00004478
        - <NBMCIND:0x9> Bit[11:8] MC_MPLL_FREQ_CONTROL.PM_MPLL_CP=0x1
        - <NBMCIND:0x9> Bit[13:12] MC_MPLL_FREQ_CONTROL.PM_MPLL_VCO_MODE=0x1
        - <NBMCIND:0x9> Bit[19:16] MC_MPLL_FREQ_CONTROL.PM_MPLL_LF_MODE=0x4
    - 333MHz:
        - <NBMCIND:0xB> MC_MPLL_DIV_CONTROL=0x00004498
        - <NBMCIND:0x9> Bit[11:8] MC_MPLL_FREQ_CONTROL.PM_MPLL_CP=0x2
        - <NBMCIND:0x9> Bit[13:12] MC_MPLL_FREQ_CONTROL.PM_MPLL_VCO_MODE=0x2
        - <NBMCIND:0x9> Bit[19:16] MC_MPLL_FREQ_CONTROL.PM_MPLL_LF_MODE=0x4
    - 400MHz:
        - <NBMCIND:0xB> MC_MPLL_DIV_CONTROL=0x00000018
        - <NBMCIND:0x9> Bit[11:8] MC_MPLL_FREQ_CONTROL.PM_MPLL_CP=0x0
        - <NBMCIND:0x9> Bit[13:12] MC_MPLL_FREQ_CONTROL.PM_MPLL_VCO_MODE=0x0
        - <NBMCIND:0x9> Bit[19:16] MC_MPLL_FREQ_CONTROL.PM_MPLL_LF_MODE=0x6
    - 533MHz:
        - <NBMCIND:0xB> MC_MPLL_DIV_CONTROL=0x00000478
        - <NBMCIND:0x9> Bit[11:8] MC_MPLL_FREQ_CONTROL.PM_MPLL_CP=0x1
        - <NBMCIND:0x9> Bit[13:12] MC_MPLL_FREQ_CONTROL.PM_MPLL_VCO_MODE=0x1
        - <NBMCIND:0x9> Bit[19:16] MC_MPLL_FREQ_CONTROL.PM_MPLL_LF_MODE=0x4
    - 667MHz:
        - <NBMCIND:0xB> MC_MPLL_DIV_CONTROL=0x00000498
        - <NBMCIND:0x9> Bit[11:8] MC_MPLL_FREQ_CONTROL.PM_MPLL_CP=0x2
        - <NBMCIND:0x9> Bit[13:12] MC_MPLL_FREQ_CONTROL.PM_MPLL_VCO_MODE=0x2
        - <NBMCIND:0x9> Bit[19:16] MC_MPLL_FREQ_CONTROL.PM_MPLL_LF_MODE=0x4
    - Note: For other PM frequencies, refer to *Table 3-1* below.
- Step 6: Program memory PLL and DLL lock time
    - Min. 10us calibration setup time: Calibration setup time = 10ns x <NBMCIND:0xA> Bit[11:8] MC_MPLL_SEQ_CONTRL.MPLL_CAL_S_TIME x 512
    - Min. 50ns calibration hold time: Calibration hold time = 10ns x <NBMCIND:0xA> Bit[15:12] MC_MPLL_SEQ_CONTROL.MPLL_CAL_H_TIME x 4
    - Min. 50us PLL lock time: MPLL lock time = 10ns x <NBMCIND:0xA> Bit[23:16] MC_MPLL_SEQ_CONTROL.MPLL_LOCK_TIME x 256
    - Min. 50us DLL lock time: MDLL lock time = 10ns x <NBMCIND:0xA> Bit[31:24] MC_MPLL_SEQ_CONTROL.MDLL_LOCK_TIME x 256

- Step 7: Recalibrate the memory PLL
  - Set < NBMCIND:0x6 > Bit[0] MC_MPLL_CONTROL.MPLL_CAL_TRIGGER = ’1’;
- Step 8: Wait for 200us;
- Step 9: Poll PLL lock signal < NBMCIND:0x6 > Bit[1] MC_MPLL_CONTROL.MPLL_LOCKED = ’1’;
- Step 10: Set < NBMCIND:0x6 > Bit[0] MC_MPLL_CONTROL.MPLL_CAL_TRIGGER = ’0’
- Step 11: Program <NBMCIND:0x2> Bit[20] MC_GENERAL_PURPOSE_2.MCLK_SRC_USE_MPLL = ‘0’ to select asynchronous clock mode.
- Step 12: Start the memory initialization sequence.

3.2 Memory Clock Changes For POWERPLAY

Note: PowerPlay MCLK switching only applies to asynchronous clock mode for memory side port

- Step 1: Program MPLL divider in PM mode by setting
  - < NBMCIND:0xB > Bit[8:0] MC_MPLL_DIV_CONTROL.PM_MPLL_FBDIV
  - < NBMCIND:0xB > Bit[13:9] MC_MPLL_DIV_CONTROL.PM_MPLL_REFDIV
  - < NBMCIND:0xB > Bit[15:14] MC_MPLL_DIV_CONTROL.PM_MPLL_POSTDIV
  - Consider the following Equation to calculate MCLK in PM mode:
    - P1 = 100MHz / (PM_MPLL_REFDIV + 1)
    - P2 = (PM_MPLL_FBDIV[2:0] + 1) * (PM_MPLL_FBDIV[8:3] + 1) * 2
    - P3 = 2 * (PM_MPLL_POSTDIV+1)
    - MCLK = P1 * P2 / P3
  - Use the settings in *Table 3-1* to get the divider settings for the required frequencies. For example:
    - 133MHz in PM mode
      - MC_MPLL_DIV_CONTROL.PM_MPLL_FBDIV = 0x78
      - MC_MPLL_DIV_CONTROL.PM_MPLL_REFDIV = 0x2
      - MC_MPLL_DIV_CONTROL.PM_MPLL_POSTDIVS = 0x3
- Step 2: If MCLK < HT_CLK in PM mode, set < NBMCIND:0x9 > Bit[29] MC_MPLL_FREQ_CONTROL.PM_MPLL_SLOWMCLK = 0x1 else 0x0
- Step 3: Program < NBMCIND:0x9 > Bit[0] MC_MPLL_FREQ_CONTROL.MPLL_PM_EN = 0x1
- Step 4: Program MPLL parameters in PM mode by setting the following:
  - <NBMCIND:0x9> Bit[11:8] MC_MPLL_FREQ_CONTROL.PM_MPLL_CP
  - <NBMCIND:0x9> Bit[13:12] MC_MPLL_FREQ_CONTROL.PM_MPLL_VCO_MODE
  - <NBMCIND:0x9> Bit[19:16] MC_MPLL_FREQ_CONTROL.PM_MPLL_LF_MODE
  - Use the settings in *Table 3-1* to program the required frequencies. For example:
    - MPLL parameter for 133MHz in PM mode
      - MC_MPLL_FREQ_CONTROL.PM_MPLL_CP = 0x1
      - MC_MPLL_FREQ_CONTROL.PM_VCO_MODE = 0x1
      - MC_MPLL_FREQ_CONTROL.PM_LF_MODE = 0x4
- Step 5: Program the memory controller settings in different PM mode speed
  - Refer to the *AMD RS780 BIOS Developer’s Guide* for more information.
- Step 6: Switch to PM mode MCLK by setting < NBMCIND:0x9 > Bit[1] MC_MPLL_FREQ_CONTROL.MPLL_FREQ_SEL = 0x1
- Step 7: Poll for < NBMCIND:0x9 > Bit[6] MC_MPLL_FREQ_CONTROL.PM_SWITCHMCLK_BUSY = 0x0

3.3 Switching Back From PM Mode to Nominal Mode

- Step 1: Set < NBMCIND:0x9 > Bit[1] MC_MPLL_FREQ_CONTROL.MPLL_FREQ_SEL = 0
- Step 2: Poll for < NBMCIND:0x9 > Bit[6] MC_MPLL_FREQ_CONTROL.PM_SWITCHMCLK_BUSY = 0x0.

Table 3-1 Memory PLL Settings For Supposed Frequencies

Target (MHz)	Achieved (MHz)	Error (MHz)	Ref_div	3-bit CMOS FB div	6-bit CMOS FB div	POST	IICP [3:0]	ILF_MODE[3 :0]	IVCO_MODE[1:0]	VCO freq (MHz)
100	100	0	1	1	4	4	0000	0110	00	800
133.3333	133.3333	0	3	1	16	4	0001	0100	01	1066.6667
166.6666	166.6666	0	1	1	5	3	0000	0110	01	1000
200	200	0	1	1	4	2	0000	0110	00	800
233.3333	233.3333	0	1	1	7	3	0000	0100	10	1400
266.6666	266.6666	0	3	1	16	2	0001	0100	01	1066.6667
300	300	0	1	1	6	2	0000	0100	01	1200
333.3333	333.3333	0	3	1	20	2	0010	0100	10	1333.3333
366.6666	366.6666	0	3	1	11	1	0001	0100	00	733.33333
400	400	0	1	1	4	1	0000	0110	00	800
433.3333	433.3333	0	3	1	13	1	0001	0100	00	866.66667
466.6666	466.6666	0	3	1	14	1	0001	0100	00	933.33333
500	500	0	1	1	5	1	0000	0110	01	1000
533.3333	533.3333	0	3	1	16	1	0001	0100	01	1066.6667
566.6666	566.6666	0	3	1	17	1	0001	0100	01	1133.3333
600	600	0	1	1	6	1	0000	0100	01	1200
633.3333	633.3333	0	3	1	19	1	0010	0100	10	1266.6667
666.6666	666.6666	0	3	1	20	1	0010	0100	10	1333.3333
100	100	0	1	1	4	4	0000	0110	00	800
112	112.5	0.5	2	1	9	4	0000	0100	00	900
124	123.8095	0.2	7	1	26	3	0010	1000	00	742.85714
136	136.1111	0.1	9	1	49	4	0011	1100	01	1088.8889
148	148.1481	0.15	9	1	40	3	0011	1000	00	888.88889
160	160	0	5	1	24	3	0010	0100	01	960
172	171.875	0.1	8	1	55	4	0011	1100	10	1375
184	184.375	0.4	8	1	59	4	0011	1100	10	1475
196	195.8333	0.15	8	1	47	3	0011	1100	01	1175
208	208.3333	0.35	4	1	25	3	0010	0100	01	1250
220	220	0	5	1	33	3	0010	1000	10	1320
232	231.25	0.75	8	1	37	2	0011	1000	00	925
244	243.75	0.25	8	1	39	2	0011	1000	01	975
256	256.25	0.25	8	1	41	2	0011	1100	01	1025
268	268.75	0.75	8	1	43	2	0011	1100	01	1075
280	280	0	5	1	28	2	0010	1000	01	1120
292	291.6666	0.35	6	1	35	2	0010	1000	01	1166.6667
304	305.5555	1.55	9	1	55	2	0011	1100	01	1222.2222
316	316.6666	0.65	3	1	19	2	0010	0100	10	1266.6667
328	327.7777	0.2	9	1	59	2	0011	1100	10	1311.1111
340	340	0	5	1	34	2	0010	1000	10	1360
352	350	2	1	1	7	2	0000	0100	10	1400
364	364.2857	0.3	7	1	51	2	0011	1100	10	1457.1429
376	375	1	4	1	15	1	0001	0100	00	750
388	387.5	0.5	8	1	31	1	0010	1000	00	775
400	400	0	1	1	4	1	0000	0110	00	800
412	412.5	0.5	8	1	33	1	0010	1000	00	825

Target (MHz)	Achieved (MHz)	Error (MHz)	Ref_div	3-bit CMOS FB div	6-bit CMOS FB div	POST	IICP [3:0]	ILF_MODE[3 :0]	IVCO_MODE[1:0]	VCO freq (MHz)
424	425	1	4	1	17	1	0001	0100	00	850
436	437.5	1.5	8	1	35	1	0010	1000	00	875
448	450	2	2	1	9	1	0000	0100	00	900
460	460	0	5	1	23	1	0010	0100	00	920
472	471.4285	0.55	7	1	33	1	0010	1000	01	942.85714
484	483.3333	0.65	6	1	29	1	0010	1000	01	966.66667
496	500	4	1	1	5	1	0000	0110	01	1000
508	511.1111	3.1	9	1	46	1	0011	1100	01	1022.2222
520	520	0	5	1	26	1	0010	1000	01	1040
532	533.3333	1.35	3	1	16	1	0001	0100	01	1066.6667
544	544.4444	0.45	9	1	49	1	0011	1100	01	1088.8889
556	555.5555	0.45	9	1	50	1	0011	1100	01	1111.1111
568	566.6666	1.35	3	1	17	1	0001	0100	01	1133.3333
580	580	0	5	1	29	1	0010	1000	01	1160
592	588.8888	3.1	9	1	53	1	0011	1100	01	1177.7778
604	600	4	1	1	6	1	0000	0100	01	1200
616	616.6666	0.65	6	1	37	1	0011	1000	01	1233.3333
628	628.5714	0.55	7	1	44	1	0011	1100	10	1257.1429
640	640	0	5	1	32	1	0010	1000	10	1280
652	650	2	2	1	13	1	0001	0100	10	1300
664	662.5	1.5	8	1	53	1	0011	1100	10	1325
676	675	1	4	1	27	1	0010	1000	10	1350

3.4 Power Saving Settings

3.4.1 Enabling Dynamic Clocks

Table 3-2 Dynamic Clocks Settings

ASIC Rev	Settings	Function/Comment
All Revs	CFG_CT_CLKGATE_HTIU <clkcfg:0xf8> = 0xcf30 CLKCFG.CLKGATE_DISABLE[31], [28], [26], [24] = 0x0 CLKCFG.CLKGATE_DISABLE2[26:24] = 0x0 CLKCFG.CLKGATE_DISABLE2[14] = 0x1 CLKCFG.CLK_TOP_SPARE_C[31:28]=0xe [27:26]=0x3 NBMCIND.MC_MCLK_CONTROL = 0xf000000	Enables North Bridge dynamic clocks to htiu and mc

3.4.2 Powering Down Efuse and Strap Block Clocks After Boot-Up

Table 3-3 Powering Down Efuse and Strap Block Clocks Settings

ASIC Rev	Settings	Function/Comment
All Revs	Intgfx mode: <CLKCFG: 0xCC> Bit[24:23] = 0x3; <GpuF0MMReg:0x3080> Bit[8] CG.CG_INTGFX_MISC.CG_CT_NB_EFUSE_CLK_DISABLE= 0x1 NB only mode: <CLKCFG: 0xCC> Bit[24:23] = 0x3	Powerdown efuse and strap block clocks in integrated graphics mode Powerdown efuse and strap block clocks in northbridge only mode

3.4.3 Powering Down Graphics Core and Memory Clocks in Northbridge-Only Mode

Table 3-4 Powering Down Graphics core and Memory Clocks in NB-Only Mode Settings

ASIC Rev	Settings	Function/Comment
All Revs	<CLKCFG:0x8C> Bit[21] = 0x1 <CLKCFG:00xE4> Bit[0] = 0x1	Powers down reference clock to graphics core PLL in northbridge only mode Powers down clock to memory controller in northbridge only mode

3.4.4 Powering Down IOC GFX Clock in No External Graphics Mode

Table 3-5 Powering Down IOC GFX Clock In No External Graphics Mode Settings

ASIC Rev	Settings	Function/Comment
All Revs	<CLKCFG:0xE8> Bit[17] = 0x1	Powers down clock to IOC GFX block in no external graphics mode

3.4.5 PWM Controller

There are five PWM controllers mapped to five GPIO pins that can be used for voltage adjustment purpose after boot-up.

Table 3-6 PWM Controller/GPIO Pins Mapping

ASIC Rev	Register setting	Function/Comment
All Revs	1.CLK_TOP_PWM1_CTRL<CLKCFG:0xB0> 2. CLK_TOP_PWM2_CTRL<CLKCFG:0xB4> 3. CLK_TOP_PWM3_CTRL <CLKCFG:0xCC> 4. CLK_TOP_PWM4_CTRL<CLKCFG:0x4C> 5. CLK_TOP_PWM5_CTRL<CLKCFG:0x50>	1. PWM control on LVDS_BLON GPIO pin 2. PWM control on LVDS_ENA_BL GPIO pin 3. PWM control on STRP_DATA GPIO pin 4. PWM control on LVDS_DIGON GPIO pin 5. PWM control on TMDS_HPD GPIO pin

Each of the above PWM registers in PWM Controller/GPIO Pins Mapping has the following register fields:

- Bit[0]: Enable the PWM controller
- Bits[12:1]: Number of cycles in pulse period of a 100MHz reference clock
- Bits[24:13]: Number of high cycles in pulse period of a 100MHz reference clock
- Bit[25]: Output enable of the GPIO

The STRP_DATA pin by default is driving low, and register setting <clkcfg:0xE0> Bit[0] = ‘1’ is required before using PWM or GPIO control.

The STRP_DATA pin is also used for core voltage scaling purposes. The CLK_TOP_PWM3_CTRL < CLKCFG:0xCC > Bit[0] = ’0’ is required to enable the graphics device driver to have control on the STRP_DATA pin.

The register settings < nbmisind:0x40 > Bit[8] = ‘1’ and Bit[10] = ’1’ are required for using PWM1 on the LVDS_BLON pin.

3.5 DOS Mode Power Saving

To disable the graphics engine clock branches when in DOS mode, program the following register in *Table 3-7* during boot.

Table 3-7 DOS Mode Power Saving Settings

ASIC Rev	Settings	Function/Comment
A12	1. CG_MISC_INPUT_3[9] = 0 <CLKCFG:0x90> 2. CG_INTGFX_MISC[31:20] = 0x9F3	Write 0x1

3.6 HTPLL VCO Mode Setting

The HTPLL VCO mode setting should be configured in high-speed mode after boot-up.

- <clkcfg:0xD4> CLK_CFG_HTPLL_CNTL Bit[16:15] = “00”
- <clkcfg:0xD4> CLK_CFG_HTPLL_CNTL Bit[2:0] = “010”
- <clkcfg:0xE8> CLK_TOP_SPARE_C Bit[16] = “0”

This page intentionally left blank.

Chapter 4
Memory Initialization

4.1 Memory Initialization

For memory controller programming information, refer to the *AMD RS780 BIOS Developer's Guide*.

This page intentionally left blank.

Chapter 5
PCIE Initialization

5.1 Introduction

This chapter describes the initialization and feature programming of the northbridge PCI Express subsystem. The northbridge implements PCI Express point-to-point links to external devices.

There are 9 configurable PCI Express ports, which can be divided into 3 groups (implemented in hardware as 3 separate cores):

- PCIE-GFX: 2 ports, 16 lanes in total. Each port is configurable from x1 to x8 link. The 2 ports can also be combined to provide 1 x16 port (default configuration).
- PCIE-GPPSB: 1 SB port and 4 GPP ports, 8 lanes in total. The SB port provides a dedicated x4 link to the southbridge. The remaining 4 lanes are distributed across the 4 GPP ports to support 4 different configurations: a) 4:0:0:0:0, b) 4:4:0:0:0, c) 4:2:2:0:0, d) 4:2:1:1:0 and e) 4:1:1:1:1 (default configuration).
- PCIE-GPP: 2 ports, 2 lanes in total. Each port provides a x1 link. The 2 ports can also be combined to provide a x2 link.

5.2 PCI Express Configuration Space

The PCI Express configuration space consists of the following four groups:

- PCIE Port Configuration Space (section *5.2.1*)
- PCIE Core Index Space (section *5.2.2*)
- PCIE Port Index Space (section *5.2.3*)
- PCIE Extended Configuration Space (section *5.2.4*)

5.2.1 PCIE Port Configuration Space

Each PCIE port has a standard Type 1 Virtual PCI-to-PCI bridge header in the PCI configuration space. These are devices 2 through 10 on PCI bus 0.

- GFX Port A: Device 2 (GFX0)
- GFX Port B: Device 3 (GFX1)
- GPPSB Port A: Device 8 (SB link, hidden by default)
- GPPSB Port B: Device 4 (GPP0)
- GPPSB Port C: Device 5 (GPP1)
- GPPSB Port D: Device 6 (GPP2)
- GPPSB Port E: Device 7 (GPP3)
- GPP Port A: Device 9 (GPP4)
- GPP Port B: Device 10 (GPP5)

5.2.2 PCIE Core Index Space

The PCIE Core Index Space contains control and status registers that are generic to all PCIE ports in the northbridge. This register space is accessed through an index/data register pair:

- NB_BIF_NB: NB_PCIE_INDX_ADDR: nbconfig: 0xe0
  - NB_PCIE_INDX_ADDR: [7:0] - Address in PCIE Core
  - GFX_GPPSB_SEL [18:16]:
    - 000 – PCIE GFX Core
    - 001 – PCIE GPPSB Core
    - 010 – PCIE GPP Core
    - 011 – Broadcast to all 3 cores
    - All other values are unused
- NB_BIF_NB: NB_PCIE_INDX_DATA: nbconfig: 0xe4

Note: Registers in the core index space are referenced with the name PCIEIND or BIF_NB.

5.2.3 PCIE Port Index Space

The Port Index Space contains control and status registers that are specific to each port within the core. Each PCIE device implements its own set of registers in this space.

Each PCIE device contains an index/data pair in its Virtual Bridge PCI configuration space to access the Port Index Space registers. Please note the following information for the index/data register pair:

- Index register: bus 0, device X, register 0xE0.
- Data register: bus 0, device X, register 0xE4.

Note: Register descriptions are referenced with the name PCIEIND_P or BIF_NBP.

5.2.4 PCIE Extended Configuration Space

PCI Express extends the PCI configuration space from 256 bytes to 4096 bytes. Extended PCIE configuration space memory maps 4KB for each device. The first 256 bytes of each 4KB are the same as PCI 2.3 configuration registers, and the remaining 3840 bytes are PCIE specific configuration registers.

The northbridge uses NBCFG:NB_BAR3_PCIEXP_MMCFG nbconfig:0x1C (BAR3) to map the PCI Express Extended Configuration Space to a 256MB range within the first 4GB of addressable memory. PCIE devices are accessed by reading/writing to a memory mapped address that is based on the base address in BAR3. The PCIE target address is formed as follows:

- Addr[11:2] = RegNum (Addr[11:8] is extended register field)
- Addr[14:12] = FunNum
- Addr[19:15] = DevNum
- Addr[20 + n-1:20] = BusNum
- Addr[33:20 + n] = Reserved for BAR3 match

Note: ‘n’ indicates how many bits are allocated for the bus number. This value is decided by BAR3BusRange in NBCFG:NB_PCI_ARB[18:16] nbconfig:0x84 register. These relations are listed in *Table 5-1* below:

Table 5-1 NBCFG:NB_PCI_ARB[18:16] nbconfig:0x84 register

ASIC Rev	NBCFG:NB_PCI_ARB[18:16]· nbconfig:0x84	‘n’
RS780 All Revs	3’b001	1
	3’b010	2
	3’b011	3
	3’b100	4
	3’b101	5
	3’b110	6
	3’b111	7
	3’b000	8

Table 5-2 contains the programming procedure to enable BAR3:

Table 5-2 Enabling BAR3

ASIC Rev	Step	Register Settings	Function/Comment
RS780 All Revs	1	NBCFG:NB_IOC_CFG_CNTL[30]=1’b1 nbconfig:0x7c NB_BAR3_PCIEXP_REG_WREN Set bit [30] to 1	Enables BAR3 Register Access.
	2	NBCFG:NB_PCI_ARB[18:16] nbconfig:0x84 BAR3BusRange Program bits [18:16]	Programs BAR3 bus range.
	3	NBCFG:NB_BAR3_PCIEXP_MMCFG[31:21] nbconfig:0x1c MEM_BASE_HIGH NBCFG:NB_BAR3_UPPER_PCIEXP_MMCFG[1:0] nbconfig:0x20 MEM_BASE_UPPER	Programs the BAR3 register.
	4	HTIUNBCFG:NB_HTIU_CFG[28]=1’b1 HTIUNBIND:0x32 NB_BAR3_PCIEXP_ENABLE Set bit [28] to 1	Enables BAR3 decoding.

Note: The BAR3 memory upper address register is nbcfg0x20 (above 4G). The RS780 could support memory up to 16G, so this register has to be set properly.

5.3 Power-On and Reset State

After a Power-On or Reset event the North Bridge puts all of its PCI Express devices into their default states, which are shown in *Table 5-3* below:

Table 5-3 Power-On and Reset State

ASIC Rev	PCI Device Number (PCIE Port)	Link Training	Number of Lanes Supported
RS780 All Revs	Dev 2 (PCIE-GFX Port A)	Disabled	1, 2, 4, 8, or 16
	Dev 3 (PCIE-GFX Port B)	Disabled	1, 2, 4, or 8
	Dev 4 (PCIE-GPPSB Port B)	Disabled	1, 2, or 4
	Dev 5 (PCIE-GPPSB Port C)	Disabled	1
	Dev 6 (PCIE-GPPSB Port D or - C in 4:2:x:x:0 mode)	Disabled	1 or 2
	Dev 7 (PCIE-GPPSB Port E or - D in 4:2:1:1:0 mode)	Disabled	1
	Dev 8 (PCIE-GPPSB Port A)	Enabled	1 or 2 or 4
	Dev 9 (PCIE-GPP Port A)	Disabled	1
	Dev 10 (PCIE-GPP Port B)	Disabled	1
Note: PCI device 8 (Dev 8) does not appear in the PCI configuration space by default.			

5.4 PCIE GFX Configurations

The x16 PCIE GFX interface is fully multiplexed to provide as many directly connected display options as possible. Each output format can be mapped to any of the four-lane groups (0-3, 4-7, 8-11 and 12-15) in the 16 available lanes. However, there are only 3 separate PLLs associated with lanes 0-3 (PLL A), 4-7 (PLL B) and 8-15 (PLL C). The supported configurations in RS780 are detailed below.

5.4.1 PCIE Modes Only

The PCIE GFX core can support up to 2 PCIE devices. Each PCIE device can be a GFX or a GPP device. *Table 5-4* outlines all the supported PCIE configurations.

Table 5-4 PCIE Configurations

Lanes	0 to 3	4 to 7	8 to 11	12 to 15
1	GFX x16 A			
2	GFX x8 A			
3			GFX x8 A	
4	GFX x8 A		GFX x8 B	
5	GPP x4 A			
6		GPP x4 B		
7			GPP x4 A	
8				GPP x4 B
9	GPP x4 A	GPP x4 B		
10	GPP x4 A		GPP x4 B	
11	GPP x4 A			GPP x4 B
12		GPP x4 B	GPP x4 A	
13			GPP x4 A	GPP x4 B
14	GFX x8 A		GPP x4 B	
15	GFX x8 A			GPP x4 B
16		GPP x4 B	GFX x8 A	
17	GPP x4 A		GFX x8 B	

The core should be configured to run in single port mode if only Port A is present and in dual port mode whenever Port B is present (regardless of Port A). The GFX ports are held from link training by default. To enable a GFX port, the corresponding HOLD_TRAINING bit must be set to 0 to allow link training to proceed.

The SBIOS is responsible for programming the lane and clock muxing specific to each case. Refer to *Chapter 7: PCIE Initialization for DDI* for programming details.

Note: This programming must be done before hold training is released.

5.4.1.1 Single Port Configuration (Default)

- PCIE_LINK_CFG – NBMISCIND:0x8
- Set MULTIPORT_CONFIG_GFX (Bits[11:8]) = 4'b0000

Table 5-5 Single Port Configuration (Default)

Device	Possible Link Width	HOLD_TRAINING Bit
Dev 2	1, 2, 4, 8 or 16	PCIE_LINK_CFG[4]

5.4.1.2 Dual Port Configuration

- PCIE_LINK_CFG – NBMISCIND:0x8
- Set MULTIPORT_CONFIG_GFX (Bits[11:8]) = 4'b0101

Table 5-6 Dual Port Configuration

Device	Possible Link Width	HOLD_TRAINING Bit
Dev 2	1, 2, 4 or 8	PCIE_LINK_CFG[4]
Dev 3	1, 2, 4 or 8	PCIE_LINK_CFG[5]

Table 5-7 Dual Port Configuration Register Settings

ASIC Rev	Step	Register Settings	Function/Comment
RS780 All Revs	1	PCIE_NBCFG_REG6 – NBMISCIND:0x36 STRAP_BIF_all_valid (active low) Set bit [31] to 1	De-asserts STRAP_BIF_all_valid for PCIE-GFX core.
	2	PCIE_LINK_CFG - NBMISCIND:0x8 MULTIPORT_CONFIG_GFX Set bits[11:8] to 4'b0101	Enables dual port configuration
	3	PCIE_NBCFG_REG6 – NBMISCIND:0x36 STRAP_BIF_all_valid (active low) Set bit [31] to 0	Asserts STRAP_BIF_all_valid for PCIE-GFX core.

5.4.2 DDI Modes Only

In the RS780, DDI is a collective term used to describe the supported display formats, which include DVI, HDMI and DisplayPort. DVI and HDMI can run in either single link (x4, DDI_SL) or dual link (x8, DDL_DL) mode. DisplayPort can run in x1, x2 or x4 mode.

Due to the fact that lanes 8-15 are sharing PLL C, the subgroups of lanes 8-11 and lanes 12-15 cannot be used simultaneously to support 2 independent display outputs. All the supported configurations are detailed in *Table 5-8* below.

Table 5-8 DDI Configurations

Lanes	0 to 3	4 to 7	8 to 11	12 to 15
1	DDI_SL			
2		DDI_SL		
3			DDI_SL	
4				DDI_SL
5	DDI_SL	DDI_SL		
6	DDI_DL			
7	DDI_SL		DDI_SL	
8	DDI_SL			DDI_SL
9		DDI_SL	DDI_SL	
10		DDI_SL		DDI_SL
11			DDI_DL	

The VBIOS/Driver will be responsible for DDI programming sequence. In the case of DDI modes only, the SBIOS can pass control over to the VBIOS/Driver after powering down any unused lanes and PLLs.

5.4.3 PCIE and DDI Combined Modes

PCIE and DDI modes can also be combined to use simultaneously, the supported configurations are listed in the following chart. For each group of 8 lanes (lanes 0-7 and lanes 8-15), only 1 DDI or 1 PCIE device can be used at any one time.

Table 5-9 PCIE and DDI Configurations

Lanes	0 to 3	4 to 7	8 to 11	12 to 15
1	GFX x8 A		DDI_SL	
2	GFX x8 A			DDI_SL
3	GFX x8 A		DDI_DL	
4	DDI_SL		GFX x8 A	
5		DDI_SL	GFX x8 A	
6	DDI_SL	DDI_SL	GFX x8 A	
7	DDI_DL		GFX x8 A	
8	GPP x4 A	GPP x4 B	DDI_SL	
9	GPP x4 A	GPP x4 B		DDI_SL
10	GPP x4 A	GPP x4 B	DDI_DL	
11	DDI_SL		GPP x4 A	GPP x4 B
12		DDI_SL	GPP x4 A	GPP x4 B
13	DDI_SL	DDI_SL	GPP x4 A	GPP x4 B
14	DDI_DL		GPP x4 A	GPP x4 B

The SBIOS is responsible for PCIE mode programming (same as section *5.4.1*) and the VBIOS/Driver is responsible for completing the DDI programming.

5.5 PCIE GPPSB Configurations

The SB link shares the same PCIE core with 4 other GPP devices. The SB port has 4 dedicated lanes and the 4 GPP ports share among 4 lanes, for a total of 8 lanes. The following configurations are supported:

- Default Configuration E (4:1:1:1:1), STRAP_BIF_LINK_CONFIG_GPPSB = 4'b0100 (section *5.5.2*)
- Configuration A (4:0:0:0:0), STRAP_BIF_LINK_CONFIG_GPPSB = 4'b0000 (section *5.5.3*)
- Configuration B (4:4:0:0:0), STRAP_BIF_LINK_CONFIG_GPPSB = 4'b0001 (section *5.5.4*)
- Configuration C (4:2:2:0:0), STRAP_BIF_LINK_CONFIG_GPPSB = 4'b0010 (section *5.5.5*)
- Configuration D (4:2:1:1:0), STRAP_BIF_LINK_CONFIG_GPPSB = 4'b0011 (section *5.5.6*)

Note: STRAP_BIF_LINK_CONFIG_GPPSB = StrapsOutputMux_7 – NBMISCIND:0x67, bits [7:4]

5.5.1 Device Remapping

The device remapping feature provides the ability to map a PCIE slot to any device number. There are 2 possible ways of remapping:

- Manual Remapping

  The slot-to-device number mapping is specified through the register setting in *Table 5-10*:

Table 5-10 Device Remapping

NBCFG: NB_PROG_DEVICE_REMAP_0 NBMISCIND:0x20	
NB_PROG_DEVMAP_EN, bit [0]	0=Device mapping disabled (default) 1=Device remapping enabled using GPP_PORT*_DEVMAP
GPP_PORTB_DEVMAP, bits [7:4]	0=Map to Device 4 1=Map to Device 5 2=Map to Device 6 3=Map to Device 7
GPP_PORTC_DEVMAP, bits [11:8]	0=Map to Device 4 1=Map to Device 5 2=Map to Device 6 3=Map to Device 7
GPP_PORTD_DEVMAP, bits [15:12]	0=Map to Device 4 1=Map to Device 5 2=Map to Device 6 3=Map to Device 7
GPP_PORTE_DEVMAP, bits [19:16]	0=Map to Device 4 1=Map to Device 5 2=Map to Device 6 3=Map to Device 7

- Automatic Remapping

  The remapping is done automatically according to the PCIE GPPSB configuration so that the slot to device number mapping is always the same. The SBIOS should program the field in *Table 5-11* to enable this feature by default.

Table 5-11 Automatic Remapping

NBCFG: NB_PROG_DEVICE_REMAP_0 NBMISCIND:0x20	
IOC_PCIE_Dev_Remap_Dis, bit [1]	0=Automatic device remapping enabled 1=Automatic device remapping disabled

5.5.2 Default Configuration E (STRAP_BIF_LINK_CONFIG_GPPSB = 4'b0100)

The default configuration only has the SB port enabled. The GPP ports are held from link training. To enable the GPP ports, the corresponding HOLD_TRAINING bit must be set to 0 to allow link training to proceed.

- PCIE_LINK_CFG – NBMISCIND:0x8

Table 5-12 Default Configuration E

Device	Associated Lanes	HOLD_TRAINING Bit
Dev 8 (Port A)	SB_RX/TX0 SB_RX/TX1 SB_RX/TX2 SB_RX/TX3	PCIE_LINK_CFG[20]
Dev 4 (Port B)	GPP_RX/TX0	PCIE_LINK_CFG[21]
Dev 5 (Port C)	GPP_RX/TX1	PCIE_LINK_CFG[22]
Dev 6 (Port D)	GPP_RX/TX2	PCIE_LINK_CFG[23]
Dev 7 (Port E)	GPP_RX/TX3	PCIE_LINK_CFG[24]

5.5.3 Configuration A (STRAP_BIF_LINK_CONFIG_GPPSB = 4'b0000)

This configuration only supports the SB port and no GPP ports.

Table 5-13 Configuration A

Device	Associated Lanes	HOLD_TRAINING Bit
Dev 8 (Port A)	SB_RX/TX0 SB_RX/TX1 SB_RX/TX2 SB_RX/TX3	PCIE_LINK_CFG[20]

5.5.4 Configuration B (STRAP_BIF_LINK_CONFIG_GPPSB = 4'b0001)

This configuration is primarily used to support high performance GPP devices. Both the SB and the GPP links will have 4-lanes.

Table 5-14 Configuration B

Device	Associated Lanes	HOLD_TRAINING Bit
Dev 8 (Port A)	SB_RX/TX0 SB_RX/TX1 SB_RX/TX2 SB_RX/TX3	PCIE_LINK_CFG[20]
Dev 4 (Port B)	GPP_RX/TX0 GPP_RX/TX1 GPP_RX/TX2 GPP_RX/TX3	PCIE_LINK_CFG[21]

5.5.5 Configuration C (STRAP_BIF_LINK_CONFIG_GPPSB = 4'b0010)

In addition to the 4-lane SB, this configuration supports two x2 GPP devices.

Table 5-15 Configuration C

Device	Associated Lanes	HOLD_TRAINING Bit
Dev 8 (Port A)	SB_RX/TX0 SB_RX/TX1 SB_RX/TX2 SB_RX/TX3	PCIE_LINK_CFG[20]
Dev 4 (Port B)	GPP_RX/TX0 GPP_RX/TX1	PCIE_LINK_CFG[21]
Dev 6 (Port C)	GPP_RX/TX2 GPP_RX/TX3	PCIE_LINK_CFG[22]

5.5.6 Configuration D (STRAP_BIF_LINK_CONFIG_GPPSB = 4'b0011)

This configuration provides a x4 SB, one x2 GPP and two x1 GPP ports.

Table 5-16 Configuration D

Device	Associated Lanes	HOLD_TRAINING Bit
Dev 8 (Port A)	SB_RX/TX0 SB_RX/TX1 SB_RX/TX2 SB_RX/TX3	PCIE_LINK_CFG[20]
Dev 4 (Port B)	GPP_RX/TX0 GPP_RX/TX1	PCIE_LINK_CFG[21]
Dev 6 (Port C)	GPP_RX/TX2	PCIE_LINK_CFG[22]
Dev 7 (Port D)	GPP_RX/TX3	PCIE_LINK_CFG[23]

5.5.7 Switching GPPSB Configurations

Since the SB link must be alive for SBIOS code execution, a special programming sequence must be performed to switch between configurations. This sequence will cause the hardware to assert reset to the GPPSB core, load the new configuration and deassert reset for the link to re-train at the new configuration.

Table 5-17 Switching GPPSB Configurations

ASIC Rev	Step	Register Settings	Function/Comment
RS780 All Revs	1	PCIE_NBCFG_REG7 - NBMISCIND:0x37 reconfig_gppsb_en Set bit[12] to 1	Enables GPPSB reconfiguration
	2	PCIE_NBCFG_REG7 - NBMISCIND:0x37 reconfig_gppsb_atomic_reset_dis Set bit[15] to 1	
	3	PCIE_NBCFG_REG7 - NBMISCIND:0x37 reconfig_gppsb_link_config_xfer_mode Set bit[17] to 1	
	4a	StrapsOutputMux_7 - NBMISCIND:0x67 STRAP_BIF_LINK_CONFIG_GPPSB Set bits[7:4] to desired configuration	Sets desired GPPSB configurations
	4b	StrapsOutputMux_6 – NBMISCIND:0x66 STRAP_BIF_all_valid (active low) Set bit [31] to 1	De-asserts STRAP_BIF_all_valid for PCIE-GPPSB core.
	5	PCIE_NBCFG_REG7 - NBMISCIND:0x37 reconfig_gppsb Read bit[14]	
	6	PCIE_NBCFG_REG7 - NBMISCIND:0x37 reconfig_gppsb Write the inversion of bit[14] read back from step 5	
	7	StrapsOutputMux_6 – NBMISCIND:0x66 STRAP_BIF_all_valid (active low) Set bit [31] to 0	Asserts STRAP_BIF_all_valid for PCIE-GPPSB core.
	8	Wait for 1ms	
	9	PCIE_LC_STATE0 – PCIEIND_P: 0xA5 in Dev 8 LC_CURRENT_STATE Poll for bits[5:0] = 5'h10	Waits until SB has trained to L0
	10	PCIE_LC_STATE0 – PCIEIND_P: 0x12A in Dev 8 VC_NEGOTIATION_PENDING Poll for bit[1] = 0	Ensures that virtual channel negotiation is completed.

5.6 PCIE GPP Configurations

The GPP core only has 2 available lanes, supporting either two x1 GPP devices or one x2 GPP device. The GPP ports are held from link training. To enable the GPP ports, the corresponding HOLD_TRAINING bit must be set to 0 to allow link training to proceed.

5.6.1 Default Configuration D

- PCIE_NBCFG_REG15 – NBMISCIND: 0x2d
- Set LINK_CONFIG (Bits[10:7]) = 4'b0011

Table 5-18 Default Configuration D

Device	Associated Lanes	HOLD_TRAINING Bit
Dev 9	GPP_RX/TX4	PCIE_NBCFG_REG15[4]
Dev 10	GPP_RX/TX5	PCIE_NBCFG_REG15[5]

5.6.2 Configuration C

- PCIE_NBCFG_REG15 – NBMISCIND: 0x2d
- Set LINK_CONFIG (Bits[10:7]) = 4'b0010

Table 5-19 Configuration C

Device	Associated Lanes	HOLD_TRAINING Bit
Dev 9	GPP_RX/TX4 GPP_RX/TX5	PCIE_NBCFG_REG15[4]

Table 5-20 Configuration C Register Settings

ASIC Rev	Step	Register Settings	Function/Comment
RS780 All Revs	1	PCIE_NBCFG_REGA – NBMISCIND:0x22 STRAP_BIF_all_valid (active low) Set bit [14] to 1	De-asserts STRAP_BIF_all_valid for PCIE-GPP core.
	2	PCIE_NBCFG_REG15 – NBMISCIND:0x2d LINK_CONFIG Set bits[10:7] to 4'b0010	Sets configuration to C
	3	PCIE_NBCFG_REG6 – NBMISCIND:0x22 STRAP_BIF_all_valid (active low) Set bit [14] to 0	Asserts STRAP_BIF_all_valid for PCIE-GPP core.

5.7 PCIE Link Training Sequence

The link training sequence cannot be interrupted and no other BIOS code can be added in between.

Note: The contents of *Table 5-21* continues onto the next two pages of this document.

Table 5-21 PCIE Link Training Sequence

ASIC Rev	Step	Register Setting	Function/Comment
RS780 All Revs	1	Release hold training for Device * by clearing the corresponding HOLD_TRAINING bit to 0.	The HOLD_TRAINING bit can be found in section *5.4* (GFX), section *5.5* (GPPSB), and section *5.6* (GPP).
	2	Delay 200us	
	3	BIF_NBP:PCIE_LC_STATE0 ·PCIEIND_P:0xA5 Read back the following values: LC_CURRENT_STATE = [5:0] LC_PREV_STATE1 = [13:8] LC_PREV_STATE2 = [21:16] LC_PREV_STATE3 = [29:24] If any read back value is 6'h3F -> then perform CF9 reset; If LC_CURRENT_STATE = 6'h00 to 6'h04, then no device is present. Keep checking for up to 40ms, if no device is present → Go to Step 8. Otherwise, go to Step 4.	Detects if there is any card present from reading back PCIE_LC_STATE0 in Port Index space of Device*.

RS780 All Revs	4	BIF_NBP:PCIE_LC_STATE0 ·PCIEIND_P:0xA5 LC_CURRENT_STATE Read back bits [5:0]. If LC_CURRENT_STATE = 6'h06 -> read back current link width by reading bits [6:4] of the following register: BIF_NBP:PCIE_LC_LINK_WIDTH_CNTL[6:4] PCIEIND_P:0xA2 LC_LINK_WIDTH_RD If [6:4]=3'h4 and lane reversal is not enable set: BIF_NB:PCIE_P_PAD_FORCE_DIS[7:4]=4'hf· PCIEIND:0x65 BIF_NB:PCIE_P_PAD_FORCE_DIS[15:12]=4'hf· PCIEIND:0x65 If [6:4]=3'h4 and lane reversal is enabled set: BIF_NB:PCIE_P_PAD_FORCE_DIS[3:0]=4'hf· PCIEIND:0x65 BIF_NB:PCIE_P_PAD_FORCE_D[11:8]=4'hf· PCIEIND:0x65 If [6:4]=3'h3 and lane reversal is not enabled set: BIF_NB:PCIE_P_PAD_FORCE_DIS[7:2]=4'hf· PCIEIND:0x65 BIF_NB:PCIE_P_PAD_FORCE_DIS[15:10]=4'hf· PCIEIND:0x65 If [6:4]=3'h3 and lane reversal is enabled set: BIF_NB:PCIE_P_PAD_FORCE_DIS[5:0]=4'hf· PCIEIND:0x65 BIF_NB:PCIE_P_PAD_FORCE_D[13:8]=4'hf· PCIEIND:0x65 If [6:4]=3'h2 and lane reversal is not enabled set: BIF_NB:PCIE_P_PAD_FORCE_DIS[7:1]=4'hf· PCIEIND:0x65 BIF_NB:PCIE_P_PAD_FORCE_DIS[15:9]=4'hf· PCIEIND:0x65 If [6:4]=3'h2 and lane reversal is enabled set: BIF_NB:PCIE_P_PAD_FORCE_DIS[6:0]=4'hf· PCIEIND:0x65 BIF_NB:PCIE_P_PAD_FORCE_D[14:8]=4'hf· PCIEIND:0x65 Toggle GPIO reset to the PCIe slot. Got to Step5. If [6:4] takes any other values than described above; go to Step 5.	CMOS option (disabled by default). This programming sequence is required if some lanes on the end point are broken. It is specific for GFX card in single slot configuration.
	5	BIF_NBP:PCIE_LC_STATE0 ·PCIEIND_P:0xA5 LC_CURRENT_STATE Read back bits [5:0]. If LC_CURRENT_STATE = 6'h07 -> Device is in compliance state (training sequence is done). Move to train the next device; If LC_CURRENT_STATE = 6'h10 -> go to step 6 Otherwise, keep polling for up to 2 seconds, then perform CF9 reset. This should only be repeated for a maximum of 15 times. If LC_CURRENT_STATE can not reach 6'h10 or 6'h07 -> go to Step 9.	Detects if link is in Compliance State or it is trained to L0 from reading back PCIE_LC_STATE0 [5:0] in Device*.

RS780 All Revs	6	BIF_NBP:PCIE_VC0_RESOURCE_STATUS[1]· pcieConfigDev*:0x12a VC_NEGOTIATION_PENDING Read bit[1] Read back value 0 means link negotiation is successful -> go to Step 8 Read back value 1 means the link needs to be re-trained -> go to Step 7	Detects if Data Link Negotiation is performed, by reading bit [1] of BIF_NBP:PCIE_VC0_RESOURCE_STATUS[1]· pcieConfigDev*: 0x12a
	7	Set bit[8] = 1 and set bits[2:0] to be equal to bits[6:4] of the following register: BIF_NBP:PCIE_LC_LINK_WIDTH_CNTL PCIEIND_P:0xA2 LC_RECONFIG_NOW, bit[8] LC_LINK_WIDTH, bit[2:0] <- LC_LINK_WIDTH_RD, bit[6:4] Wait for 5ms after the bits above are set, then go back to Step 2 (stay in this loop indefinitely).	Retrains the link.
	8	For AMD GFX Cards only -> go to section *5.12.9.1* After the implementation of RV370/RV380 Initialization Workaround, and after reading back the Device ID for the non-AMD Device, as well as AMD devices for which this workaround is not applicable, perform the following: BIF_NB:PCIE_CI_CNTL[9]=1'b0PCIEIND:0x20 CI_RC_ORDERING_DIS Clear bit[9] to 0.	RV370/RV380 Initialization Workaround.
	9	Hide the bridge for non-hot-plug device; Set the hold training bit to 1 (see Step 1); Power down the port, then move to the train the next device.	When no device is detected or the link cannot be trained properly, than these 3 steps should be performed.

5.8 Overall PCIE Programming Sequence

The overall PCIE programming sequence can be divided into the following parts:

- PCIE-GFX Core Initialization (section *5.9*)
- PCIE-GPPSB and PCIE-GPP Cores Initialization (*5.10*)
- Dynamic Link Width Control (*5.11*)
- PCI Enumeration and Special Features Programming Sequence (*5.12*)

Note: Section *5.9* and section *5.10* should be implemented as separate threads in the SBIOS so that they can be executed in parallel.

5.9 PCIE-GFX Core Initialization

The initialization sequence should be executed in the same order as the sections are organized.

5.9.1 REFCLK Options

- External Clock Mode (Default Mode): an external clock chip is used to drive a dedicated GFX REFCLK

  Note: For ASIC Rev A11 ONLY - When accessing PCIE_NBCFG_REG10 (NBMISCIND:0x28), the write enable bit must be set for both reads and writes.

Table 5-22 External Clock Mode

ASIC Rev	Step	Register Settings	Function/Comment
RS780 All Revs	1	PCIE_NBCFG_REG8 – NBMISCIND:0x38 B_PCLK_BIDIR_TX_EN Set bit[29] to 0	Disables the GFX REFCLK transmitter so that the GFX REFCLK PAD can be driven by an external source.
	2	PCIE_NBCFG_REG8 – NBMISCIND:0x38 B_PCLK_BIDIR_RX_EN Set bit[28] to 1	Enables GFX REFCLK receiver to receive the REFCLK from an external source.
	3	PCIE_NBCFG_REG10 – NBMISCIND:0x28 B_PREFCLK_SEL_A Set bits[7:6] to 2'b01	Selects the GFX REFCLK to be the source for PLL A.
	4	PCIE_NBCFG_REG10 – NBMISCIND:0x28 B_PREFCLK_SEL_B Set bits[9:8] to 2'b01	Selects the GFX REFCLK to be the source for PLL B.
	5	PCIE_NBCFG_REG10 – NBMISCIND:0x28 B_PREFCLK_SEL_C Set bits[11:10] to 2'b01	Selects the GFX REFCLK to be the source for PLL C. All 3 PLLs will be configured to have the same source in SBIOS. The PLLs may have different sources in DDI modes and the setting will be overwritten by the VBIOS/Driver.
	6	StrapsOutputMux_C – NBMISCIND:0x6C REFCLK_BIDIR_SEL Set bit[31] to 1	Selects the single ended GFX REFCLK to be the source for core logic.

- Internal Clock Mode: SB REFCLK is routed internally to be the source of GFX REFCLK

Table 5-23 Internal Clock Mode

ASIC Rev	Step	Register Settings	Function/Comment
RS780 All Revs	1	PCIE_NBCFG_REG8 – NBMISCIND:0x38 B_PCLK_BIDIR_TX_EN Set bit[29] to 1	Enables the GFX REFCLK transmitter so that the GFX REFCLK PAD can be driven by the SB REFCLK.
	2	PCIE_NBCFG_REG8 – NBMISCIND:0x38 B_PCLK_BIDIR_RX_EN Set bit[28] to 0	Disables GFX REFCLK receiver from receiving the REFCLK from an external source.
	3	PCIE_NBCFG_REG10 – NBMISCIND:0x28 B_PREFCLK_SEL_A Set bits[7:6] to 2'b00	Selects SB REFCLK to be the source for PLL A.
	4	PCIE_NBCFG_REG10 – NBMISCIND:0x28 B_PREFCLK_SEL_B Set bits[9:8] to 2'b00	Selects SB REFCLK to be the source for PLL B.

ASIC Rev	Step	Register Settings	Function/Comment
	5	PCIE_NBCFG_REG10 – NBMISCIND:0x28 B_PREFCLK_SEL_C Set bits[11:10] to 2'b00	Selects SB REFCLK to be the source for PLL C. All 3 PLLs will be configured to have the same source in SBIOS. The PLLs may have different sources in DDI modes and the setting will be overwritten by the VBIOS/Driver.
	6	StrapsOutputMux_C – NBMISCIND:0x6C REFCLK_BIDIR_SEL Set bit[31] to 0	Selects the single ended SB REFCLK to be the source for core logic.

5.9.2 Lane Reversal (CMOS Option - Disabled by Default)

There should be 2 CMOS options for Port A and Port B:

- PCIE-GFX Port A Lane Reversal (Single/Dual Configuration)
- PCIE-GFX Port B Lane Reversal (Dual Configuration)

Table 5-24 Lane Reversal (CMOS Option - Disabled by Default)

ASIC Rev	Step	Register Settings	Function/Comment
RS780 All Revs	1	PCIE_NBCFG_REG6 – NBMISCIND:0x36 STRAP_BIF_all_valid (active low) Set bit [31] to 1	De-asserts STRAP_BIF_all_valid for PCIE-GFX core.
	2	PCIE_NBCFG_REG3 - NBMISCIND:0x33 STRAP_BIF_REVERSE_LANES_GFX_A Set bit[2] to 1	Enables lane reversal for GFX Port A
	3	PCIE_NBCFG_REG3 - NBMISCIND:0x33 STRAP_BIF_REVERSE_LANES_GFX_B Set bit[3] to 1	Enables lane reversal for GFX Port B
	4	PCIE_NBCFG_REG6 – NBMISCIND:0x36 STRAP_BIF_all_valid (active low) Set bit [31] to 0	Asserts STRAP_BIF_all_valid for PCIE-GFX core.

5.9.3 GFX Overclocking

Table 5-25 GFX Overclocking

ASIC Rev	Step	Register Settings	Function/Comment
RS780 All Revs	1	Program the external clock chip to a different frequency	
	2	PCIE_NBCFG_REG6 – NBMISCIND:0x36 B_P90PLL_IBIAS Set bits[13:4] to 10'hB5	Increases PLL BW for 6G operation.

5.9.4 Reset PCIE-GFX Core

Table 5-26 Reset PCIE-GFX Core

ASIC Rev	Step	Register Settings	Function/Comment
RS780 All Revs	1	PCIE_LINK_CFG – NBMISCIND:0x8 CALIB_RESET_GFX GLOBAL_RESET_GFX Set bits[15:14] to 2'b11	Asserts both calibration reset and global reset
	2	PCIE_LINK_CFG – NBMISCIND:0x8 CALIB_RESET_GFX Set bit[14] to 0	De-asserts calibration reset
	3	Wait for at least 200us	
	4	PCIE_LINK_CFG – NBMISCIND:0x8 GLOBAL_RESET_GFX Set bit[15] to 0	De-asserts global reset

5.9.5 Reset PCIE-GFX Slot

Table 5-27 Reset PCIE-GFX Slot

ASIC Rev	Step	Register Settings	Function/Comment
RS780 All Revs	1	Desktop reference board: SLP_S2/GPM9# Mobile reference board: INTH#/GPIO36	Program the GPIO in the SB700 to reset the PCIE-GFX slot.
	2	Use the Delay Training CMOS Option described in the next section	Program delay link training timer.

5.9.6 Delay Training Option (CMOS Option – Default 2ms)

Some PCIE devices may require additional initialization time after a reset is de-asserted before they can train the link properly. A typical delay of 2ms is sufficient for most devices. In order to accommodate devices that require additional delay, a CMOS option with a selectable time from 0 to 200 ms, with increments of 1ms, should be implemented. Training to the slots should not be released until the timer expires.

5.9.7 Transmitter Drive Strength (CMOS Option – Default 22mA)

The transmitter driving strength can be adjusted to give better signal integrity for the PCIE lanes.

Table 5-28 Transmitter Driver Strength (CMOS Option - Default 22mA)

ASIC Rev	Step	Register Settings	Function/Comment
RS780 All Revs	1	PCIE_NBCFG_REG5 – NBMISCIND:0x35 B_P90TX_DRV_STR[1:0] for GFX Set bits[3:2] according to CMOS option	Possible CMOS options: 2'b00: 18mA nominal 2'b01: 20mA nominal 2'b10: 22mA nominal 2'b11: 24mA nominal

5.9.8 Program PCIE Memory Mapped Configuration Space

Table 5-29 Program PCIE Memory Mapped Configuration Space

ASIC Rev	Step	Register Settings	Function/Comment
RS780 All Revs	1	NB_IOC_CFG_CNTL – NBCONFIG:0x7C NB_BAR3_PCIEXP_REG_WREN Set bit[30] to 1	Enables writes to BAR3 register
	2	NB_PCI_ARB – NBCONFIG:0x84 BAR3BusRange Set bits[18:16] to 3'b000	See section *5.2.4* for details
	3	NB_BAR3_PCIEXP_MMCFG – NBCONFIG:0x1C MEM_BASE_HIGH Set bits[31:0] to 32'hE0000000	
	4	NB_BAR3_UPPER_PCIEXP_MMCFG – NBCONFIG:0x20 MEM_BASE_UPPER Set bits[1:0] to 2'b00	Sets memory upper address to be above 4G
	5	NB_IOC_CFG_CNTL – NBCONFIG:0x7C NB_BAR3_PCIEXP_REG_WREN Set bits[30] to 0	Disables BAR3 writes
	6	NB_HTIU_CFG – HTIUNBIND:0x32 NB_BAR3_PCIEXP_ENABLE Set bits[28] to 1	Enables BAR3 decoding

5.9.9 GEN1 Software Compliance (CMOS Option – Disabled by Default)

There should be 2 CMOS options for Port A and Port B:

- PCIE-GFX Port A GEN1 Software Compliance (Single/Dual Configuration)
- PCIE-GFX Port B GEN1 Software Compliance (Dual Configuration)

Table 5-30 GEN1 Software Compliance (CMOS Option - Disabled by Default)

ASIC Rev	Register Settings	Function/Comment
RS780 All Revs	BIF_NBP:PCIEP_STRAP_LC -- PCIEIND_P:0xC0 STRAP_FORCE_COMPLIANCE Set bit[13] to 1	Forces transmitter to output compliance pattern at Gen1 rate.

5.9.10 GEN2 Software Compliance (CMOS Option – Disabled by Default)

There should be 2 CMOS options for Port A and Port B:

- PCIE-GFX Port A GEN2 Software Compliance (Single/Dual Configuration)
- PCIE-GFX Port B GEN2 Software Compliance (Dual Configuration)

Table 5-31 GEN2 Software Compliance (CMOS Option - Disabled by Default)

ASIC Rev	Step	Register Settings	Function/Comment
RS780 All Revs	1	BIF_NBP:PCIE_LC_SPEED_CNTL -- PCIEIND_P:0xA4 LC_GEN2_EN_STRAP Set bit[0] to 1	Enables GEN2 capability of the device.
	2	NBCFG: PCIE_NBCFG_REG4 – NBMISCIND: 0x34 STRAP_BIF_DE_EMPHASIS_SEL (port A) Set bit[5] to 1 NBCFG: PCIE_NBCFG_REGA – NBMISCIND: 0x22 STRAP_BIF_DE_EMPHASIS_SEL (port B) Set bit[6] to 1	Advertises -3.5 dB de-emphasis value in TS1 Data Rate Identifier
	3	BIF_NBP:LINK_CNTL2 -- pcieConfigDev*:0x88 TARGET_LINK_SPEED Set bits[3:0] to 4'h2	Advertise the link speed to be Gen2.
	4	BIF_NBP:LINK_CNTL2 -- pcieConfigDev*:0x88 ENTER_COMPLIANCE Set bit[4] to 1	Forces transmitter to output compliance pattern at Gen2 rate.

5.9.11 De-Emphasis Strength -3.5dB/-6dB in GEN2 (CMOS Option - Disabled by Default)

Table 5-32 De-Emphasis Strength -3.5dB/-6dB in GEN2 Settings

ASIC Rev	Step	Register Settings	Function/Comment
RS780 All Revs	1	NBCFG: PCIE_NBCFG_REG4 – NBMISCIND: 0x34 STRAP_BIF_DE_EMPHASIS_SEL (port A) Set bit[5] to 0 NBCFG: PCIE_NBCFG_REGA – NBMISCIND: 0x22 STRAP_BIF_DE_EMPHASIS_SEL (port B) Set bit[6] to 0	Advertises -6 dB de-emphasis value in TS1 Data Rate Identifier. Default value is -3.5dB.

5.9.12 Core Initialization

Table 5-33 Core Initialization

ASIC Rev	Step	Register Settings	Function/Comment
RS780 All Revs	1	BIF_NBP:PCIE_RX_CNTL -- PCIEIND_P:0x70 RX_RCB_CPL_TIMEOUT Set bits[18:16] to 3'h4	Sets timeout to 100ms/4 = 25ms
	2	BIF_NBP:PCIE_RX_CNTL -- PCIEIND_P:0x70 RX_RCB_CPL_TIMEOUT_MODE Set bit[19] to 1	RCB Cpl timeout on link down (to shorten enumeration time).
	3	BIF_NB:PCIE_CI_CNTL -- PCIEIND:0x20 CI_SLV_ORDERING_DIS Clear bit[8] to 0	Enable slave ordering logic.
	4	BIF_NB:PCIE_CNTL -- PCIEIND:0x10 RX_SB_ADJ_PAYLOAD_SIZE Set bits[12:10] to 3'b100	Sets DMA payload size to 64 bytes.
	5	BIF_NBP:PCIEP_PORT_CNTL -- PCIEIND_P :0x10 SLV_PORT_REQ_EN Clear bit[0] to 0	Blocks DMA traffic during C3 state.
	6	BIF_NB:PCIE_CI_CNTL -- PCIEIND:0x20 CI_RC_ORDERING_DIS Set bit[9] to 1	Disables RC ordering logic so that Picard can return a dummy master completion without receiving an ACK.
	7	BIF_NB:PCIE_HW_DEBUG -- PCIEIND:0x2 HW_00_DEBUG (REGS_DLP_IGNORE_IN_L1_EN) Set bit [0] to 1	Ignores DLLPs during L1 so that txclk can be turned off .
	8	BIF_NBP:PCIE_LC_TRAINING_CNTL -- PCIEIND_P:0xA1 LC_DONT_GO_TO_L0S_IF_L1_ARMED Set bit [11] to 1	Prevents LC to go from L0 to Rcv_L0s if L1 is armed.
	9	NBCFG:StrapsOutputMux_A -- NBMISCIND:0x6A Set bit [17] to 1	CMGOOD_OVERRIDE for end point initiated lane degradation.
	10	BIF_NBP:PCIE_LC_CNTL2 -- PCIEIND_P:0xB1 LC_WAIT_FOR_LANES_IN_LW_NEG Set bit [23] to 1	Sets the timer in Config state from 20us to 1us for short and up/down reconfiguration.
	11	BIF_NBP:PCIE_LC_CNTL2 -- PCIEIND_P:0xB1 LC_DEASSERT_RX_EN_IN_L0s Set bit[19] to 1	De-asserts RX_EN in L0s.
	12	BIF_NBP:PCIE_LC_CNTL2 -- PCIEIND_P:0xB1 LC_ENABLE_RX_CR_EN_DEASSERTION Set bit [28] to 1	Enables de-assertion of PG2RX_CR_EN to lock clock recovery parameter when lane is in electrical idle in L0s.
	13	NBCFG:PCIE_NBCFG_REG4 -- NBMISCIND:0x34 B_P90RX_INCAL_FORCE Set bit[10] to 1	Turns off offset calibration.
	14	NBCFG:PCIE_NBCFG_REG4 -- NBMISCIND:0x34 B_P90RX_CLKG_EN Set bit[22] to 1	Enables Rx Clock gating in CDR

ASIC Rev	Step	Register Settings	Function/Comment
	15	BIF_NBP:PCIE_LC_CNTL -- PCIEIND_P:0xA0 LC_16X_CLEAR_TX_PIPE Set bits[7:4] to4'h3	Sets number of TX Clocks to drain TX Pipe to 3.
	16	BIF_NB:PCIE_P_CNTL -- PCIEIND:0x40 P_ELEC_IDLE_MODE Set bits [15:14] to 2'b10	Lets PI use Electrical Idle from PHY when turning off PLL in L1 at Gen2 speed instead Inferred Electrical Idle. NOTE: LC still uses Inferred Electrical Idle.
	17	BIF_NBP:PCIE_LC_CNTL2 -- PCIEIND_P:0xB1 LC_BLOCK_EL_IDLE_IN_L0 Set bit[20] to 1	Prevents the Electrical Idle from causing a transition from Rcv_L0 to Rcv_L0s.
	18	BIF_NBP:PCIE_LC_TRAINING_CNTL -- PCIEIND_P:0xA1 LC_DONT_GO_TO_L0S_IF_L1_ARMED Set bit [11] to 1.	Prevents the LTSSM from going to Rcv_L0s if it has already acknowledged a request to go to L1.
	19	BIF_NB:PCIE_P_CNTL -- PCIEIND:0x40 RXP_REALIGN_ON_EACH_TSX_OR_SKP Set bit[28] to 0	LDSK only taking deskew on deskewing error detect
	20	BIF_NB:PCIE_STRAP_PI -- PCIEIND:0xC2 STRAP_LDSK_X1_BYPASS Set bit [14] to 1	Bypasses lane de-skew logic if in x1
	21	NBCFG:PCIE_NBCFG_REG5 -- NBMISCIND:0x35 B_PG2PLL_IDLEDET_TH[1:0] Set bits [22:21] to 2'b10	Sets Electrical Idle Threshold
	22	NBCFG: PCIE_NBCFG_REG4 – NBMISCIND: 0x34 STRAP_BIF_DE_EMPHASIS_SEL (port A) Set bit[5] to 0 NBCFG: PCIE_NBCFG_REGA – NBMISCIND: 0x22 STRAP_BIF_DE_EMPHASIS_SEL (port B) Set bit[6] to 0	Advertises -6 dB de-emphasis value in TS1 Data Rate Identifier Only if CMOS Option in section *5.9.11* is enabled.
	23	BIF_NBP:PCIE_LC_SPEED_CNTL -- PCIEIND_P:0xA4 LC_GEN2_EN_STRAP Set bit[0] to 0	Disables GEN2 capability of the device.
	24	BIF_NBP:PCIE_LC_LINK_WIDTH_CNTL[13]=1'b1·PCIEIND_P:0xA2 LC_UPCONFIGURE_DIS Set bit[13] to 1	Disables advertising Upconfigure Support.
	25	NBCFG: PCIE_NBCFG_REG9 – NBMISCIND: 0x39 STRAP_BIF_DSN_EN Set bit[10] to 0	
	26	BIF_NBP:PCIE_STRAP_MISC2[0] – PCIEIND:0xC1 STRAP_LINK_BW_NOTIFICATION_CAP_EN Set bit[0] to 1	This capacity is required since links wider than x1 and/or multiple link speed are supported
RS780 A13 and beyond	27	PCIEP_HW_DEBUG - PCIEIND_P:0x02 HW_11_DEBUG Set bit[11] to 1	Enables NVG86 ECO. Note: This is applicable to port A only.

ASIC Rev	Step	Register Settings	Function/Comment
RS780 All Revs	28	BIF_NB:PCIE_STRAP_MISC2[2] - PCIEIND:0xC1 STRAP_MSTCPL_TIMEOUT_EN Set bit[2] to 0	Hides and disables the completion timeout method.
RS780 All Revs	29	PCIE_NBCFG_REG6[28] - NBMISCIND:0x36 STRAP_BIF_DEEMPH_BIF_SEL_A STRAP_BIF_DEEMPH_BIF_SEL_B Set bit[28] to 1	Use the bif_core de-emphasis strength by default.
RS780 All Revs	30	BIF_NB:PCIE_CNTL2 -- PCIEIND:0x1C TX_ARB_ROUND_ROBIN_EN set to 1 TX_ARB_SLV_LIMIT set to 4 TX_ARB_MST_LIMIT set to 4	Set TX arbitration algorithm to round robin.
RS780 All Revs	31	PCIEP_HW_DEBUG - PCIEIND_P:0x02 HW_12_DEBUG Set bit[12] to 1	Internal interrupt generation not depending on INT_DIS bit.
RS780 All Revs	32	PCIEP_HW_DEBUG - PCIEIND_P:0x02 HW_13_DEBUG Set bit[13] to 1	Internal pme generation not depending on Native PME bit set.
	33	PCIE_LC_TRAINING_CNTL – PCIEIND_P:0xA1 LC_RESET_ASPM_L1_NAK_TIMER Clear bit [26] to 0	Workaround for Broadcom Network Card bug. For all ports.
	34	PCIE_STRAP_PI - PCIEIND:0xC2 STRAP_PHY_RX_INCAL_FORCE set bit[25] to 1	Work around for L1 efficiency degradation due to PCIE eIDLE PHY glitch For all ports

5.9.13 Autonomous GEN2 Speed Change (CMOS Option – Disabled by Default)

The following 2 options are available:

- PCIE-GFX Port A Autonomous GEN2 (Single/Dual Configuration)
- PCIE-GFX Port B Autonomous GEN2 (Dual Slot Configuration)

Table 5-34 Autonomous GEN2 Speed Change (CMOS Option - Disabled by Default)

ASIC Rev	Step	Register Settings	Function/Comment
RS780 All Revs	1	BIF_NBP:LINK_CNTL2 -- pcieCopcieConfigDev*:0x88 TARGET_LINK_SPEED Set bits[3:0] to 4'h2	Sets TARGET_LINK_SPEED to GEN2.
	2	NBCFG: PCIE_NBCFG_REG4 – NBMISCIND: 0x34 STRAP_BIF_DE_EMPHASIS_SEL (port A) Set bit[5] to 1 NBCFG: PCIE_NBCFG_REGA – NBMISCIND: 0x22 STRAP_BIF_DE_EMPHASIS_SEL (port B) Set bit[6] to 1	Advertises -3.5 dB de-emphasis value in TS1 Data Rate Identifier
	3	BIF_NBP:PCIE_LC_SPEED_CNTL -- PCIEIND_P:0xA4 LC_GEN2_EN_STRAP Set bit[0] to 1	Disables GEN2 capability of the device.
	4	BIF_NBP:PCIEP_STRAP_LC -- PCIEIND_P:0xC0 STRAP_AUTO_RC_SPEED_NEGOTIATION_DIS Set bit [15] to 0	Sets AUTO RC SPEED NEGOTIATION

	5	BIF_NBP:PCIE_LC_SPEED_CNTL -- PCIEIND_P:0xA4 LC_MULT_UPSTREAM_AUTO_SPD_CHNG_EN Set bit [29] to 1	Enables Gen2 Speed Change on any surprised link down
	6	BIF_NBP:PCIE_LC_LINK_WIDTH_CNTL[13]=1'b0·PCIEIND_P:0xA2 LC_UPCONFIGURE_DIS Clear bit[13] to 0.	Clears up gating off Upconfigure Support.

5.9.14 Link Training

Release hold training (by setting the corresponding hold training bit in the table below) and then start the link training procedure outlined in Section 5.7 PCIE Link Training Sequence.

Table 5-35 Link Training

Device	HOLD_TRAINING Bit
Dev 2	PCIE_LINK_CFG[4]
Dev 3	PCIE_LINK_CFG[5]

5.9.15 Power Down Control

In order to save power, inactive lanes and PLLs should be powered down.

5.9.15.1 Inactive Lanes

There are a total of 16 register bits assigned to control the powering down of inactive lanes. The transmitter and the receiver of a lane can be powered down separately; in the case of an inactive lane, both the transmitter and the receiver should be powered down.

Each register bit controls the powering down of 2 lanes; the corresponding register bit should be set to 1 when both lanes are inactive. The register and lane mappings are specified as follows:

- Transmitter: B_PTX_PDNB_FDIS = PCIE_P_PAD_FORCE_DIS – PCIEIND: 0x65, bits[7:0]
- Receiver: B_PRX_PDNB_FDIS = PCIE_P_PAD_FORCE_DIS – PCIEIND: 0x65, bits[15:8]

Table 5-36 Inactive Lanes

Inactive Lanes	B_PRX_PDNB_FDIS								B_PTX_PDNB_FDIS							
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0-1								1								1
2-3							1								1	
4-5						1								1		
6-7					1								1			
8-9				1								1				
10-11			1								1					
12-13		1								1						
14-15	1								1							

5.9.15.2 Inactive PLLs

There are 3 PLLs associated with lanes 0-3 (PLL A), 4-7 (PLL B) and 8-15 (PLL C). In order to achieve maximum power saving, the transmitter and receiver output buffers of the PLL can also be powered down separately. There are 3 different parameters to control each PLL:

- B_PPLL_PDNB_FDIS: Power down the PLL completely, no current consumption except for leakage.
- B_P90_PLL_BUF_PDNB_TX_FDIS: Power down the transmitter output buffers.
- B_P90_PLL_BUF_PDNB_RX_FDIS: Power down the receiver output buffers.

A PLL can be powered down completely (all 3 parameters set to 1) when all the lanes associated with it are inactive.

The receiver of a PLL can be powered down when lanes associated with it are running in DDI mode due to the absence of receiver data stream in DDI mode. (The receiver of a lane is also powered down in DDI mode, which is automatically done by hardware).

The register control is specified as follows:

Note: For ASIC Rev A11 ONLY - When accessing PCIE_NBCFG_REG16 (NBMISCIND:0x2e), the write enable bit must be set for both reads and writes.

- B_PPLL_PDNB_FDIS = PCIE_NBCFG_REG16 – NBMISCIND: 0x2e, bits[6:4]
- B_P90PLL_BUF_PDNB_TX_FDIS = PCIE_NBCFG_REG16 – NBMISCIND: 0x2e, bits[18:16]
- B_P90PLL_BUF_PDNB_RX_FDIS = PCIE_NBCFG_REG16 – NBMISCIND: 0x2e, bits[22:20]

Table 5-37 Inactive PLLs

Inactive Lanes	B_P90PLL_BUF_PDNB_RX_FDIS			B_P90PLL_BUF_PDNB_TX_FDIS			B_PPLL_PDNB_FDIS		
	22	21	20	18	17	16	6	5	4
0-3			1			1			1
4-7		1			1			1	
8-15	1			1			1		

5.9.15.3 Powering Down in PCIE Only Modes

5.9.15.3.1 Finding the Inactive Lanes and PLLs

The following procedure can be used to find out which of the 16 lanes are active.

- Single Slot Configuration

Table 5-38 Single Slot Configuration Settings

ASIC Rev	Step	Register Settings	Function/Comment
RS780 All Revs	1	BIF_NB: PCIE_DEBUG_CNTL – PCIEIND: 0x12 DEBUG_PORT_EN Set bits[7:0] to 8'h1	Select Port A
	2	BIF_NB: PCIE_LC_STATUS2 – PCIEIND: 0x29 LC_TOTAL_INACTIVE_LANES Read back bits[15:0]	Each of the 16 bits of the read back value represents a lane. Bit [0] represents lane 0. Bit [15] represents lane 15. 1=Inactive lane 0=Active lane

- Dual Slot Configuration

Table 5-39 Dual Slot Configuration Settings

ASIC Rev	Step	Register Settings	Function/Comment
RS780 All Revs	1	BIF_NB: PCIE_DEBUG_CNTL – PCIEIND: 0x12 DEBUG_PORT_EN Set bits[7:0] to 8'h1	Select Port A
	2	BIF_NB: PCIE_LC_STATUS2 – PCIEIND: 0x29 LC_TOTAL_INACTIVE_LANES Read back bits[7:0]	Each of the 8 bits of the read back value represents a lane. Bit [0] represents lane 0. Bit [7] represents lane 7. 1=Inactive lane 0=Active lane
	3	BIF_NB: PCIE_DEBUG_CNTL – PCIEIND: 0x12 DEBUG_PORT_EN Set bits[7:0] to 8'h2	Select Port B
	4	BIF_NB: PCIE_LC_STATUS2 – PCIEIND: 0x29 LC_TOTAL_INACTIVE_LANES Read back bits[7:0]	Each of the 8 bits of the read back value represents a lane. Bit [0] represents lane 8. Bit [7] represents lane 15. 1=Inactive lane 0=Active lane

Table 5-40 Inactive Lanes and PLLS

Lanes	0 to 3	4 to 7	8 to 11	12 to 15
1	GFX x16 A			
2	GFX x8 A			
3			GFX x8 A	
4	GFX x8 A		GFX x8 B	
5	GPP x4 A			
6		GPP x4 B		
7			GPP x4 A	
8				GPP x4 B
9	GPP x4 A	GPP x4 B		
10	GPP x4 A		GPP x4 B	
11	GPP x4 A			GPP x4 B
12		GPP x4 B	GPP x4 A	
13			GPP x4 A	GPP x4 B
14	GFX x8 A		GPP x4 B	
15	GFX x8 A			GPP x4 B
16		GPP x4 B	GFX x8 A	
17	GPP x4 A		GFX x8 B	

Table 5-41 (for normal mode) and *Table 5-42* (for reversal mode) specify the inactive lanes (0-15) and inactive PLLs (A, B, C) in each of the supported configurations in *Table 5-40* above. The register settings for the power down can then be worked out based on the mapping in section *5.9.15.1* (“Inactive Lanes”) and in section *5.9.15.2* (“Inactive PLLs”).

Table 5-41 Normal Mode

Lanes	Lane Reversal Disabled			
	x0	x1, x2	x4	x8
1	0-15 A, B, C	2-15 B, C	4-15 B, C	8-15 C
2	Same as case 1			
3	0-15 A, B, C	0-7, 10-15 A, B	0-7, 12-15 A, B	0-7 A, B
4	Port A: 0-7 A, B Port B: 8-15 C	Port A: 2-7 B Port B: 10-15 No inactive PLL	Port A: 4-7 B Port B: 12-15 No inactive PLL	Port A: No inactive lane No inactive PLL Port B: No inactive lane No inactive PLL
5	Same as case 1			N/A
6	0-15 A, B, C	0-3, 6-7, 8-15 C	0-3, 8-15 C	N/A N/A
7	Same as case 3			N/A
8	0-15 A, B, C	0-11, 14-15 A, B	0-11 A, B	N/A N/A
9	Port A: 0-3, 8-11 A, C Port B: 4-7, 12-15 B, C	Port A: 2-3, 8-11 C Port B: 6-7, 12-15 C	Port A: 8-11 C Port B: 12-15 C	N/A
10	Same as case 4			N/A
11	Same as case 15			N/A
12	Same as case 16			N/A

13	Port A: 0-3, 8-11 A Port B: 4-7, 12-15 B C if A&B are x0	Port A: 0-3, 10-11 A Port B: 4-7, 14-15 B	Port A: 0-3 A Port B: 4-7 B	N/A
14	Same as case 4			
15	Port A: 0-7 A, B Port B: 8-15 C	Port A: 2-7 B Port B: 8-11, 14-15 No inactive PLL	Port A: 4-7 B Port B: 8-11 No inactive PLL	Port A: No inactive lane No inactive PLL Port B: N/A
16	Port A: 8-15 C Port B: 0-7 A, B	Port A: 10-15 No inactive PLL Port B: 0-3, 6-7 A	Port A: 12-15 No inactive PLL Port B: 0-3 A	Port A: No inactive lane No inactive PLL Port B: N/A
17	Same as case 4			

Table 5-42 Reversal Mode

Lanes	Lane Reversal Enabled			
	x0	x1, x2	x4	x8
1	0-15 A, B, C	0-13 A, B	0-11 A, B	0-7 A, B
2	Same as case 1			
3	Same as case 1			
4	Port A: 0-7 A, B Port B: 8-15 C	Port A: 0-5 A Port B: 8-13 No inactive PLL	Port A: 0-3 A Port B: 8-11 No inactive PLL	Port A: No inactive lane No inactive PLL Port B: No inactive lane No inactive PLL
5	Same as case 1			N/A
6	Lane reversal is not supported in this configuration			
7	Lane reversal is not supported in this configuration			
8	Lane reversal is not supported in this configuration			
9	Lane reversal is not supported in this configuration			
10	Lane reversal is not supported in this configuration			
11	Lane reversal is not supported in this configuration			
12	Lane reversal is not supported in this configuration			
13	Lane reversal is not supported in this configuration			
14	Same as case 4			
15	Lane reversal is not supported in this configuration			
16	Lane reversal is not supported in this configuration			
17	Same as case 4			

5.9.15.3.2 Selecting TXCLK Source For Core Logic

The programming in this step must be executed *before* the register writes for the power down. The PCIE core logic requires a TXCLK which comes from either PLL A or PLL C; therefore, the clock muxes must be programmed to select the clock source from a running PLL.

- PLL A is to be powered down

Table 5-43 Selecting TXCLK Source For Core Logic

ASIC Rev	Step	Register Settings	Function/Comment
RS780 All Revs	1	PCIE_PDNB_CNTL – NBMISCIND: 0x7 GFX_TXCLK_SEL Set bit[16] to 1	Selects PLL C to be the source clock for TXCLK_PERM
	2	PCIE_PDNB_CNTL – NBMISCIND: 0x7 GFX_TXCLK_SND_RCV_0_SEL GFX_TXCLK_SND_RCV_1_SEL GFX_TXCLK_SND_RCV_2_SEL GFX_TXCLK_SND_RCV_3_SEL Set bit[12] = 1 Set bit[13] = 1 Set bit[14] = 1 Set bit[15] = 1	Selects PLL C to be the source clock for TXCLK SND and RCV
	3	PCIE_PDNB_CNTL – NBMISCIND: 0x7 IO_TXCLK_A_SEL IO_TXCLK_B_SEL IO_TXCLK_C_SEL Set bits[21:20] = 2'b10 Set bits[23:22] = 2'b10 Set bits[25:24] = 2'b10	Selects PLL C to be the source clock for B_PTX_DATA_CLK

- PLL C is to be powered down

  No programming is required in this case since the default is selecting PLL A.

- PLL A and PLL C are both powered down

  This case should never happen when any PCIE link is trained. Even though only Lanes 4-7 are used (PLL B is associated with these physical lanes), either PLL A or PLL C must be on to provide a clock for the core logic.

5.9.15.3.3 Turning Off REFCLK Receiver Buffers

If no link is trained, then the REFCLK receiver buffer should be turned off to save power.

Table 5-44 Turning Off REFCLK Receiver Buffers Settings

ASIC Rev	Step	Register Settings	Function/Comment
RS780 All Revs	1	PCIE_NBCFG_REG8 – NBMISCIND:0x38 B_PCLK_BIDIR_RX_EN Set bit[28] to 0	Disables GFX REFCLK receiver to receive the REFCLK from an external source

5.9.15.3.4 Turning Off Electrical Idle Detectors

The electrical idle detectors should be powered off when:

- No compliance card is detected.
- No GFX link is trained (TMDS/HDMI/DP modes are irrelevant as the electrical ide detectors are unused in those modes).

Table 5-45 Turning Off Electrical Idle Detectors Settings

ASIC Rev	Step	Register Settings	Function/Comment
RS780 All Revs	1	PCIE_NBCFG_REG4 – NBMISCIND:0x34 B_PG2RX_IDLEDET_EN Set bit[30] to 0	Disables the electrical idle detectors for all 16 GFX lanes

5.9.15.4 Powering Down in DDI Only Modes

Table 5-46 Powering Down In DDI Only Modes

Lanes	0 to 3	4 to 7	8 to 11	12 to 15
1	DDI_SL			
2		DDI_SL		
3			DDI_SL	
4				DDI_SL
5	DDI_SL	DDI_SL		
6	DDI_DL			
7	DDI_SL		DDI_SL	
8	DDI_SL			DDI_SL
9		DDI_SL	DDI_SL	
10		DDI_SL		DDI_SL
11			DDI_DL	

The register settings for power down in DDI only modes can be worked out based on the information found in *Table 5-47*:

Table 5-47 Powering Down in DDI Only Modes Register Setting Information

Case	Inactive Lanes	Inactive PLLs (Full Power Down)	PLLs with Inactive RX Buffers
1	4-15	B, C	A
2	0-3, 8-15	A, C	B
3	0-7, 12-15	A, B	C
4	0-11	A, B	C
5	8-15	C	A, B
6	8-15	C	A, B
7	4-7, 12-15	B	A, C
8	4-11	B	A, C
9	0-3, 12-15	A	B, C
10	0-3, 8-11	A	B, C
11	0-7	A, B	C

5.9.15.5 Powering Down in Combined PCIE and DDI Modes

If both PCIE and DDI devices are present, the first step is to work out the power down register settings individually assuming PCIE only and DDI only modes. Then the 2 register settings should be “ANDed” together for a final setting that would only power down the lanes and PLLs which are unused in both PCIE and DDI modes.

The programming sequence in the subsections from this point onward should be executed after the VBIOS post is completed.

5.9.16 Software Initiated Speed Change to GEN2 (CMOS Option – Disabled by Default)

The following 2 CMOS options should be available:

- PCIE-GFX Port A Software GEN2 (Single/Dual Configuration)
- PCIE-GFX Port B Software GEN2 (Dual Configuration)

Table 5-48 Software Initiated Speed Change to GEN2 (CMOS Option - Disabled by Default)

ASIC Rev	Step	Register Settings	Function/Comment
RS780 All Revs	1	Read back bits [3:0] BIF_NBP:LINK_CAP [3:0] · pcieConfigDev*:0x64 LINK_SPEED If read back values of [3:0] is 1 -> exit the sequence as the RC does not support Gen2 If read back value of [3:0] is 2 -> go to Step 2	Reads back Maximum Link speed of the given PCI Express Link advertised from the RC.
	2	In the Configuration Space of the EP device, read back bits [3:0]: BIF_NBP:LINK_CAP [3:0] · pcieConfigDev*:0x64 LINK_SPEED If read back values of [3:0] is 1 -> exit the sequence as the EP does not support Gen2 If read back value of [3:0] is 2 -> go to Step 3	Reads back Maximum Link speed of the given PCI Express Link advertised from the EP.
	3	Read back bit [24] of the: BIF_NBP:PCIE_LC_SPEED_CNTL[24] · PCIEIND_P:0xA4 LC_OTHER_SIDE_SUPPORTS_GEN2 If read back value of bit [24] is 1 -> go to Step 5 Otherwise -> go to Step 4	Checks if the other side of the link supports Gen2.
	4a	BIF_NBP:PCIE_LC_SPEED_CNTL[0] – PCIEIND_P:0xA4 LC_GEN2_EN_STRAP Set bit[0] to 1 For AMD devices only, in the Configuration Space of the EP device: BIF_NBP:PCIE_LC_SPEED_CNTL[0] -- PCIEIND_P:0xA4 LC_GEN2_EN_STRAP Set bit[0] to 1	PCIE-GFX Device * advertises that it supports Gen2 Only for AMD Devices.

ASIC Rev	Step	Register Settings	Function/Comment
	4b	GPP Core: NBCFG: PCIE_NBCFG_REG9 – NBMISCIND: 0x39 STRAP_BIF_DE_EMPHASIS_SEL (port A) Set bit[31] to 1 NBCFG: PCIE_NBCFG_REGA – NBMISCIND: 0x22 STRAP_BIF_DE_EMPHASIS_SEL (port B) Set bit[5] to 1 GPPSB Core: NBCFG: PCIE_NBCFG_REG4 – NBMISCIND: 0x34 STRAP_BIF_DE_EMPHASIS_SEL (port B) Set bit[31] to 1 NBCFG: PCIE_NBCFG_REG7 – NBMISCIND: 0x37 STRAP_BIF_DE_EMPHASIS_SEL (port C) Set bit[5] to 1 NBCFG: PCIE_NBCFG_REG7 – NBMISCIND: 0x37 STRAP_BIF_DE_EMPHASIS_SEL (port D) Set bit[6] to 1 NBCFG: PCIE_NBCFG_REG7 – NBMISCIND: 0x37 STRAP_BIF_DE_EMPHASIS_SEL (port E) Set bit[7] to 1	Advertises -3.5 dB de-emphasis value in TS1 Data Rate Identifier
	4c	BIF_NBP:PCIE_LC_LINK_WIDTH_CNTL – PCIEIND_P:0xA2 LC_UPCONFIGURE_DIS Clear bit[13] to 0.	Clears up gating off Upconfigure Support.
	5	BIF_NBP:LINK_CNTL2 -- pcieCopcieConfigDev*:0x88 TARGET_LINK_SPEED Set bits[3:0] to 4'h2 In the Configuration Space of the EP device: BIF_NBP:LINK_CNTL2[3:0] -- pcieCopcieConfigDev*:0x88 TARGET_LINK_SPEED Set bits[3:0] to 4'h2	Sets PCIE-GFX/PCIE-GFX2 Port * link speed to be Gen2
	6	BIF_NBP:PCIE_LC_SPEED_CNTL -- PCIEIND_P:0xA4 LC_GO_TO_RECOVERY Set bit [18] to 1.	Initiates Recovery
	7	Pool back [5:0] of: BIF_NBP:PCIE_LC_STATE0[5:0] · PCIEIND_P:0xA5 LC_CURRENT_STATE Until 5'h10.	Waits until link retrains to L0.
	8	Read back bit [24] of the: BIF_NBP:PCIE_LC_SPEED_CNTL[24] · PCIEIND_P:0xA4 LC_OTHER_SIDE_SUPPORTS_GEN2 If read back value of bit [24] is 1 -> go to next step Otherwise -> skip the sequence	Checks if the other side of the link supports Gen2.

ASIC Rev	Step	Register Settings	Function/Comment
	9	BIF_NBP:PCIE_LC_SPEED_CNTL -- PCIEIND_P:0xA4 LC_INITIATE_LINK_SPEED_CHANGE Set bit[7] to 1	Initiates Link Speed Change

5.9.17 Active State Power Management (ASPM)

5.9.17.1 ASPM L1 (CMOS Option - Enabled by Default)

The SBIOS should first check if the card is an AMD graphics card (it must check the AMD vendor ID, and check that it is a graphics card) and that the ASPM L1 CMOS option is set. Once this has been checked, the below programming sequence should be performed.

The following 2 CMOS options should be available:

- PCIE-GFX Port A ASPM L1 (Single/Dual Configuration)
- PCIE-GFX Port B ASPM L1 (Dual Configuration)

Table 5-49 ASPM L1 (CMOS Option - Enabled by Default)

ASIC Rev	Step	Register Settings	Function/Comment
RS780 All Revs	1	BIF_NBP:LINK_CNTL[1]=1'b1· pcieConfigDev*:0x68 PM_CONTROL Set bit [1] to 1	Enables L1 in North Bridge
	2	In the graphics card(s) register space: BIF:PCIE_LC_CNTL[15:12]=4'h6· PCIEIND_P :0xA0 LC_L1_INACTIVITY Set bits[15:12] to 4'h6	Sets the L1 inactivity timer in the endpoint to be 10000 TXCLKs Gen1: 10 000 x 4ns = 40us Gen2: 10 000 x 2ns = 20us
	3	In the graphics card(s) register space: BIF_NBP:LINK_CNTL[1]=1'b1· pcieConfigDev*:0x68 PM_CONTROL Set bit [1] to 1	Enables L1 in the graphics card(s)

L1 should be DISABLED if any of the following device IDs is detected:

- [01D0] G72
- [01D1] G72
- [01D2] G72
- [01D3] G72
- [01D5] G72
- [01D7] GeForce Go7300
- [01D8] GeForce Go7400
- [01DC] G72GLm
- [01DE] G72GL
- [01DF] G72

5.9.17.2 Powering Off PLL During L1/L23 (CMOS Option – Disabled by Default)

This feature requires ASPM L1 to be enabled for all the ports and the PLL will only be powered off when all the ports are in L1.

Table 5-50 Powering Off PLL During L1/L23 (CMOS Option - Disabled by Default)

ASIC Rev	Step	Register Settings	Function/Comment
RS780 All Revs	1	PCIE_P_CNTL – PCIEIND: 0x40 P_PWRDN_EN Set bit[0] to 1	Enables powering down transmitter and receiver pads along with PLL macros
	2	PCIE_P_CNTL – PCIEIND: 0x40 P_PLL_PWRDN_IN_L1L23 Set bit[3] to 1	Enables PLL power down during L1
	3	PCIE_P_CNTL – PCIEIND: 0x40 P_PLL_BUF_PDNB Set bit[4] to 0	Active-low signal to enable PLL buffers to be powered down during L1
	4	PCIE_P_CNTL – PCIEIND: 0x40 P_PLL_PDNB Set bit[9] to 0	Active-low signal to enable PLL to be powered down during L1
	5	PCIE_P_CNTL – PCIEIND: 0x40 P_ALLOW_PRX_FRONTEND_SHUTOFF Set bit[12] to 1	Allows RX front end to be shutoff during L1 when PLL power down is enabled
	6	BIF_NB:PCIE_HW_DEBUG - PCIEIND:0x2 HW_08_DEBUG Set bit[8] to 1	PLL_OFF_INSTABILITY_FIX_ENABLE

5.9.17.3 L0s (CMOS Option – Disabled by Default)

The transmitter and the receiver can go into L0s independently. The following 4 CMOS options should be available:

- PCIE-GFX Port A Transmitter L0s (Single/Dual Configuration)
- PCIE-GFX Port A Receiver L0s (Single/Dual Configuration)
- PCIE-GFX Port B Transmitter L0s (Dual Configuration)
- PCIE-GFX Port B Receiver L0s (Dual Configuration)

To enable Transmitter L0s use the following information in *Table 5-51*:

Table 5-51 Transmitter L0s Settings

ASIC Rev	Step	Register Settings	Function/Comment
RS780 All Revs	1	PCIE_LC_TRAINING_CNTL –PCIEIND_P:0xA1 LC_DONT_GO_TO_L0S_IF_L1_ARMED Set bit[11] to 1	Disables L0s entry if en route to L1
	2	PCIE_LC_CNTL – PCIEIND_P: 0xA0 LC_L0S_INACTIVITY Set bits[11:8] to 4'h8	Sets L0s inactivity timer to 10us
	3	LINK_CNTL – pcieConfigDev*: 0x68 PM_CONTROL Set bit[0] to 1	Enables L0s

To enable Receiver L0s (registers are programmed on the end point's register space) use the following information in *Table 5-52*:

Table 5-52 Receiver L0s Settings

ASIC Rev	Step	Register Settings	Function/Comment
RS780 All Revs	1	Read bit[10] of the following register: BIF:LINK_CAP[10]· pcieConfigDev2:0x64 LINK_CAP If read back value of bit [10] is 1 -> go to next step. Otherwise, skip this sequence.	Bit[10] = 1 means the device supports L0s ->go to Step 2. Bit[10] = 0 means the device does not support L0s. Stop
	2	BIF:LINK_CNTL[1:0]=4'h8· pcieConfigDev*:0xA0 LC_L0S_INACTIVITY Set bit[11:8] to 4'h8	This step is only applicable if the device is an AMD graphics card Set L0s inactivity timer to 1000 TXCLKs. Gen1: 1000x4ns=4us Gen2: 1000x2ns=2us
	3	BIF:LINK_CNTL[0]=1'b1· pcieConfigDev*:0x68 PM_CONTROL Set bit [0] to 1.	L0s should not be enabled for AMD GFX cards earlier than R5xxx. Only the R5xxx and higher generations support L0s properly.

5.9.18 Clock Gating

5.9.18.1 TXCLK Gating (CMOS Option – Disabled by Default)

Table 5-53 TXCLK Gating (CMOS Option - Disabled by Default)

ASIC Rev	Step	Register Settings	Function/Comment
RS780 All Revs	1	PCIE_CONFIG_CNTL – PCIEIND: 0x11 DYN_CLK_LATENCY Set bits[3:0] to 4'b1100	Sets dynamic clock latency in bif_core
	2	PCIE_PDNB_CNTL – NBMISCIND : 0x7 ENABLE_CLKGATE_GFX_TXCLK ENABLE_CLKGATE_GFX_TXCLK_L0S ENABLE_CLKGATE_GFX_TXCLK_SND_RCV Set bit[2:0] to 3'b111	Enables clock gating on all TXCLK branches
	3	PCIE_P_CNTL – PCIEIND: 0x40 P_TXCLK_SND_PWRDN Set bit[5] to 1	Allows TXCLK_SND to be powered down
	4	PCIE_P_CNTL – PCIEIND: 0x40 P_TXCLK_RCV_PWRDN Set bit[6] to 1	Allows TXCLK_RCV to be powered down

5.9.18.2 LCLK Gating (CMOS Option – Disabled by Default)

Table 5-54 LCLK Gating (CMOS Option - Disabled by Default)

ASIC Rev	Step	Register Settings	Function/Comment
RS780 All Revs	1	CLKCFG: 0x94 CLKGATE_DISABLE Set bit[16] to 0	Enables LCLK gating for the GFX core

5.9.18.3 Shutting Off TXCLK Permanently (CMOS Option – Disabled by Default)

The TXCLK can be shut off permanently to save power when the no lane is active for PCIE purpose. When this feature is enabled, all register reads/writes to the core will be invalid.

Table 5-55 Shutting Off TXCLK Permanently (CMOS Option - Disabled by Default)

ASIC Rev	Step	Register Settings	Function/Comment
RS780 All Revs	1	PCIE_PDNB_CNTL – NBMISCIND : 0x7 GFX_PERM2_TXCLK_STOP Set bit[3] to 1	Shuts off TXCLK permanently in the GFX core

5.10 PCIE-GPPSB and PCIE-GPP Cores Initialization

Note: The initialization sequence should be executed in the same order as the sections are organized.

5.10.1 REFCLK Options

- External Clock Mode (Default Mode): an external clock chip is used to drive the REFCLK on the GPP slots

Table 5-56 External Clock Mode (Default Mode)

ASIC Rev	Register Settings	Function/Comment
RS780 All Revs	PCIE_NBCFG_REG8 – NBMISCIND:0x38 B_PCLK_TX_EN Set bit[26] to 0	Disables the GPP REFCLK transmitter so that the GPP slots can be driven by an external source.

- Internal Clock Mode: SB REFCLK is used to drive the GPP slots

Table 5-57 Internal Clock Mode

ASIC Rev	Register Settings	Function/Comment
RS780 All Revs	PCIE_NBCFG_REG8 – NBMISCIND:0x38 B_PCLK_TX_EN Set bit[26] to 1	Enables the GPP REFCLK transmitter so that the GPP slots can be driven by an external source.

5.10.2 Lane Reversal (CMOS Option – Disabled by Default)

There should be 6 CMOS options for each GPP port:

- PCIE-GPPSB Port B Lane Reversal
- PCIE-GPPSB Port C Lane Reversal
- PCIE-GPPSB Port D Lane Reversal
- PCIE-GPPSB Port E Lane Reversal
- PCIE-GPP Port A Lane Reversal
- PCIE-GPP Port B Lane Reversal

Table 5-58 GPPSB Core

ASIC Rev	Step	Register Settings	Function/Comment
RS780 All Revs	1	StrapsOutputMux_6 – NBMISCIND:0x66 STRAP_BIF_all_valid (active low) Set bit [31] to 1	De-asserts STRAP_BIF_all_valid for PCIE-GPPSB core.
	2	PCIE_NBCFG_REG3 - NBMISCIND:0x33 STRAP_BIF_REVERSE_LANES_GPPSB_B Set bit[4] to 1	Enables lane reversal for GPPSB Port B
	3	PCIE_NBCFG_REG3 - NBMISCIND:0x33 STRAP_BIF_REVERSE_LANES_GPPSB_C Set bit[5] to 1	Enables lane reversal for GPPSB Port C
	4	PCIE_NBCFG_REG3 - NBMISCIND:0x33 STRAP_BIF_REVERSE_LANES_GPPSB_D Set bit[6] to 1	Enables lane reversal for GPPSB Port D
	5	PCIE_NBCFG_REG3 - NBMISCIND:0x33 STRAP_BIF_REVERSE_LANES_GPPSB_E Set bit[7] to 1	Enables lane reversal for GPPSB Port E
	6	StrapsOutputMux_6 – NBMISCIND:0x66 STRAP_BIF_all_valid (active low) Set bit [31] to 0	Asserts STRAP_BIF_all_valid for PCIE-GPPSB core.

Table 5-59 GPP Core

ASIC Rev	Step	Register Settings	Function/Comment
RS780 All Revs	1	PCIE_NBCFG_REGA – NBMISCIND:0x22 STRAP_BIF_all_valid (active low) Set bit [14] to 1	De-asserts STRAP_BIF_all_valid for PCIE-GPP core.
	2	PCIE_NBCFG_REGA – NBMISCIND:0x22 STRAP_BIF_REVERSE_LANES_GPP_A Set bit[31] to 1	Enables lane reversal for GPP Port A
	3	PCIE_NBCFG_REGB – NBMISCIND:0x23 STRAP_BIF_REVERSE_LANES_GPP_B Set bit[0] to 1	Enables lane reversal for GPP Port B
	4	PCIE_NBCFG_REG6 – NBMISCIND:0x22 STRAP_BIF_all_valid (active low) Set bit [14] to 0	Asserts STRAP_BIF_all_valid for PCIE-GPP core.

5.10.3 Transmitter Drive Strength (CMOS Option – Disabled 22mA)

Table 5-60 GPPSB Core:

ASIC Rev	Step	Register Settings	Function/Comment
RS780 All Revs	1	StrapsOutputMux_8 – NBMISCIND:0x68 B_P90TX_DRV_STR[1:0] for GPPSB Set bits[6:5] according to CMOS option	Possible CMOS options: 2'b00: 18mA nominal 2'b01: 20mA nominal 2'b10: 22mA nominal 2'b11: 24mA nominal

Table 5-61 GPP Core:

ASIC Rev	Step	Register Settings	Function/Comment
RS780 All Revs	1	PCIE_NBCFG_REGC – NBMISCIND:0x24 B_P90TX_DRV_STR[1:0] for GPP Set bits[26:25] according to CMOS option	Possible CMOS options: 2'b00: 18mA nominal 2'b01: 20mA nominal 2'b10: 22mA nominal 2'b11: 24mA nominal

5.10.4 Reset PCIE-GPP Slot

Table 5-62 Reset PCIE-GPP Slot

ASIC Rev	Step	Register Settings	Function/Comment
RS780 All Revs	1	Desktop reference board: AZ_DOCK_RST#/GPM8# Mobile reference board: LAN_RST#/GPIO13	Programs the GPIO in SB700 to reset the PCIE-GPP slots.
	2	Use the Delay Training CMOS Option described in the next section	Programs delay link training timer.

5.10.5 GEN1 Software Compliance (CMOS Option – Disabled by Default)

There should be 6 CMOS options for each GPP port:

- PCIE-GPPSB Port B Lane Reversal
- PCIE-GPPSB Port C Lane Reversal
- PCIE-GPPSB Port D Lane Reversal
- PCIE-GPPSB Port E Lane Reversal
- PCIE-GPP Port A Lane Reversal
- PCIE-GPP Port B Lane Reversal

Table 5-63 GEN1 Software Compliance (CMOS Option - Disabled by Default)

ASIC Rev	Register Settings	Function/Comment
RS780 All Revs	BIF_NBP:PCIEP_STRAP_LC -- PCIEIND_P:0xC0 STRAP_FORCE_COMPLIANCE Set bit[13] to 1	Forces transmitter to output compliance pattern at Gen1 rate.

5.10.6 GEN2 Software Compliance (CMOS Option – Disabled by Default)

There should be 6 CMOS options for each GPP port:

- PCIE-GPPSB Port B Lane Reversal
- PCIE-GPPSB Port C Lane Reversal
- PCIE-GPPSB Port D Lane Reversal
- PCIE-GPPSB Port E Lane Reversal
- PCIE-GPP Port A Lane Reversal
- PCIE-GPP Port B Lane Reversal

Table 5-64 GEN 2 Software Compliance (CMOS Option - Disabled by Default)

ASIC Rev	Step	Register Settings	Function/Comment
RS780 All Revs	1	BIF_NBP:PCIE_LC_SPEED_CNTL -- PCIEIND_P:0xA4 LC_GEN2_EN_STRAP Set bit[0] to 1	Enables GEN2 capability of the device.
	2	GPP Core: NBCFG: PCIE_NBCFG_REG9 – NBMISCIND: 0x39 STRAP_BIF_DE_EMPHASIS_SEL (port A) Set bit[31] to 1 NBCFG: PCIE_NBCFG_REGA – NBMISCIND: 0x22 STRAP_BIF_DE_EMPHASIS_SEL (port B) Set bit[5] to 1 GPPSB Core: NBCFG: PCIE_NBCFG_REG4 – NBMISCIND: 0x34 STRAP_BIF_DE_EMPHASIS_SEL (port B) Set bit[31] to 1 NBCFG: PCIE_NBCFG_REG7 – NBMISCIND: 0x37 STRAP_BIF_DE_EMPHASIS_SEL (port C) Set bit[5] to 1 NBCFG: PCIE_NBCFG_REG7 – NBMISCIND: 0x37 STRAP_BIF_DE_EMPHASIS_SEL (port D) Set bit[6] to 1 NBCFG: PCIE_NBCFG_REG7 – NBMISCIND: 0x37 STRAP_BIF_DE_EMPHASIS_SEL (port E) Set bit[7] to 1	Advertises -3.5 dB de-emphasis value in TS1 Data Rate Identifier
	3	BIF_NBP:LINK_CNTL2 -- pcieConfigDev*:0x88 TARGET_LINK_SPEED Set bits[3:0] to 4'h2	Advertises the link speed to be Gen2.
	4	BIF_NBP:LINK_CNTL2 -- pcieConfigDev*:0x88 ENTER_COMPLIANCE Set bit[4] to 1	Forces transmitter to output compliance pattern at Gen2 rate.

5.10.7 De-Emphasis Strength -3.5dB/-6dB in GEN2 (CMOS Option - Disabled by Default)

Table 5-65 De-Emphasis Strength -3.5dB/-6dB in GEN2 Settings

ASIC Rev	Step	Register Settings	Function/Comment
RS780 All Revs	1	GPP Core: NBCFG: PCIE_NBCFG_REG9 – NBMISCIND: 0x39 STRAP_BIF_DE_EMPHASIS_SEL (port A) Set bit[31] to 0 NBCFG: PCIE_NBCFG_REGA – NBMISCIND: 0x22 STRAP_BIF_DE_EMPHASIS_SEL (port B) Set bit[5] to 0 GPPSB Core: NBCFG: PCIE_NBCFG_REG4 – NBMISCIND: 0x34 STRAP_BIF_DE_EMPHASIS_SEL (port B) Set bit[31] to 0 NBCFG: PCIE_NBCFG_REG7 – NBMISCIND: 0x37 STRAP_BIF_DE_EMPHASIS_SEL (port C) Set bit[5] to 0 NBCFG: PCIE_NBCFG_REG7 – NBMISCIND: 0x37 STRAP_BIF_DE_EMPHASIS_SEL (port D) Set bit[6] to 0 NBCFG: PCIE_NBCFG_REG7 – NBMISCIND: 0x37 STRAP_BIF_DE_EMPHASIS_SEL (port E) Set bit[7] to 0	Advertises -6 dB de-emphasis value in TS1 Data Rate Identifier Default is -3.5dB.

5.10.8 Core Initialization

Table 5-66 Core Initialization

ASIC Rev	Step	Register Settings	Function/Comment
RS780 All Revs	1	BIF_NBP:PCIE_RX_CNTL -- PCIEIND_P:0x70 RX_RCB_CPL_TIMEOUT Set bits[18:16] to 3'h4	Sets timeout to 100ms/4 = 25ms
	2	BIF_NBP:PCIE_RX_CNTL -- PCIEIND_P:0x70 RX_RCB_CPL_TIMEOUT_MODE Set bit[19] to 1	RCB Cpl timeout on link down (to shorten enumeration time).
	3	BIF_NB:PCIE_CI_CNTL -- PCIEIND:0x20 CI_SLV_ORDERING_DIS Clear bit[8] to 0	Enables slave ordering logic.
	4	BIF_NB:PCIE_CNTL -- PCIEIND:0x10 RX_SB_ADJ_PAYLOAD_SIZE Set bits[12:10] to 3'b100	Sets DMA payload size to 64 bytes.
	5	BIF_NBP:PCIEP_PORT_CNTL -- PCIEIND_P :0x10 SLV_PORT_REQ_EN Clear bit[0] to 0 EXCEPT FOR SB (DEVICE 8)	Blocks DMA traffic during C3 state except for SB (device 8).
	6	BIF_NB:PCIE_CI_CNTL -- PCIEIND:0x20 CI_RC_ORDERING_DIS Set bit[9] to 1	Disables RC ordering logic so that Picard can return a dummy master completion without receiving an ACK.
	7	BIF_NB:PCIE_HW_DEBUG -- PCIEIND:0x2 HW_00_DEBUG (REGS_DLP_IGNORE_IN_L1_EN) Set bit [0] to 1	Ignores DLLPs during L1 so that txclk can be turned off .
	8	BIF_NBP:PCIE_LC_TRAINING_CNTL -- PCIEIND_P:0xA1 LC_DONT_GO_TO_L0S_IF_L1_ARMED Set bit [11] to 1	Prevents LC to go from L0 to Rcv_L0s if L1 is armed.
	9	BIF_NBP:PCIE_LC_CNTL2 -- PCIEIND_P:0xB1 LC_WAIT_FOR_LANES_IN_LW_NEG Set bit [23] to 1	Sets the timer in Config state from 20us to 1us for short and up/down reconfiguration.
	10	BIF_NBP:PCIE_LC_CNTL2 -- PCIEIND_P:0xB1 LC_DEASSERT_RX_EN_IN_L0s Set bit[19] to 1	De-asserts RX_EN in L0s.
	11	BIF_NBP:PCIE_LC_CNTL2 -- PCIEIND_P:0xB1 LC_ENABLE_RX_CR_EN_DEASSERTION Set bit [28] to 1	Enables de-assertion of PG2RX_CR_EN to lock clock recovery parameter when lane is in electrical idle in L0s.
	12	GPPSB Core: NBCFG: StrapsOutputMux_7 -- NBMISCIND:0x67 B_P90RX_INCAL_FORCE Set bit[14] to 1 GPP Core: NBCFG: PCIE_NBCFG_REGC – NBMISCIND:0x24 B_P90RX_INCAL_FORCE Set bit[29] to 1	Turns off offset calibration.

	13	GPPSB Core: NBCFG: StrapsOutputMux_7 – NBMISCIND:0x67 B_P90RX_CLKG_EN Set bit[26] to 1 GPP Core: NBCFG: PCIE_NBCFG_REGC – NBMISCIND:0x24 B_P90RX_CLKG_EN Set bit[28] to 1	Enables Rx Clock gating in CDR
	14	BIF_NBP:PCIE_LC_CNTL -- PCIEIND_P:0xA0 LC_16X_CLEAR_TX_PIPE Set bits[7:4] to4'h3	Sets number of TX Clocks to drain TX Pipe to 3.
	16	BIF_NB:PCIE_P_CNTL -- PCIEIND:0x40 P_ELEC_IDLE_MODE Set bits [15:14] to 2'b10	Lets PI use Electrical Idle from PHY when turning off PLL in L1 at Gen2 speed instead Inferred Electrical Idle. Note: LC still uses Inferred Electrical Idle.
	17	BIF_NBP:PCIE_LC_CNTL2 -- PCIEIND_P:0xB1 LC_BLOCK_EL_IDLE_IN_L0 Set bit[20] to 1	Prevents the Electrical Idle from causing a transition from Rcv_L0 to Rcv_L0s.
	18	BIF_NBP:PCIE_LC_TRAINING_CNTL -- PCIEIND_P:0xA1 LC_DONT_GO_TO_L0S_IF_L1_ARMED Set bit [11] to 1.	Prevents the LTSSM from going to Rcv_L0s if it has already acknowledged a request to go to L1.
	19	BIF_NB:PCIE_P_CNTL -- PCIEIND:0x40 RXP_REALIGN_ON_EACH_TSX_OR_SKP Set bit[28] to 0	LDSK only taking deskew on deskewing error detect
	20	BIF_NB:PCIE_STRAP_PI -- PCIEIND:0xC2 STRAP_LDSK_X1_BYPASS Set bit [14] to 1	Bypasses lane de-skew logic if in x1
	21	GPPSB Core: NBCFG: StrapsOutputMux_A – NBMISCIND:0x6A B_PG2PLL_IDLEDET_TH[1:0] Set bits [23:22] to 2'b10 GPP Core: NBCFG: PCIE_NBCFG_REGC – NBMISCIND:0x24 B_PG2PLL_IDLEDET_TH[1:0] Set bits [17:16] to 2'b10	Sets Electrical Idle Threshold

	22	GPP Core: NBCFG: PCIE_NBCFG_REG9 – NBMISCIND: 0x39 STRAP_BIF_DE_EMPHASIS_SEL (port A) Set bit[31] to 0 NBCFG: PCIE_NBCFG_REGA – NBMISCIND: 0x22 STRAP_BIF_DE_EMPHASIS_SEL (port B) Set bit[5] to 0 GPPSB Core: NBCFG: PCIE_NBCFG_REG4 – NBMISCIND: 0x34 STRAP_BIF_DE_EMPHASIS_SEL (port B) Set bit[31] to 0 NBCFG: PCIE_NBCFG_REG7 – NBMISCIND: 0x37 STRAP_BIF_DE_EMPHASIS_SEL (port C) Set bit[5] to 0 NBCFG: PCIE_NBCFG_REG7 – NBMISCIND: 0x37 STRAP_BIF_DE_EMPHASIS_SEL (port D) Set bit[6] to 0 NBCFG: PCIE_NBCFG_REG7 – NBMISCIND: 0x37 STRAP_BIF_DE_EMPHASIS_SEL (port E) Set bit[7] to 0	Only if the CMOS option in section *5.10.7* is enabled.
	23	BIF_NBP:PCIE_LC_SPEED_CNTL – PCIEIND_P:0xA4 LC_GEN2_EN_STRAP Set bit[0] to 0	Enables GEN2 capability of the device.
	24	BIF_NBP:PCIE_LC_LINK_WIDTH_CNTL[13]=1'b1·PCIEIND_P:0xA2 LC_UPCONFIGURE_DIS Set bit[13] to 1.	Disables advertising Upconfigure Support.
	25	NBCFG: PCIE_NBCFG_REGA – NBMISCIND: 0x22 STRAP_BIF_DSN_EN (GPP) Set bit[3] to 0	
	26	NBCFG: trapsOutputMux_8 – NBMISCIND:0x68 STRAP_BIF_DSN_EN (GPPSB) Set bit[19] to 0	
	27	BIF_NBP:PCIE_STRAP_MISC2[0] – PCIEIND:0xC1 STRAP_LINK_BW_NOTIFICATION_CAP_EN Set bit[0] to 1	This capability is required since links that are wider than x1 and/or multiple link speed are supported.
	28	BIF_NB:PCIE_STRAP_MISC2[2] - PCIEIND:0xC1 STRAP_MSTCPL_TIMEOUT_EN Set bit[2] to 0	Hides and disables the completion timeout method

	29	GPPSB: StrapsOutputMux_7 - NBMISCIND:0x67 STRAP_BIF_DEEMPH_BIF_SEL_A Set bit[10] to 1 PCIE_NBCFG_REG6 - NBMISCIND:0x36 STRAP_BIF_DEEMPH_SEL_B, C, D, E Set bit[29] to 1 GPP: PCIE_NBCFG_REG9 - NBMISCIND: 0x39 STRAP_BIF_DEEMPH_BIF_SEL_A, B Set bit[30] to 1	Uses the bif_core de-emphasis strength by default
	30	BIF_NB:PCIE_CNTL2 -- PCIEIND:0x1C TX_ARB_ROUND_ROBIN_EN set to 1 TX_ARB_SLV_LIMIT set to 4 TX_ARB_MST_LIMIT set to 4	Set TX arbitration algorithm to round robin
	31	PCIEP_HW_DEBUG - PCIEIND_P:0x02 HW_12_DEBUG Set bit[12] to 1	Internal interrupt generation not dependent on INT_DIS bit
	32	PCIEP_HW_DEBUG - PCIEIND_P:0x02 HW_13_DEBUG Set bit[13] to 1	Internal PME generation not dependent on Native PME bit
	33	BIF_NBP:PCIE_LC_CNTL[23]=1'b1· PCIEIND_P:0xA0 LC_L1_IMMEDIATE_ACK For NB-SB link (device 8) only. set bit [23] to 1	Always ACK an ASPM L1 entry DLLP to workaround Credit Control issue on PM_NAK message of SB700 and SB800.
	34	PCIE_LC_TRAINING_CNTL – PCIEIND_P:0xA1 LC_RESET_ASPM_L1_NAK_TIMER Clear bit [26] to 0	Workaround for Broadcom Network Card bug. For all ports.
	35	PCIE_STRAP_PI - PCIEIND:0xC2 STRAP_PHY_RX_INCAL_FORCE set bit[25] to 1	Work around for L1 efficiency degradation due to PCIE eIDLE PHY glitch For all ports

5.10.9 Device Remapping

The SBIOS should enable device remapping by default, see section *5.5.1*.

5.10.10 Dynamic Slave CPL Buffer Allocation (CMOS Option – Enabled by Default)

This feature is only required for the PCIE-GPPSB and PCIE GPP cores.

Table 5-67 Dynamic Slave CPL Buffer Allocation

ASIC Rev	Step	Register Settings	Function/Comment
RS780 All Revs	1	PCIEP_PORT_CNTL – PCIEIND_P: 0x10 CI_SLV_CPL_STATIC_ALLOC_LIMIT Set bits [14:8] depending on the configuration (the default is 0, meaning 128 slots)	Config A (4:0:0:0:0): Use default values Config B (4:4:0:0:0): Use default values Config C (4:2:2:0:0): Port A: Set bits[14:8] = 7'd32 Port B: Set bits[14:8] = 7'd16 Port C: Set bits[14:8] = 7'd16 Config D (4:2:1:1:0): Port A: Set bits[14:8] = 7'd32 Port B: Set bits[14:8] = 7'd16 Port C: Set bits[14:8] = 7'd12 Port D: Set bits[14:8] = 7'd12 Config E (4:1:1:1:1): Port A: Set bits[14:8] = 7'd32 Port B: Set bits[14:8] = 7'd12 Port C: Set bits[14:8] = 7'd12 Port D: Set bits[14:8] = 7'd12 Port E: Set bits[14:8] = 7'd12
	2	PCIE_CI_CNTL – PCIEIND: 0x20 CI_SLV_CPL_ALLOC_MODE Set bit[11] to 1	Enables dynamic buffer allocation

5.10.11 Autonomous GEN2 Speed Change (CMOS Option – Disabled by Default)

There should be 6 CMOS options for each GPP port:

- PCIE-GPPSB Port B Autonomous GEN2
- PCIE-GPPSB Port C Autonomous GEN2
- PCIE-GPPSB Port D Autonomous GEN2
- PCIE-GPPSB Port E Autonomous GEN2
- PCIE-GPP Port A Autonomous GEN2
- PCIE-GPP Port B Autonomous GEN2

Table 5-68 Autonomous GEN2 Speed Change

ASIC Rev	Step	Register Settings	Function/Comment
RS780 All Revs	1	BIF_NBP:LINK_CNTL2 -- pcieCopcieConfigDev*:0x88 TARGET_LINK_SPEED Set bits[3:0] to 4'h2	Sets TARGET_LINK_SPEED to GEN2.
	2	GPP Core: NBCFG: PCIE_NBCFG_REG9 – NBMISCIND: 0x39 STRAP_BIF_DE_EMPHASIS_SEL (port A) Set bit[31] to 1 NBCFG: PCIE_NBCFG_REGA – NBMISCIND: 0x22 STRAP_BIF_DE_EMPHASIS_SEL (port B) Set bit[5] to 1 GPPSB Core: NBCFG: PCIE_NBCFG_REG4 – NBMISCIND: 0x34 STRAP_BIF_DE_EMPHASIS_SEL (port B) Set bit[31] to 1 NBCFG: PCIE_NBCFG_REG7 – NBMISCIND: 0x37 STRAP_BIF_DE_EMPHASIS_SEL (port C) Set bit[5] to 1 NBCFG: PCIE_NBCFG_REG7 – NBMISCIND: 0x37 STRAP_BIF_DE_EMPHASIS_SEL (port D) Set bit[6] to 1 NBCFG: PCIE_NBCFG_REG7 – NBMISCIND: 0x37 STRAP_BIF_DE_EMPHASIS_SEL (port E) Set bit[7] to 1	Advertises -3.5 dB de-emphasis value in TS1 Data Rate Identifier
	3	BIF_NBP:PCIE_LC_SPEED_CNTL – PCIEIND_P:0xA4 LC_GEN2_EN_STRAP Set bit[0] to 1	Disables GEN2 capability of the device.
	4	BIF_NBP:PCIEP_STRAP_LC – PCIEIND_P:0xC0 STRAP_AUTO_RC_SPEED_NEGOTIATION_DIS Set bit [15] to 0	Sets AUTO RC SPEED NEGOTIATION
	5	BIF_NBP:PCIE_LC_SPEED_CNTL – PCIEIND_P:0xA4 LC_MULT_UPSTREAM_AUTO_SPD_CHNG_EN Set bit [29] to 1	Enables Gen2 Speed Change on any surprised link down
	6	BIF_NBP:PCIE_LC_LINK_WIDTH_CNTL[13]=1'b0·PCIEIND_P:0xA2 LC_UPCONFIGURE_DIS Clear bit[13] to 0.	Clears up gating off Upconfigure Support.

5.10.12 Link Training

Release hold training (by setting the corresponding hold training bit in section *5.5*) and then start the link training procedure outlined in section *5.7*.

5.10.13 Power Down Control

5.10.13.1 GPPSB Core

5.10.13.1.1 Inactive Lanes

The inactive lane(s) of each port is determined by performing the procedure found in *Table 5-69*:

Table 5-69 Inactive Lanes

ASIC Rev	Step	Register Setting	Function/Comment
RS780 All Revs	1	BIF_NB: PCIE_DEBUG_CNTL – PCIEIND: 0x12 DEBUG_PORT_EN Set bits[7:0] according to the Port	8'h1 = Port A 8'h2 = Port B 8'h3 = Port C 8'h4 = Port D 8'h5 = Port E
	2	BIF_NB: PCIE_LC_STATUS2 – PCIEIND: 0x29 LC_TOTAL_INACTIVE_LANES Read back bits[3:0]	Bit [0] represents lane 0 of the lanes assigned to the port. For example: If Lanes 4-7 are assigned to Port B, then bit [0] of the read back value represents whether lane 4 is active or not. 1=Inactive lane 0=Active lane

The per core indirect register PCIE_P_PAD_FORCE_DIS (PCIEIND 0x65) is used to power down inactive lanes in the GPPSB core. The transmitter and the receiver of each lane can be powered down independently; the register bit to lane mapping is shown in *Table 5-70* below:

Table 5-70 Transmitter and Receiver Shut Down

Lane	Receiver Shut Down	Transmitter Shut Down
SB TX/RX0	8	0
SB TX/RX1	9	1
SB TX/RX2	10	2
SB TX/RX3	11	3
GPP TX/RX0	12	4
GPP TX/RX1	13	5
GPP TX/RX2	14	6
GPP TX/RX3	15	7

- 4 dedicated SB lanes (Port A)

Table 5-71 SB Lanes

ASIC Rev	Step	Register Settings	Function/Comment
RS780 All Revs	1	Read back bit[6:4] of: BIF_NBP:PCIE_LC_LINK_WIDTH_CNTL[6:4] PCIEIND_P:0xA2 in Dev8 to determine the current link width. If width is x4 → skip this sequence If width is x2 → go to Step 2 If width is x1 → go to Step.3	Encoding of LC_LINK_WIDTH_RD: 000 = x16 001 = x1 010 = x2 011 = x4 100 = x8 101 = x12 (not supported) 110 = x16
	2	PCIE_P_PAD_FORCE_DIS – PCIEIND: 0x65 Set bits[3:2] to 2'b11 Set bits[11:10] to 2'b11	Powers down SB Lanes 3&2
	3	PCIE_P_PAD_FORCE_DIS – PCIEIND: 0x65 Set bits[3:1] to 3'b111 Set bits[11:9] to 3'b111	Powers down SB Lanes 3-1

- The 4 GPP lanes

Table 5-72, *Table 5-73*, *Table 5-74*, *Table 5-75*, and *Table 5-76* show the corresponding power down bits for each port in different configurations.

Table 5-72 GPP Port B (Lane Reversal Disabled)

Configuration	Lane Reversal Disabled		
	x0	x1	x2
Config B (4:4:0:0:0)	[7:4], [15:12]	[7:5], [15:13]	[7:6], [15:14]
Config C (4:2:2:0:0)	[5:4], [13:12]	5, 13	No inactive lanes
Config D (4:2:1:1:0)	Same as Config C		
Config E (4:1:1:1:1)	4, 12	No inactive lanes	N/A

Table 5-73 GPP Port B (Lane Reversal Enabled)

Configuration	Lane Reversal Enabled		
	x0	x1	x2
Config B (4:4:0:0:0)	[7:4], [15:12]	[6:4], [14:12]	[5:4], [13:12]
Config C (4:2:2:0:0)	[5:4], [13:12]	4, 12	No inactive lanes
Config D (4:2:1:1:0)	Same as Config C		
Config E (4:1:1:1:1)	4, 12	No inactive lanes	N/A

Table 5-74 GPP Port C

Configuration	x0	x1	x2
Config B (4:4:0:0:0)	N/A		
Config C (4:2:2:0:0)	[7:6], [15:14]	Reversal Disabled: 7, 15 Reversal Enabled: 6, 14	No inactive lanes
Config D (4:2:1:1:0)	6, 14	No inactive lanes	N/A
Config E (4:1:1:1:1)	5, 13	No inactive lanes	N/A

Table 5-75 GPP Port D

Configuration	x0	x1	x2
Config B (4:4:0:0:0)	N/A		
Config C (4:2:2:0:0)	N/A		
Config D (4:2:1:1:0)	7, 15	No inactive lanes	N/A
Config E (4:1:1:1:1)	6, 14	No inactive lanes	N/A

Table 5-76 GPP Port E

Configuration	x0	x1	x2
Config B (4:4:0:0:0)	N/A		
Config C (4:2:2:0:0)	N/A		
Config D (4:2:1:1:0)	N/A		
Config E (4:1:1:1:1)	7, 15	No inactive lanes	N/A

5.10.13.2 Turning Off Electrical Idle Detectors

The electrical idle detectors should be powered off when no GPP links is trained.

Table 5-77 Turning Off Electrical Idle Detectors Settings

ASIC Rev	Step	Register Settings	Function/Comment
RS780 All Revs	1	PCIE_NBCFG_REGC – NBMISCIND:0x24 B_PG2RX_IDLEDET_EN Set bit[19] to 0	Disables the electrical idle detectors for 2 GPP lanes.

5.10.13.1.3 Inactive PLL

Since there is only 1 PLL for the GPPSB core and the SB link is always alive, the PLL for this core can never be powered down by software programming. The only occasion that the PLL can be powered down is when all the ports in the GPPSB core are in L1/L23 power save states; the hardware is responsible for the power down in this case.

5.10.13.2 GPP Core

5.10.13.2.1 Inactive Lanes

The inactive lane(s) of each port is determined by performing the procedure found in *Table 5-78*.

Table 5-78 Inactive Lanes

ASIC Rev	Step	Register Settings	Function/Comment
RS780 All Revs	1	BIF_NB: PCIE_DEBUG_CNTL – PCIEIND: 0x12 DEBUG_PORT_EN Set bits[7:0] according to the Port	8'h1 = Port A 8'h2 = Port B
	2	BIF_NB: PCIE_LC_STATUS2 – PCIEIND: 0x29 LC_TOTAL_INACTIVE_LANES Read back bits[1:0]	Bit [0] represents lane 0 of the lanes assigned to the port 1=Inactive lane 0=Active lane

The per core indirect register PCIE_P_PAD_FORCE_DIS (PCIEIND 0x65) is used to power down inactive lanes in the GPP core. The transmitter and the receiver of each lane can be powered down independently; the register bit to lane mapping is shown in *Table 5-79*.

Table 5-79 Transmitter and Receiver Shut Down

Lane	Receiver Shut Down	Transmitter Shut Down
GPP TX/RX4	8	0
GPP TX/RX5	9	1

Table 5-80 and *Table 5-81* show the corresponding power down bits for each port in different configurations.

Table 5-80 GPP Port A

Configuration	x0	x1	x2
Config C (2:0)	[1:0], [9:8]	Reversal Disabled: 1, 9 Reversal Enabled: 0, 8	No inactive lanes
Config D (1:1)	0, 8	No inactive lanes	N/A

Table 5-81 GPP Port B

Configuration	x0	x1	x2
Config C (2:0)	N/A		
Config D (1:1)	1, 9	No inactive lanes	N/A

5.10.13.2.2 Inactive PLL

There is a dedicated PLL for the 2 lanes in the GPP core. If both lanes are inactive, the PLL can be powered down completely.

Table 5-82 Inactive PLL

ASIC Rev	Register Settings	Function/Comment
RS780 All Revs	NBCFG: PCIE_NBCFG_REG16 – NBMISCIND: 0x2E B_PPLL_PDNB_FDIS_GPP B_P90PLL_BUF_PDNB_TX_FDIS_GPP B_P90PLL_BUF_PDNB_RX_FDIS_GPP Set bit[27] = 1 Set bits[31:30] = 2'b11	Powers down the PLL completely. The only current consumption is leakage current.

5.10.14 Software Initiated Speed Change to GEN2 (CMOS Option – Disabled by Default)

The following 6 CMOS options should be available:

- PCIE-GPPSB Port B Software GEN2
- PCIE-GPPSB Port C Software GEN2
- PCIE-GPPSB Port D Software GEN2
- PCIE-GPPSB Port E Software GEN2
- PCIE-GPP Port A Software GEN2
- PCIE-GPP Port B Software GEN2

Table 5-83 Software Initiated Speed Change to GEN 2 (CMOS Option - Disabled by Default)

ASIC Rev	Step	Register Settings	Function/Comment
RS780 All Revs	1	Read back bits [3:0] BIF_NBP:LINK_CAP [3:0] · pcieConfigDev*:0x64 LINK_SPEED If read back values of [3:0] is 1 -> exit the sequence as the RC does not support Gen2 If read back value of [3:0] is 2 -> go to Step 2	Reads back Maximum Link speed of the given PCI Express Link advertised from the RC.
	2	In the Configuration Space of the EP device, read back bits [3:0]: BIF_NBP:LINK_CAP [3:0] · pcieConfigDev*:0x64 LINK_SPEED If read back values of [3:0] is 1 -> exit the sequence as the EP does not support Gen2 If read back value of [3:0] is 2 -> go to Step 3	Reads back Maximum Link speed of the given PCI Express Link advertised from the EP.
	3	Read back bit [24] of the: BIF_NBP:PCIE_LC_SPEED_CNTL[24] · PCIEIND_P:0xA4 LC_OTHER_SIDE_SUPPORTS_GEN2 If read back value of bit [24] is 1 -> go to Step 5 Otherwise -> go to Step 4	Checks if the other side of the link supports Gen2.

	4a	BIF_NBP:PCIE_LC_SPEED_CNTL[0] -- PCIEIND_P:0xA4 LC_GEN2_EN_STRAP Set bit[0] to 1 For AMD devices only, in the Configuration Space of the EP device: BIF_NBP:PCIE_LC_SPEED_CNTL[0] -- PCIEIND_P:0xA4 LC_GEN2_EN_STRAP Set bit[0] to 1	Device * advertises that it supports Gen2 Only for AMD Devices.
	4b	GPP Core: NBCFG: PCIE_NBCFG_REG9 – NBMISCIND: 0x39 STRAP_BIF_DE_EMPHASIS_SEL (port A) Set bit[31] to 1 NBCFG: PCIE_NBCFG_REGA – NBMISCIND: 0x22 STRAP_BIF_DE_EMPHASIS_SEL (port B) Set bit[5] to 1 GPPSB Core: NBCFG: PCIE_NBCFG_REG4 – NBMISCIND: 0x34 STRAP_BIF_DE_EMPHASIS_SEL (port B) Set bit[31] to 1 NBCFG: PCIE_NBCFG_REG7 – NBMISCIND: 0x37 STRAP_BIF_DE_EMPHASIS_SEL (port C) Set bit[5] to 1 NBCFG: PCIE_NBCFG_REG7 – NBMISCIND: 0x37 STRAP_BIF_DE_EMPHASIS_SEL (port D) Set bit[6] to 1 NBCFG: PCIE_NBCFG_REG7 – NBMISCIND: 0x37 STRAP_BIF_DE_EMPHASIS_SEL (port E) Set bit[7] to 1	Advertises -3.5 dB de-emphasis value in TS1 Data Rate Identifier
	4c	BIF_NBP:PCIE_LC_LINK_WIDTH_CNTL - PCIEIND_P:0xA2 LC_UPCONFIGURE_DIS Clear bit[13] to 0.	Clears up gating off Upconfigure Support.
	5	BIF_NBP:LINK_CNTL2 -- pcieCopcieConfigDev*:0x88 TARGET_LINK_SPEED Set bits[3:0] to 4’h2 In the Configuration Space of the EP device: BIF_NBP:LINK_CNTL2[3:0] -- pcieCopcieConfigDev*:0x88 TARGET_LINK_SPEED Set bits[3:0] to 4’h2	Sets Port * link speed to be Gen2
	6	BIF_NBP:PCIE_LC_SPEED_CNTL -- PCIEIND_P:0xA4 LC_GO_TO_RECOVERY Set bit [18] to 1.	Initiates Recovery

	7	Pool back [5:0] of: BIF_NBP:PCIE_LC_STATE0[5:0] · PCIEIND_P:0xA5 LC_CURRENT_STATE Until 5'h10.	Waits until link retrains to L0.
	8	Read back bit [24] of the: BIF_NBP:PCIE_LC_SPEED_CNTL[24] · PCIEIND_P:0xA4 LC_OTHER_SIDE_SUPPORTS_GEN2 If read back value of bit [24] is 1 -> go to next step Otherwise -> skip the sequence	Checks if the other side of the link supports Gen2.
	9	BIF_NBP:PCIE_LC_SPEED_CNTL -- PCIEIND_P:0xA4 LC_INITIATE_LINK_SPEED_CHANGE Set bit[7] to 1	Initiates Link Speed Change

5.10.15 Active State Power Management (ASPM)

5.10.15.1 ASPM L1 (CMOS Option - Enabled by Default)

There should be 7 CMOS options:

- PCIE-GPPSB Port A ASPM L1
- PCIE-GPPSB Port B ASPM L1
- PCIE-GPPSB Port C ASPM L1
- PCIE-GPPSB Port D ASPM L1
- PCIE-GPPSB Port E ASPM L1
- PCIE-GPP Port A ASPM L1
- PCIE-GPP Port B ASPM L1

Table 5-84 ASPM L1 (CMOS Option - Disabled by Default)

ASIC Rev	Step	Register Settings	Function/Comment
RS780 All Revs	1	BIF_NBP:LINK_CNTL[1]=1'b1· pcieConfigDev*:0x68 PM_CONTROL Set bit [1] to 1	Enables L1 in North Bridge side of the link.
	2	In the register space of SB and AMD EP Devices only: BIF:PCIE_LC_CNTL[15:12]=4'h6· PCIEIND_P :0xA0 LC_L1_INACTIVITY Set bits[15:12] to 4'h6	For SB and AMD EP Devices: Sets the L1 inactivity timer in to be 10000 TXCLKs Gen1: 10 000 x 4ns = 40us Gen2: 10 000 x 2ns = 20us
	3	In the EP device, follow the capability list to find the PCIE capability (capability ID = 0x10). BIF_NBP:LINK_CNTL[1]=1'b1· pcieConfigDev*:0x68 PM_CONTROL Set bit [1] to 1	Enables L1 in the graphics card(s)

5.10.15.2 Powering Off PLL During L1/L23 (CMOS Option – Disabled by Default)

There should be 2 CMOS options:

- PCIE-GPPSB Powering Off PLL during L1/L23
- PCIE-GPP Powering Off PLL during L1/L23

This feature requires ASPM L1 to be enabled for all the ports in the core and the PLL will only be powered off when all the ports are in L1.

Table 5-85 Powering Off PLL During L1/L23 (CMOS Option - Disabled by Default)

ASIC Rev	Step	Register Settings	Function/Comment

RS780 All Revs	1	PCIE_P_CNTL – PCIEIND: 0x40 P_PWRDN_EN Set bit[0] to 1	Enables powering down transmitter and receiver pads along with PLL macros
	2	PCIE_P_CNTL – PCIEIND: 0x40 P_PLL_PWRDN_IN_L1L23 Set bit[3] to 1	Enables PLL power down during L1
	3	PCIE_P_CNTL – PCIEIND: 0x40 P_PLL_BUF_PDNB Set bit[4] to 0	Active-low signal to enable PLL buffers to be powered down during L1
	4	PCIE_P_CNTL – PCIEIND: 0x40 P_PLL_PDNB Set bit[9] to 0	Active-low signal to enable PLL to be powered down during L1
	5	PCIE_P_CNTL – PCIEIND: 0x40 P_ALLOW_PRX_FRONTEND_SHUTOFF Set bit[12] to 1	Allows RX front end to be shutoff during L1 when PLL power down is enabled
	6	BIF_NB:PCIE_HW_DEBUG -- PCIEIND:0x2 HW_08_DEBUG Set bit [8] to 1	PLL_OFF_INSTABILITY_FIX_ENABLE

Note: The following sequence(s) should be performed if there are any untrained ports in the core:

- In the register space of the core with untrained ports:

Table 5-86 Core Untrained Ports

ASIC Rev	Step	Register Settings	Function/Comment
RS780 All Revs	1	PCIE_HW_DEBUG – PCIEIND: 0x02 HW_03_DEBUG Set bit[3] to 1	REGS_INACTIVE_LANES_TRIGGER_PLL_PDN

- In the register space of each of the untrained ports:

Table 5-87 Untrained Ports

ASIC Rev	Step	Register Settings	Function/Comment
RS780 All Revs	1	PCIE_LC_CNTL2 – PCIEIND_P: 0xB1 LC_ASSERT_INACTIVE_DURING_HOLD Set bit[22] to 1	Asserts the INACTIVE_LANES signals when CHIP_BIF_hold_training is high

5.10.15.3 L0s (CMOS Option – Disabled by Default)

The following 14 CMOS options should be available:

- PCIE-GPPSB Port [A-E] Transmitter L0s
- PCIE-GPPSB Port [A-E] Receiver L0s
- PCIE-GPP Port [A,B] Transmitter L0s
- PCIE-GPP Port [A,B] Receiver L0s

To enable Transmitter L0s use the register settings in *Table 5-88*.

Table 5-88 Transmitter L0s Settings

ASIC Rev	Step	Register Settings	Function/Comment

RS780 All Revs	1	PCIE_LC_TRAINING_CNTL –PCIEIND_P:0xA1 LC_DONT_GO_TO_L0S_IF_L1_ARMED Set bit[11] to 1	Disables L0s entry if en route to L1
	2	PCIE_LC_CNTL – PCIEIND_P: 0xA0 LC_L0S_INACTIVITY Set bits[11:8] to 4'h8	Sets L0s inactivity timer to 10us
	3	LINK_CNTL – pcieConfigDev*: 0x68 PM_CONTROL Set bit[0] to 1	Enables L0s

To enable Receiver L0s (registers are programmed on the end point's register space) use the register settings in *Table 5-89*:

Table 5-89 Receiver L0s Settings

ASIC Rev	Step	Register Settings	Function/Comment
RS780 All Revs	1	Read bit[10] of the following register: BIF:LINK_CAP[10]· pcieConfigDev2:0x64 LINK_CAP If read back value of bit [10] is 1 -> go to next step. Otherwise, skip this sequence.	Bit[10] = 1 means the device supports L0s ->go to Step 2. Bit[10] = 0 means the device does not support L0s. Stop
	2	BIF:LINK_CNTL[1:0]=4'h8· pcieConfigDev*:0xA0 LC_L0S_INACTIVITY Set bit[11:8] to 4'h8	This step is only applicable if the device is an AMD graphics card Set L0s inactivity timer to 1000 TXCLKs. Gen1: 1000x4ns=4us Gen2: 1000x2ns=2us
	3	BIF:LINK_CNTL[0]=1'b1· pcieConfigDev*:0x68 PM_CONTROL Set bit [0] to 1.	L0s should not be enabled for AMD GFX cards earlier than R5xxx. Only the R5xxx and higher generations support L0s properly.

5.10.16 Clock Gating

5.10.16.1 TXCLK Gating (CMOS Option – Disabled by Default)

Table 5-90 GPPSB Core

ASIC Rev	Step	Register Settings	Function/Comment
RS780 All Revs	1	PCIE_CONFIG_CNTL – PCIEIND: 0x11 DYN_CLK_LATENCY Set bits[3:0] to 4'b1100	Sets dynamic clock latency in bif_core
	2	PCIE_PDNB_CNTL – NBMISCIND : 0x7 ENABLE_CLKGATE_GPPSB_TXCLK ENABLE_CLKGATE_GPPSB_TXCLK_L0S ENABLE_CLKGATE_GPPSB_TXCLK_SND_RCV Set bit[6:4] to 3'b111	Enables clock gating on all TXCLK branches
	3	PCIE_P_CNTL – PCIEIND: 0x40 P_TXCLK_SND_PWRDN Set bit[5] to 1	Allows TXCLK_SND to be powered down
	4	PCIE_P_CNTL – PCIEIND: 0x40 P_TXCLK_RCV_PWRDN Set bit[6] to 1	Allows TXCLK_RCV to be powered down

Table 5-91 GPP Core

ASIC Rev	Step	Register Settings	Function/Comment
RS780 All Revs	1	PCIE_CONFIG_CNTL – PCIEIND: 0x11 DYN_CLK_LATENCY Set bits[3:0] to 4'b1100	Sets dynamic clock latency in bif_core
	2	PCIE_PDNB_CNTL – NBMISCIND : 0x7 ENABLE_CLKGATE_GPP_TXCLK ENABLE_CLKGATE_GPP_TXCLK_L0S ENABLE_CLKGATE_GPP_TXCLK_SND_RCV Set bit[10:8] to 3'b111	Enables clock gating on all TXCLK branches
	3	PCIE_P_CNTL – PCIEIND: 0x40 P_TXCLK_SND_PWRDN Set bit[5] to 1	Allows TXCLK_SND to be powered down
	4	PCIE_P_CNTL – PCIEIND: 0x40 P_TXCLK_RCV_PWRDN Set bit[6] to 1	Allows TXCLK_RCV to be powered down

5.10.16.2 LCLK Gating (CMOS Option – Disabled by Default)

Table 5-92 GPPSB Core

ASIC Rev	Step	Register Settings	Function/Comment
RS780 All Revs	1	CLKCFG: 0x94 CLKGATE_DISABLE Set bit[24] to 0	Enables LCLK gating for the GPPSB core

Table 5-93 GPP Core

ASIC Rev	Step	Register Settings	Function/Comment
RS780 All Revs	1	CLKCFG: 0xE8 CLK_TOP_SPARE_C Set bit[31] to 1	Enables LCLK gating for the GPP core

5.10.16.3 Shutting Off TXCLK Permanently (CMOS Option – Disabled by Default)

The TXCLK can be shut off permanently to save power when the whole GPP core is inactive. When this feature is enabled, all register reads/writes to the core will be invalid.

Table 5-94 Shutting Off TXCLK Permanently (CMOS Option - Disabled by Default)

ASIC Rev	Step	Register Settings	Function/Comment
RS780 All Revs	1	PCIE_PDNB_CNTL – NBMISCIND : 0x7 GPP_PERM2_TXCLK_STOP Set bit[11] to 1	Shuts off TXCLK permanently in the GPP core

5.10.17 Non-Posted VC1 Traffic Support on SB Link (CMOS Option – Disabled by Default)

Note: Steps 1-3 in the programming sequence in *Table 5-95* have to be done in both the NB and the SB register space.

Table 5-95 Non-Posted VC1 Traffic Support on SB Link (CMOS Option - Disabled by Default)

ASIC Rev	Step	Register Settings	Function/Comment
RS780 All Revs	1	BIF_NBP:PCIE_VC1_RESOURCE_CNTL -- pcieConfigDev8:0x130 TC_VC_MAP_TC1_7 Set bit[7:1] to 7'h7F	Maps Traffic Class 1-7 to VC1
	2	PCIE_VC1_RESOURCE_CNTL -- pcieConfigDev8: 0x130 VC_ID Set bit[26:24] to 2'b01	This field assigns a VC ID to the VC resource (assign VC ID to 1)
	3	PCIE_VC1_RESOURCE_CNTL -- pcieConfigDev8: 0x130 VC_ENABLE Set bit[31] to 1	Enables VC1
	4	Poll bit [1] of: PCIE_VC1_RESOURCE_CNTL -- pcieConfigDev8: 0x124 VC_NEGOTIATION_PENDING Poll bit[1] until 0	Bit[1] = 0 means that VC1 flow control initialization is successful

5.11 Dynamic Link Width Control (CMOS Option – Disabled by Default)

Dynamic link width control is a power saving feature which reconfigures the link to run with fewer lanes than the maximum available lanes.

- The GFX links can switch among widths of: x1, x2, x4, x8 and x16.
- The GPP links can switch among widths of: x1, x2 and x4.
- The SB link can switch among widths of: x1, x2 and x4.

There are 3 types of dynamic link width control mechanisms:

- Long Reconfiguration: the link goes down and retrains to a different width. This mechanism should only be used on an AMD-AMD link.
- Short Reconfiguration: the link retrains to a different width by going through the recovery state (i.e. the link does not go down). This mechanism should only be used on an AMD-AMD link.
- Up/Down Reconfiguration: the link retrains according to PCIE 2.0 Base Spec Compliant Reconfiguration. This mechanism should only be used on PCIE 2.0 Base Spec Compliant Devices.

The reconfiguration programming sequence is outlined in the table below. The sequence is for Up/Down reconfiguration only.

Table 5-96 Dynamic Link Width Control

ASIC Rev	Step	Register Settings	Function/Comment
RS780 All Revs	1a	Read back bit [9] of the following register: BIF_NBP:PCIE_LC_LINK_WIDTH_CNTL[9] ·PCIEIND_P:0xA2 LC_RENEGOTIATION_SUPPORT If the read back value of bit [9] is 0 than skip this sequence, otherwise ->go to Step 1b	Checks if the other End supports Up/Down Reconfiguration.
	1b	BIF_NBP:PCIE_LC_LINK_WIDTH_CNTL[12]=1'b1 PCIEIND_P:0xA2 LC_UPCONFIGURE_SUPPORT Set bit [12] to 1	Advertises support for Up/Down Reconfiguration.
	1c	BIF_NBP:PCIE_LC_LINK_WIDTH_CNTL[13]=1'b0·PCIEIND_P:0xA2 LC_UPCONFIGURE_DIS Clear bit[13] to 0.	Clears up gating off Upconfigure Support.
	1d	Read back current link width by reading bits [6:4] of the following register: BIF_NBP:PCIE_LC_LINK_WIDTH_CNTL[6:4] -- PCIEIND_P:0xA2 LC_LINK_WIDTH_RD Read back intended link width by reading bits [2:0] of the same register: BIF_NBP:PCIE_LC_LINK_WIDTH_CNTL[2:0] -- PCIEIND_P:0xA2 LC_LINK_WIDTH	Encoding of the link width: 000 = x16 001 = x1 010 = x2 011 = x4 100 = x8 101 = x12 (not supported) 110 = x16

Table 5-96 Dynamic Link Width Control

ASIC Rev	Step	Register Settings	Function/Comment
	2	If the current link width is less than the intended link width based on CMOS option selected, then grey out the link width selected in the CMOS option and skip this sequence. If the current link width is equal 3'b101, then the link must be retrained to x8, and then proceed with the sequence, and in step 4 set the desired link width to 3'b100. If the current link width is equal to intended link width, then skip this sequence. If the current link width is greater than the intended link width, then proceed with the sequence.	This check prevents users from trying to train the link to greater link width than maximum value limited with number of physical lanes connected on the PCIE slot. This step prevents users from trying to train the link in x12, and it downgrades to the first lower link width value supported, which is x8.
	3	BIF_NB:PCIE_P_CNTL[0]=1'b1-- PCIEIND:0x40 P_PWRDN_EN Set bit [0] to 1 On the AMD GFX card: BIF:PCIE_P_CNTL[0]=1'b0· PCIEIND:0xB0 P_PWRDN_EN Set bit [0] to 0	Enables powering down transmitter and receiver pads along with PLL macros. For AMD cards only.
	4	BIF_NBP:PCIE_LC_LINK_WIDTH_CNTL -- PCIEIND_P:0xA2 LC_LINK_WIDTH Set bits[2:0] to desired link width	Sets the desired link width using the encoding scheme in Step 1 based on CMOS option selected.
	5a	BIF_NBP:PCIE_LC_LINK_WIDTH_CNTL[10]=1'b1·PCIEIND_P:0xA2 LC_RENEGOTIATE_EN Set bit [10] to 1.	Enables Up/Down Reconfiguration
	5b	For AMD cards only, in the private register space of the EP device: BIF_NBP:PCIE_LC_CNTL2[23]=2'b1· PCIEIND_P:0xB1 LC_WAIT_FOR_LANES_IN_LW_NEG Set bit [23] to 1	Make ConfigStep2-2b timeout to 1us
	5c	BIF_NBP:PCIE_LC_LINK_WIDTH_CNTL -- PCIEIND_P:0xA2 LC_RECONFIG_NOW Set bit [8] to 1	Starts reconfiguration
	6	BIF_NBP:LINK_STATUS[11] ·pcieConfigDev*:0x6a LINK_TRAINING Poll bit[11] until 0	Ensures that link training is completed. Hardware clears this bit once Link training is complete.
	7	BIF_NBP:PCIE_LC_LINK_WIDTH_CNTL -- PCIEIND_P:0xA2 LC_RECONFIG_NOW Poll bit[8] until 0	Ensures that reconfiguration is completed. Hardware clears this bit once reconfiguration is complete.
	8	BIF_NBP:PCIE_VC0_RESOURCE_STATUS pcieConfigDev*:0x12a VC_NEGOTIATION_PENDING Poll bit[1] until 0	Ensures that virtual channel negotiation is done
	9	Link width change is completed. Power down unused lanes and PLL (if applicable)	Unused lanes should be powered off. See Power Down Control subsections in Section 5.9 and Section 5.10.

Table 5-96 Dynamic Link Width Control

ASIC Rev	Step	Register Settings	Function/Comment
	10	BIF_NB:PCIE_P_CNTL[0]=1'b0· PCIEIND:0x40 P_PWRDN_EN Clear bit [0] to 0 On the AMD GFX card: BIF:PCIE_P_CNTL[0]=1'b1· PCIEIND:0xB0 P_PWRDN_EN Set bit [0] to 1	Disables powering down transmitter and receiver pads along with PLL macros For AMD cards only in dual PCIE-GFX

5.12 PCI Enumeration and Special Features Programming Sequence

5.12.1 PCI Enumeration

The SBIOS scans all of the PCI buses looking for P2P bridges. When a P2P bridge is located, it is assigned bus numbers and a routine is performed to check if this P2P bridge is PCIE. If it is PCIE, then PCIE root port initialization is performed on that P2P bridge. If an error occurs during this initialization, the BIOS Vendor routine calls a chipset-specific routine to hide the PCIE P2P bridge that generated the error. The following PCIE registers are touched by the PCIE root port initialization.

Note: All PCIE registers are accessed through PCIE memory mapped configuration space.

5.12.2 Program the Common Clock Configuration

- Call OEM routine to determine if clocks are common to PCIE Port and endpoint.
- Program CommonClockConfig (bit6) = 1 in PCIE Link control reg in P2P and endpoint.
- Re-train the link. Uses PCIE LinkControl and LinkStatus regs.
  - Note: LinkControl & LinkStatus are accessed via memory mapped config space.
- Loop with a delay of 1ms between each read of PCIE LinkStatus
  - if bit11=0 then exit loop
  - if loop count == 100 then exit loop with error flag set this will cause Port to be hidden.

5.12.3 Slot Power Limit (CMOS Option - Default 75W)

Table 5-97 Slot Power Limit (CMOS Option – Default 75W)

ASIC Rev	Register Settings	Function/Comment
RS780 All Revs	BIF_NBP:SLOT_CAP · pcieConfigDev12:0x6c SLOT_PWR_LIMIT_VALUE[14:7]=8'h4B SLOT_PWR_LIMIT_SCALE[16:15]=2'h0 Set bits [14:7] to 7'h4B Set bits [16:15] to 2'h0	Sets Power Limit to 75W Implement x1 Multiplier

5.12.4 Update Hot-Plug Info

```
if hot-plug then
    Clear HotPlug controller cmd & status regs
    Program other PCIE capability & HWINIT bits
endif
```

5.12.5 Disable Immediate Timeout on Link Down

Table 5-98 Disable Immediate Timeout on Link Down

ASIC Rev	Register Settings	Function/Comment
RS780 All Revs	BIF_NBP:PCIE_RX_CNTL[19] = 1'b0 PCIEIND_P:0x70 RX_RCB_CPL_TIMEOUT_MODE Clear bit [9] to 1	Disables immediate RCB timeout on link down

5.12.6 Register Locking

Table 5-99 Register Locking

ASIC Rev	Step	Register Settings	Function/Comment
RS780 All Revs	1	BIF_NB:PCIE_CNTL[0]=1'b1· PCIEIND:0x10 HWINIT_WR_LOCK Set bit[0] to 1	This should be set for all 3 cores (GFX, GPPSB and GPP). Makes the HWINIT registers read-only.
	2	NBCFG:NB_CNTL[1]=1'b1· NBMISCIND:0x0 HIDE_NB_AGP_CAP Set bit[0] to 1	Hides AGP Capabilities

5.12.7 Optional Features

The optional features for the GFX core (sections *5.9.16* to *5.9.18*) and the GPPSB/GPP cores (sections *5.10.14* to *5.10.17*) should be executed here if the CMOS option(s) is enabled.

5.12.8 Dynamic Link Width Control

The dynamic link width control code (section *5.11*) can be executed here if a link width change is required.

5.12.9 Special Features Programming Sequence

5.12.9.1 RV370/RV380 Graphics Card Initialization

Table 5-100 lists the affected AMD device IDs:

Table 5-100 Affected AMD Device IDs

First Device	Second Device
3E50	3E70
3E52	3E72
3E54	3E74
3150	3170
3154	3174
3151	3171
3152	3172
5B60	5B70
5B64	5B74
5460	5470
5464	5474
5B62	5B72
5B63	5B73
5B65	5B75
5B61	5B71
5B66	5B76
5B67	5B77
5460	5470
5461	5471
5462	5472
5463	5473
5464	5474
5465	5475
5466	5476
5467	5477
5B66	5B76
5B67	5B77
5466	5476
5467	5477

Table 5-101 Clock Recovery Phase Filter Size Settings

ASIC Rev	Step	Register Settings	Function/Comment
RS780 All Revs	1	PCIE_NBCFG_REG8 - NBMISCIND:0x38 B_P90RX_CRPHSIZE Set bits[20:19] to 2'b0	Sets the clock recovery phase filter size specifically for the RV370.

```
//RV370/RV380 Graphics Card Initialization
if devNum == 2/3 then
    Program bus0/dev2 secondary/subordinate bus numbers = 32
    Read VendorID as word from bus32/dev0/func0
    if VendorID == 0xFFFF then
        write to PCIEIND_P:0xA2 to re-train the link: set bit [8] to 1;
        set bits [2:0] = bits [6:4].
        Loop back to section 5.7.
    endif
    if VendorID == 0x1002(AMD) then
        bus0/dev2 non-prefetch memory window(reg:0x20) = 0xC000C000
        bus0/dev2 PCI Cmd bits [7:0] = 0x02 enable memory decode
        bus32/dev0 BAR2 = 0xC0000000
        read back bus32/dev0 BAR2
        if bits [31:16] != 0xC000 then
            Do CF9 reset
        endif
        read bus32/dev0 PCICmd as word
        if PCICmd == 0xFFFF then
            Do CF9 reset
        endif
        PCICmd[1] = 1 and write back
        read bus32/dev0 PCICmd as word
        if PCICmd == 0xFFFF then
            Do CF9 reset
        endif
        Write 0xB700 to memory address 0xC0000120 (claimed by BAR2 of PCIE GFX Card at bus32/dev0)
        NOP
        if [0xC0000120] != 0xB700 then
            Do CF9 reset
        endif
        Write 0x13 to memory address 0xC0000124
        NOP
        if [0xC0000124] != 0x13 then
            Do CF9 reset
        endif
        if !([0xC000012C] and BIT8) then
            GOTO linkDone and move on to the next device.
        else
            Do CF9 reset
        endif
    endif

linkDone:
    Clear bus0/dev2 secondary/subordinate bus numbers(including secondary latency)
    Clear bus0/dev2 non-prefetch memory window
    Clear bus0/dev2 bits [7:0] of PCI Cmd register
endif
```

5.12.9.2 Nvidia External Graphics Card Initialization

To workaround the S3 Resume issue with the Nvidia card, perform the following: when the system boots up: record the SSVID/SSDID (address 0x2C) to CMOS. During the S3 resume, read the SSVID/SSDID and compare the result with the expected number in CMOS. If it is different, then restore the new value from CMOS to offset 0x40 in Nvidia.

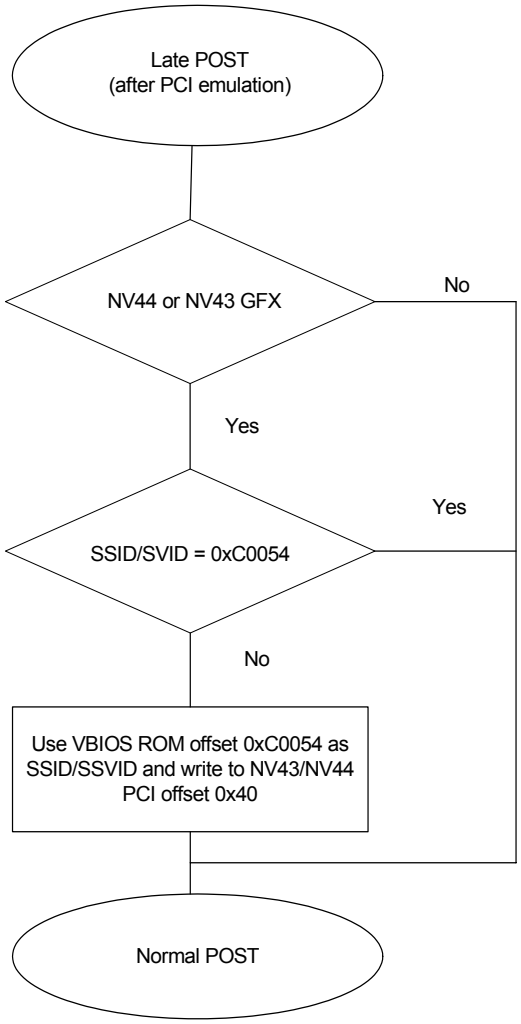

Figure 5-1 Nvidia External Graphics Card Initialization

5.12.9.3 Hot Plug Support

Table 5-102 Hot Plug Support

ASIC Rev	Step	Register Settings	Function/Comment
RS780 All Revs	1	BIF_NBP:PCIE_ROOT_CONTROL -- PCIEIND_P:0x74 PM_INTERRUPT_EN Set bit 3 to 1'b1	Enables Interrupt generation on PME event
	2	BIF_NBP: PCIEP_PORT_CNTL -- PCIEIND_P:0x10 HOTPLUG_MSG_EN Set bit 2 to 1	Enables hotplug message

Perform the following steps when the system detects that a device is plugged in:

- Step 1: Enable the GPP lanes.
- Step 2: Enable the training
- Step 3: Check the link status, and retrain the link if the training failed through PCIEIND_P: 0xA2.
  - Step 3.1: Perform the following steps to retrain the link:

- Step 3.1.1: Detect if the card is trained to L0 from PCIEIND_P: 0xA5 bits [5:0]. Move to the next step if it is 6'h10. Otherwise repeat up to 1 second, then set the HOLD_TRAINING bit to 1.
- Step 3.1.2: Detect if Data Link Negotiation is done from the VC_NEGOTIATION field in PCIE_VC0_RESOURCE_STATUS. Move to the next step if it is 0. Otherwise write to PCIEIND_P: 0xA2 to re-train the link: set bit [8] to 1; set bits [2:0] = bits [6:4], wait for 5ms, then loop back to Step 1. Stay in this loop for a maximum of 15 times. Set HOLD_TRAINING to 1 if the hot plug device failed the checking.

Perform the following steps when the system detects that a hot plug device is removed:

- Step 1: Issue a dummy CFG read to the removed device (expect FF back)
- Step 2: Check the A5 register to see if it has any value between 00 to 04, disable the training, and power-down the lanes. If the value is not between 00 to 04, then issue a dummy CFG read, and check the A5 register. Stay in this loop for a maximum of 5 times.
- Step 3: Put Device into D3.

5.12.9.4 Atheros Card Initialization

For Atheros XB6x device, L1 can be enabled (if CMOS option is to enable L1).

When BIOS detects that an XB6x device is present (Vendor ID 0x168c) behind a PCIE bridge, the following in *Table 5-103* should occur.

Note: If the settings below are not configured properly, then there is a possibility of a hard hang.

Table 5-103 Atheros Card Initialization

ASIC Rev	Step	Register Settings	Function/Comment
RS780 All Revs	1	BIF_NBP:LINK_CNTL[1]=1'b1· pcieConfigDev2:0x68 PM_CONTROL Set bit [1] to 1.	For the device behind which the XB6x is found and CMOS L1 option is enabled, enable L1 support.
	2	Set XB6x 0x70C=0x0F003F01	Enables workaround on Atherous side.
	3	Set XB6x 0x70[1:0]=xxxxxxx2 Set bit [1] to 1.	Enables L1 support on Atheros side if CMOS L1 option is enabled.

5.12.9.5 System Information Table Setting for PowerExpress Mode

Table 5-104 Sytem Information Table Setting for PowerExpress Mode

ASIC Rev	Step	Register Settings	Function/Comment
RS880 All Revs	1	BIF_NBP:LINK_CNTL2 -- pcieConfigDev*:0x88 TARGET_LINK_SPEED	4'h2: Advertises the link speed to be Gen2. 4'h1: Advertises the link speed to be Gen1.
	2	Check the register in step 1 for all ports, then based on whether PowerExpress mode is running, set bit [6] in the Integrated information table accordingly.	Bit [6] = 0 when [(no Gen2 GPP devices are populated) && (((PCIE gfx link is trained in Gen2) && (Running in PowerExpress mode)) || (PCIE gfx link is trained in Gen1))] Bit [6] = 1 when [(any Gen2 GPP device is populated) || ((PCIE GFX link is trained in Gen2) && (not runnning in PowerExpress mode))]

Chapter 6
Graphics Core Settings

6.1 Bus Interface (BIF)

For the most part, the RS780 BIF is based on the RV610 (laka) design with the front-end PCIE interface removed. Although the PCIE-specific registers still exist, most of them do not perform any function (writes do not affect operation).

The following are other notable differences:

- DEVICE_IDs, MAJOR_REV_IDs, and MINOR_REV_IDs are hardcoded and set at the ASIC level. See section *6.2*.
- CFG_ATI_REV_ID is now available in CONFIG_CNTL (as in other integrated graphics devices). See section *6.3*.
- GFX_DEBUG_BAR has been added (as in other integrated graphics devices). See section *6.4*.
- The graphics device now appears as a PCI device (as opposed to a PCIE device).
  - Removes PCI-e capabilities from CAP_PTR linked list in PCI configuration space
- BIOS_SCRATCH_0 to BIOS_SCRATCH_15 registers are available
- Straps must be programmed by the SBIOS. See section *6.6*.
- Master-abort status is available via CFG status bit. See section *6.7*.

6.2 DEVICE_IDs, MAJOR_REV_IDs, MINOR_REV_IDs

The graphics functions (F0 in single display mode, or F0/F1 in dual display mode) and audio function if enabled (F1 in single display mode, or F2 in dual display mode) have the following IDs:

- VENDOR_ID:
  - 0x1002 – ATI if VENDOR_ID[1] eFuse is not set
  - 0x1022 – AMD if VENDOR_ID[1] eFuse is set
- Graphics DEVICE_ID:
  - First graphics device: 0x9610 + (CFG_FAMILY_ID[4:2] eFuse)
  - Second graphics device: 0x9630 + (CFG_FAMILY_ID[4:2] eFuse)
  - Audio device: 0x960F
- MAJOR_REV_ID/MINOR_REV_ID:
  - Graphics devices: MAJOR_REV_ID = 0x0, MINOR_REV_ID = 0x0 in A11
  - Audio device: MAJOR_REV_ID = 0x0, MINOR_REV_ID = 0x0 in A11

6.3 CFG_ATI_REV_ID

The CFG_ATI_REV_ID field reads back the following:

Table 6-1 CFG_ATI_REV_ID Read Back Values

EFUSE_CFG_FAMILY_ID[6:5]	A11	A12	A13
00	0x00	0x01	0x01
01	0x00	0x01	0x03
10	0x00	0x01	0x01
11	0x00	0x01	0x03

6.4 GFX_DEBUG_BAR

The GFX_DEBUG_BAR provides a way for the software driver to access the BIF's configuration space. Memory-mapped accesses that hit this aperture are converted into the corresponding configuration read/write cycles. This function is intended for the driver and might not be a concern of the SBIOS.

This function is enabled by first setting NB_GC_STRAPS.GFX_DEBUG_BAR_EN = 1. Next, function 0 BAR 6 should be written. The 1MB aperture is fixed as a non-prefetchable region that supports 32b addressing only.

6.5 Gpuioreg BAR For Accessing nbconfig Registers (A12)

If this function is enabled, then the accesses to internal graphics IO space, with offset 0x60/0x64, are forwarded to nbconfig:0x60/0x64. To enable the decoding with the IOC the following bit should be set:

- NBMISCIND (offset 0x1): Bit [8] needs to be 1

6.6 Initialization

Set the following CFG registers:

- NB_GC_STRAPS
- NB_INTERRUPT_PIN

BIF requires several “strap” bits to be set before it can function. These bits should be loaded into NB_NBMISCCFG: NB_BIF_SPARE as follows:

Table 6-2 Strap Bits

NB_NBMISCCFG: NB_BIF_SPARE	Field	Description
Bit 10	MSI_DATA_FIX_EN	This is for the RS780 ASIC revision A13 and above 0=Disable ECO for MSI DATA bug causing incorrect MSI to be written 1=Enable
Bit 9	MSI_BE_FIX_DIS	This is for the RS780 ASIC revision A12 and above 0=Enable ECO for MSI BE alignment bug causing MSI to go to the incorrect address 1=Disable
Bit 8	CFG_BIF_BIOS_ROM_EN	Not used. Leave as ‘0’
Bit 7	BIF_MEM_AP_SIZE_STRAP_SEL	Write as ‘1’
Bit 6	BIF_AUDIO_EN_STRAP_SEL	Write as ‘1’
Bit 5	RCU_BIF_config_done	See below: at this point, write as ‘0’
Bit 4	SLV_BD_RAD_FORCE_EN	Write as ‘0’: effects back-door access (below)
Bit 3	SLV_BD_RAD_MWr4_DIS	Write as ‘0’: effects back-door access (below)
Bit 2	SLV_BD_RAD_MWr3_DIS	Write as ‘0’: effects back-door access (below)
Bit 1	CFG_BIF_MSI_EN	1=Enable MSI 0 =Disable MSI
Bit 0	Reg_BIF_RST_DIS	Write as ‘0’: used to control driver resets of BIF

Before BIF can be used, the SBIOS must provide strap values to it. After hard reset, BIF is in strap mode where normal cycles are not permitted; but instead BIF interprets posted memory mapped writes as writes to various ROMSTRAP and EFUSE registers.

It is suggested that these registers be written in the following order before any regular BIF registers are written (i.e., before PCI config cycles to the graphics device). Only the 20b LSBs of the byte address are relevant for these cycles (the MSBs are ignored). To prevent confusion within the CPU and northbridge, it is suggested that the MSBs be selected such that the addresses fall within the memory-mapped BAR that will eventually be used by the graphics device.

Table 6-3 Initialization

Address LSB[19:0]	Register	Value
0x15000	CC_BIF_ROMSTRAP0	Recommended: 0x2C006300
0x15010	CC_BIF_ROMSTRAP1	Recommended: 0x03015330
0x15020	CC_BIF_ROMSTRAP2	Recommended: 0x04000040 Commonly used bits: Bit [5]=BIF_64BAR_EN_A Bits [9:7]=BIF_MEM_AP_SIZE[2:0] Bit [10]=BIF_REG_AP_SIZE[1] Bit [25]=BIF_DUALFUNC_DISPLAY_EN Bit [26]=BIF_AUDIO_EN Bit [27]=BIF_MSI_DIS Bit [31]=BIF_TRUSTED_CFG_EN
0x15030	CC_BIF_ROMSTRAP3	Recommended: 0x00001002

0x15040	CC_BIF_EFUSE0	Recommended: 0x00000000
0x15050	CC_BIF_EFUSE1	Recommended: 0x00000000
0x15220	CC_BIF_ROMSTRAP5	Recommended: 0x03C03800
0x15060	CC_BIF_ID_STRAPS	Recommended: 0x00000000
Note: See RV610 documentation for the bit-fields of these registers.		

Finally, NB_BIF_SPARE[5] should be set. This is a work-around and may be removed in future devices. Once that is set, BIF switches into normal functional mode and the traditional PCI configuration cycles can begin.

6.7 Master Abort Status

The master abort status for BIF is now available via a sticky bit in APCCFG. The bit is reset when a '1' is written to it:

Table 6-4 Master Abort Status

Setting	Function/Comment
APC_AGP_PCI_STATUS.MASTER_ABORT	Read: 0=No master abort received 1=Master abort received Write: 0=No change 1=Reset master abort received status

6.8 HDP/MC Write Combiner

A write from the HT interface to the frame buffer follows the following path:

- HT -> IOC -> BIF -> HDP -> MC -> HT.

The IOC segments a 64B write into four 16B writes which would normally result in four 16B writes back to the CPU via the HT interface. This is undesirable due to HT inefficiencies in transferring small data sizes.

Table 6-5 Write Combiner Control Registers

Setting	Function/Comment
HDP_HOST_PATH_CNTL.WRITE_COMBINE_EN	0=Disable the write combiner in HDP 1=Enable the write combiner in HDP (2 x 16B -> 32B)
MC_ARB:RAMCFG.REQUEST_512B	0=Enable the write combiner in MC (2 x 32B -> 64B) 1=Disable the write combiner in MC

6.9 Graphics UMA FB Size

For UMA graphics the recommended UMA FB size depends on the total amount of system memory that is available (according to the values in *Table 6-6*). Some additional small performance improvements may be gained by increasing the FB to 512MB.

Table 6-6 Recommended UMAFB Size

Total System Memory (GB)	UMA Framebuffer Size (MB)
< = 512 MB	64 MB
768 MB, 1GB	128 MB
> = 1GB	256 MB

6.10 Suggested FB Interleaving Ratios

64MB of SP are recommended for performance improvements. The interleaving ratio should be set based on the system configuration, as shown in *Table 6-7* below. For other SP speeds use the closest value from *Table 6-7*.

Note: The optimal ratio is sensitive to both the benchmark and the exact system configuration. On a given system, for a given benchmark, the optimal ratio may differ slightly from the values *Table 6-7*.

- Fast UMA = HT1.8 AND 2 channels of memory
- Slow UMA = HT1 OR 1 channel of memory

Table 6-7 Suggested FB Interleaving Ratios

SP Speed	UMA Speed	Interleaving Ratio
667	Slow	5:11
667	Fast	4:12
533	Slow	4:12
533	Fast	3:13
400	Slow	3:13
400	Fast	1:7

Chapter 7
PCIE Initialization for DDI

7.1 PCIE Modes

Table 7-1 PCIE Modes

Lanes	0 to 3	4 to 7	8 to 11	12 to 15
1	GFX x16			
2	GFX x8 A			
3			GFX x8 A	
4	GFX x8 A		GFX x8 B	
5	GPP x4 A			
6		GPP x4 B		
7			GPP x4 A	
8				GPP x4 B
9	GPP x4 A	GPP x4 B		
10	GPP x4 A		GPP x4 B	
11	GPP x4 A			GPP x4 B
12		GPP x4 B	GPP x4 A	
13			GPP x4 A	GPP x4 B
14	GFX x8 A		GPP x4 B	
15	GFX x8 A			GPP x4 B
16		GPP x4 B	GFX x8 A	
17	GPP x4 A		GFX x8 B	

7.1.1 Case 1: PCIE 1x16 GFX

This is the default case and no programming is required.

7.1.2 Case 2: PCIE 1x8 GFX on Lanes 0-7

Table 7-2 TX Lane Muxing

Step	Register Settings	Function/Comment
1	PCIE_NBCFG_REGF -- NBMISCIND: 0x27 PCIE_TX_MUX_LEVEL2_2 Set bit[6] to 1	Disables PCIE mode on PHY Lanes 8-11
2	PCIE_NBCFG_REGF -- NBMISCIND: 0x27 PCIE_TX_MUX_LEVEL2_3 Set bit[7] to 1	Disables PCIE mode on PHY Lanes 12-15

7.1.3 Case 3: PCIE 1x8 GFX on Lanes 8-15

- Step 1: Clock Muxing Control
  - TXCLK

Table 7-3 TXCLK

Step	Register Settings	Function/Comment
1	PCIE_PDNB_CNTL -- NBMISCIND: 0x7 GFX_TXCLK_SEL Set bit[16] to 1	Selects PLL C to be the source of TXCLK_PERM for PCIE (bif_core)
2	PCIE_PDNB_CNTL -- NBMISCIND: 0x7 GFX_TXCLK_SND_RCV_[0-3]_SEL Set bits[15:12] to 4'hF	Selects PLL C to be the source of TXCLK_SND and TXCLK_RCV for PCIE (bif_core)
3	PCIE_PDNB_CNTL -- NBMISCIND: 0x7 IO_TXCLK_C_SEL Set bits[25:24] to 2'b10	Selects PLL C to be the source of B_PTX_DATA_CLK for the PCIE lanes (Lanes 8-15)
4	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 Reg_Turn_Off_Both_PLLs Set bits[1:0] to 2'b00	Allows the 3 PLLs to be independently powered down

  - RXCLK

Table 7-4 RXCLK

Step	Register Settings	Function/Comment
1	PCIE_NBCFG_REGF -- NBMISCIND: 0x27 PCIE_RX_MUX_SEL0 Set bit[9:8] to 2'b10	Routes RXCLK from PHY Lanes 8-11 to bif_core Lanes 0-3
2	PCIE_NBCFG_REGF -- NBMISCIND: 0x27 PCIE_RX_MUX_SEL1 Set bit[11:10] to 2'b10	Routes RXCLK from PHY Lanes 12-15 to bif_core Lanes 4-7
3	PCIE_NBCFG_REGF -- NBMISCIND: 0x27 PCIE_RX_MUX_SEL2 Set bit[13:12] to 2'b10	Routes RXCLK from PHY Lanes 0-3 to bif_core Lanes 8-11
4	PCIE_NBCFG_REGF -- NBMISCIND: 0x27 PCIE_RX_MUX_SEL3 Set bit[15:14] to 2'b10	Routes RXCLK from PHY Lanes 4-7 to bif_core Lanes 12-15

- Step 2: Lane Muxing Control
  - TX Lane Muxing

Table 7-5 TX Lane Muxing

Step	Register Settings	Function/Comment
1	PCIE_NBCFG_REGF -- NBMISCIND: 0x27 PCIE_TX_MUX_LEVEL1_2 Set bit[2] to 1	Routes TX_DATA from bif_core Lanes 0-3 to PHY Lanes 8-11
2	PCIE_NBCFG_REGF -- NBMISCIND: 0x27 PCIE_TX_MUX_LEVEL1_3 Set bit[3] to 1	Routes TX_DATA from bif_core Lanes 4-7 to PHY Lanes 12-15
3	PCIE_NBCFG_REGF -- NBMISCIND: 0x27 PCIE_TX_MUX_LEVEL2_0 Set bit[4] to 1	Disables PCIE mode on PHY Lanes 0-3
4	PCIE_NBCFG_REGF -- NBMISCIND: 0x27 PCIE_TX_MUX_LEVEL2_1 Set bit[5] to 1	Disables PCIE mode on PHY Lanes 4-7

  - RX Lane Muxing

    The RX lane muxing for RX_DATA has the same control as the RXCLK muxing, so this is already handled in Step 1.

- Step 3: Initialization Sequence for PCIE PHY

  This step is not needed for PCIE mode. The bif_core will take care of the sequence.

7.1.4 Case 4: PCIE 2x8

- Step 0: Set dual slot configuration. No extra programming is required.

Table 7-6 Dual Slot Configuration

Step	Register Settings	Function/Comment
1	PCIE_LINK_CFG - NBMISCIND:0x8 MULTIPORT_CONFIG_GFX Set bits[11:8] to 4'b0101	Sets dual slot configuration

7.1.5 Case 5: PCIE 1x4 GPP on Lanes 0-3

This is a degraded version of Case 2, PCIE 1x8 GFX on Lanes 0-7. See the programming for Case 2 (section *7.1.2*).

7.1.6 Case 6: PCIE 1x4 GPP on Lanes 4-7

This is a degraded version of Case 9, PCIE 2x4 GPPs on Lanes 0-7. See the programming for Case 9 (section *7.1.9*).

7.1.7 Case 7: PCIE 1x4 GPP on Lanes 8-11

This is a degraded version of Case 3, PCIE 1x8 GFX on Lanes 8-15. See the programming for Case 3 (section *7.1.3*).

7.1.8 Case 8: PCIE 1x4 GPP on Lanes 12-15

This is a degraded version of Case 13, PCIE 2x4 GPPs on Lanes 8-15. See the programming for Case 13 (section *7.1.13*).

7.1.9 Case 9: PCIE 2x4 GPPs on Lanes 0-7

- Step 0: Set dual slot configuration:

Table 7-7 Dual Slot Configuration

Step	Register Settings	Function/Comment
1	PCIE_LINK_CFG - NBMISCIND:0x8 MULTIPORT_CONFIG_GFX Set bits[11:8] to 4'b0101	Sets dual slot configuration

- Step 1: Clock Muxing Control
  - TXCLK

Table 7-8 TXCLK

Step	Register Settings	Function/Comment
1	PCIE_PDNB_CNTL -- NBMISCIND: 0x7 GFX_TXCLK_SEL Set bit[16] to 0	Selects PLL A to be the source of TXCLK_PERM for PCIE (bif_core)
2	PCIE_PDNB_CNTL -- NBMISCIND: 0x7 GFX_TXCLK_SND_RCV_[0-3]_SEL Set bits[15:12] to 4'h0	Selects PLL A to be the source of TXCLK_SND and TXCLK_RCV for PCIE (bif_core)
3	PCIE_PDNB_CNTL -- NBMISCIND: 0x7 IO_TXCLK_A_SEL Set bits[21:20] to 2'b00	Selects PLL A to be the source of B_PTX_DATA_CLK for the PCIE lanes (Lanes 0-7)
4	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 Reg_Turn_Off_Both_PLLs Set bit[0] to 0	Allows PLL C to be independently powered down since it is not used for PCIE in this mode

  - RXCLK

Table 7-9 RXCLK

Step	Register Settings	Function/Comment
1	PCIE_NBCFG_REGF -- NBMISCIND: 0x27 PCIE_RX_MUX_SEL0 Set bit[9:8] to 2'b00	Routes RXCLK from PHY Lanes 0-3 to bif_core Lanes 0-3
2	PCIE_NBCFG_REGF -- NBMISCIND: 0x27 PCIE_RX_MUX_SEL1 Set bit[11:10] to 2'b01	Routes RXCLK from PHY Lanes 8-11 to bif_core Lanes 4-7
3	PCIE_NBCFG_REGF -- NBMISCIND: 0x27 PCIE_RX_MUX_SEL2 Set bit[13:12] to 2'b11	Routes RXCLK from PHY Lanes 4-7 to bif_core Lanes 8-11
4	PCIE_NBCFG_REGF -- NBMISCIND: 0x27 PCIE_RX_MUX_SEL3 Set bit[15:14] to 2'b00	Routes RXCLK from PHY Lanes 12-15 to bif_core Lanes 12-15

- Step 2: Lane Muxing Control
  - TX Lane Muxing

Table 7-10 TX Lane Muxing

Step	Register Settings	Function/Comment
1	PCIE_NBCFG_REGF -- NBMISCIND: 0x27 PCIE_TX_MUX_LEVEL1_1 Set bit[1] to 1	Routes TX_DATA from bif_core Lanes 8-11 to PHY Lanes 4-7 Hardware always routes TX_DATA from bif_core Lanes 0-3 to PHY Lanes 0-3
2	PCIE_NBCFG_REGF -- NBMISCIND: 0x27 PCIE_TX_MUX_LEVEL2_2 Set bit[6] to 1	Disables PCIE mode on PHY Lanes 8-11
3	PCIE_NBCFG_REGF -- NBMISCIND: 0x27 PCIE_TX_MUX_LEVEL2_3 Set bit[7] to 1	Disables PCIE mode on PHY Lanes 12-15

  - RX Lane Muxing

    The RX lane muxing for RX_DATA has the same control as the RXCLK muxing. This is already handled in Step 1.

- Step 3: Initialization Sequence for PCIE PHY

  This step is not needed for PCIE mode. The bif_core will take care of the sequence.

7.1.10 Case 10: PCIE 1x4 GPP on Lanes 0-3 and 1x4 GPP on Lanes 8-11

This is a degraded version of Case 4, PCIE 2x8. See the programming for Case 4 (section *7.1.4*).

7.1.11 Case 11: PCIE 1x4 GPP on Lanes 0-3 and 1x4 GPP on Lanes 12-15

This is a degraded version of Case 15, PCIE 1x8 GFX on Lanes 0-7 and 1x4 GPP on Lanes 12-15. See programming for Case 15 (section *7.1.15*).

7.1.12 Case 12: PCIE 1x4 GPP on Lanes 4-7 and 1x8 GFX on Lanes 8-15

This is a degraded version of Case 16, PCIE 1x8 GFX on Lanes 8-15 and 1x4 GPP on Lanes 4-7. See the programming for Case 16 (section *7.1.16*).

7.1.13 Case 13: PCIE 2x4 GPPs on Lanes 8-15

- Step 0: Set dual slot configuration:

Table 7-11 Dual Slot Configuration

Step	Register Settings	Function/Comment
1	PCIE_LINK_CFG - NBMISCIND:0x8 MULTIPORT_CONFIG_GFX Set bits[11:8] to 4'b0101	Sets dual slot configuration

- Step 1: Clock Muxing Control
  - TXCLK

Table 7-12 TXCLK

Step	Register Settings	Function/Comment
1	PCIE_PDNB_CNTL -- NBMISCIND: 0x7 GFX_TXCLK_SEL Set bit[16] to 1	Selects PLL C to be the source of TXCLK_PERM for PCIE (bif_core)
2	PCIE_PDNB_CNTL -- NBMISCIND: 0x7 GFX_TXCLK_SND_RCV_[0-3]_SEL Set bits[15:12] to 4'hF	Selects PLL C to be the source of TXCLK_SND and TXCLK_RCV for PCIE (bif_core)
3	PCIE_PDNB_CNTL -- NBMISCIND: 0x7 IO_TXCLK_C_SEL Set bits[25:24] to 2'b10	Selects PLL C to be the source of B_PTX_DATA_CLK for the PCIE lanes (Lanes 8-15)
4	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 Reg_Turn_Off_Both_PLLs Set bit[1:0] to 2'b11	Ports A and B share PLL C

  - RXCLK

Table 7-13 RXCLK

Step	Register Settings	Function/Comment
1	PCIE_NBCFG_REGF -- NBMISCIND: 0x27 PCIE_RX_MUX_SEL0 Set bit[9:8] to 2'b10	Routes RXCLK from PHY Lanes 8-11 to bif_core Lanes 0-3
2	PCIE_NBCFG_REGF -- NBMISCIND: 0x27 PCIE_RX_MUX_SEL1 Set bit[11:10] to 2'b11	Routes RXCLK from PHY Lanes 0-3 to bif_core Lanes 4-7
3	PCIE_NBCFG_REGF -- NBMISCIND: 0x27 PCIE_RX_MUX_SEL2 Set bit[13:12] to 2'b01	Routes RXCLK from PHY Lanes 12-15 to bif_core Lanes 8-11
4	PCIE_NBCFG_REGF -- NBMISCIND: 0x27 PCIE_RX_MUX_SEL3 Set bit[15:14] to 2'b10	Routes RXCLK from PHY Lanes 4-7 to bif_core Lanes 12-15

- Step 2: Lane Muxing Control
  - TX Lane Muxing

Table 7-14 TX Lane Muxing

Step	Register Settings	Function/Comment
1	PCIE_NBCFG_REGF -- NBMISCIND: 0x27 PCIE_TX_MUX_LEVEL1_2 Set bit[2] to 1	Routes TX_DATA from bif_core Lanes 0-3 to PHY Lanes 8-11
2	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 PCIE_TX_MUX_LEVEL0 Set bit[14] to 1	Routes TX_DATA from bif_core Lanes 8-11 to PHY Lanes 12-15
3	PCIE_NBCFG_REGF -- NBMISCIND: 0x27 PCIE_TX_MUX_LEVEL2_0 Set bit[4] to 1	Disables PCIE mode on PHY Lanes 0-3
4	PCIE_NBCFG_REGF -- NBMISCIND: 0x27 PCIE_TX_MUX_LEVEL2_1 Set bit[5] to 1	Disables PCIE mode on PHY Lanes 4-7

  - RX Lane Muxing

    The RX lane muxing for RX_DATA has the same control as the RXCLK muxing. This is already handled in Step 1.

- Step 3: Initialization Sequence for PCIE PHY

  This is not needed for PCIE mode. The bif_core will take care of the sequence.

7.1.14 Case 14: PCIE 1x8 GFX on Lanes 0-7 and 1x4 GPP on Lanes 8-11

This is a degraded version of Case 4, PCIE 2x8 GFX. See the programming for Case 4 (section *7.1.4*).

7.1.15 Case 15: PCIE 1x8 GFX on Lanes 0-7 and 1x4 GPP on Lanes 12-15

- Step 0: Set dual slot configuration:

Table 7-15 Dual Slot Configuration

Step	Register Settings	Function/Comment
1	PCIE_LINK_CFG - NBMISCIND:0x8 MULTIPORT_CONFIG_GFX Set bits[11:8] to 4'b0101	Sets dual slot configuration

- Step 1: Clock Muxing Control
  - TXCLK

Table 7-16 TXCLK

Step	Register Settings	Function/Comment
1	PCIE_PDNB_CNTL -- NBMISCIND: 0x7 GFX_TXCLK_SEL Set bit[16] to 0	Selects PLL A to be the source of TXCLK_PERM for PCIE (bif_core)
2	PCIE_PDNB_CNTL -- NBMISCIND: 0x7 GFX_TXCLK_SND_RCV_[0-3]_SEL Set bits[15:12] to 4'h0	Selects PLL A to be the source of TXCLK_SND and TXCLK_RCV for PCIE (bif_core)
3	PCIE_PDNB_CNTL -- NBMISCIND: 0x7 IO_TXCLK_A_SEL Set bits[21:20] to 2'b00	Selects PLL A to be the source of B_PTX_DATA_CLK for the PCIE lanes (Lanes 8-15)
4	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 Reg_Turn_Off_Both_PLLs Set bits[1:0] to 2'b01	Allows PLL C to be independently powered down since PLLA is used to drive TXCLK_PERM

  - RXCLK

Table 7-17 RXCLK

Step	Register Settings	Function/Comment
1	PCIE_NBCFG_REGF -- NBMISCIND: 0x27 PCIE_RX_MUX_SEL0 Set bit[9:8] to 2'b00	Routes RXCLK from PHY Lanes 0-3 to bif_core Lanes 0-3
2	PCIE_NBCFG_REGF -- NBMISCIND: 0x27 PCIE_RX_MUX_SEL1 Set bit[11:10] to 2'b00	Routes RXCLK from PHY Lanes 4-7 to bif_core Lanes 4-7
3	PCIE_NBCFG_REGF -- NBMISCIND: 0x27 PCIE_RX_MUX_SEL2 Set bit[13:12] to 2'b01	Routes RXCLK from PHY Lanes 12-15 to bif_core Lanes 8-11
4	PCIE_NBCFG_REGF -- NBMISCIND: 0x27 PCIE_RX_MUX_SEL3 Set bit[15:14] to 2'b11	Routes RXCLK from PHY Lanes 8-11 to bif_core Lanes 12-15

- Step 2: Lane Muxing Control
  - TX Lane Muxing

Table 7-18 TX Lane Muxing

Step	Register Settings	Function/Comment
1	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 PCIE_TX_MUX_LEVEL0 Set bit[14] to 1	Routes TX_DATA from bif_core Lanes 8-11 to PHY Lanes 12-15
2	PCIE_NBCFG_REGF -- NBMISCIND: 0x27 PCIE_TX_MUX_LEVEL2_2 Set bit[6] to 1	Disables PCIE mode on PHY Lanes 8-11

  - RX Lane Muxing

    The RX lane muxing for RX_DATA has the same control as the RXCLK muxing. This is already handled in Step 1.

- Step 3: Initialization Sequence for PCIE PHY

  This is not needed for PCIE mode. The bif_core will take care of the sequence.

7.1.16 Case 16: PCIE 1x8 GFX on Lanes 8-15 and 1x4 GPP on Lanes 4-7

- Step 0: Set dual slot configuration:

Table 7-19 Dual Slot Configuration

Step	Register Settings	Function/Comment
1	PCIE_LINK_CFG - NBMISCIND:0x8 MULTIPORT_CONFIG_GFX Set bits[11:8] to 4'b0101	Sets dual slot configuration

- Step 1: Clock Muxing Control
  - TXCLK

Table 7-20 TXCLK

Step	Register Settings	Function/Comment
1	PCIE_PDNB_CNTL -- NBMISCIND: 0x7 GFX_TXCLK_SEL Set bit[16] to 1	Selects PLL C to be the source of TXCLK_PERM for PCIE (bif_core)
2	PCIE_PDNB_CNTL -- NBMISCIND: 0x7 GFX_TXCLK_SND_RCV_[0-3]_SEL Set bits[15:12] to 4'hF	Selects PLL C to be the source of TXCLK_SND and TXCLK_RCV for PCIE (bif_core)
3	PCIE_PDNB_CNTL -- NBMISCIND: 0x7 IO_TXCLK_C_SEL Set bits[25:24] to 2'b10	Selects PLL C to be the source of B_PTX_DATA_CLK for the PCIE lanes (Lanes 8-15)
4	PCIE_PDNB_CNTL -- NBMISCIND: 0x7 IO_TXCLK_B_SEL Set bits[23:22] to 2'b10	Selects PLL C to be the source of B_PTX_DATA_CLK for the PCIE lanes (Lanes 4-7)
5	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 Reg_Turn_Off_Both_PLLs Set bits[1:0] to 2'b10	Allows the PLL A and PLL B to be independently powered down since only PLL C is used for PCIE TXCLK_PERM

- RXCLK

Table 7-21 RXCLK

Step	Register Settings	Function/Comment
1	PCIE_NBCFG_REGF -- NBMISCIND: 0x27 PCIE_RX_MUX_SEL0 Set bit[9:8] to 2'b10	Routes RXCLK from PHY Lanes 8-11 to bif_core Lanes 0-3
2	PCIE_NBCFG_REGF -- NBMISCIND: 0x27 PCIE_RX_MUX_SEL1 Set bit[11:10] to 2'b10	Routes RXCLK from PHY Lanes 12-15 to bif_core Lanes 4-7
3	PCIE_NBCFG_REGF -- NBMISCIND: 0x27 PCIE_RX_MUX_SEL2 Set bit[13:12] to 2'b11	Routes RXCLK from PHY Lanes 4-7 to bif_core Lanes 8-11
4	PCIE_NBCFG_REGF -- NBMISCIND: 0x27 PCIE_RX_MUX_SEL3 Set bit[15:14] to 2'b01	Routes RXCLK from PHY Lanes 0-3 to bif_core Lanes 12-15

- Step 2: Lane Muxing Control
  - TX Lane Muxing

Table 7-22 TX Lane Muxing

Step	Register Settings	Function/Comment
1	PCIE_NBCFG_REGF -- NBMISCIND: 0x27 PCIE_TX_MUX_LEVEL1_2 Set bit[2] to 1	Routes TX_DATA from bif_core Lanes 0-3 to PHY Lanes 8-11
2	PCIE_NBCFG_REGF -- NBMISCIND: 0x27 PCIE_TX_MUX_LEVEL1_3 Set bit[3] to 1	Routes TX_DATA from bif_core Lanes 4-7 to PHY Lanes 12-15
3	PCIE_NBCFG_REGF -- NBMISCIND: 0x27 PCIE_TX_MUX_LEVEL1_1 Set bit[1] to 1	Routes TX_DATA from bif_core Lanes 8-11 to PHY Lanes 4-7
4	PCIE_NBCFG_REGF -- NBMISCIND: 0x27 PCIE_TX_MUX_LEVEL2_0 Set bit[4] to 1	Disables PCIE mode on PHY Lanes 0-3

  - RX Lane Muxing

    The RX lane muxing for RX_DATA has the same control as the RXCLK muxing. This is already handled in Step 1.

- Step 3: Initialization Sequence for PCIE PHY

  This is not needed for PCIE mode. The bif_core will take care of the sequence.

7.1.17 Case 17: PCIE 1x4 GPP on Lanes 0-3 and 1x8 GFX on Lanes 8-15

This is a degraded version of Case 4, PCIE 2x8 GFX. See the programming in Case 4 (section *7.1.4*).

7.2 DDI Modes

Table 7-23 PCIE Modes

Lanes	0 to 3	4 to 7	8 to 11	12 to 15
1	DDI_SL			
2		DDI_SL		
3			DDI_SL	
4				DDI_SL
5	DDI_SL	DDI_SL		
6	DDI_DL			
7	DDI_SL		DDI_SL	
8	DDI_SL			DDI_SL
9		DDI_SL	DDI_SL	
10		DDI_SL		DDI_SL
11			DDI_DL	
12			DDI_DL Mirror	

There are 3 different clock modes for TMDS/SDVO:

- Coherent Mode. In this mode, the clock is embedded in the data stream sent from display to PCIE; therefore, no special programming is required.
- PHY Incoherent Mode. In this mode, the display clock input from the display PLL is sent to the Tx as the incoherent clock.
- Core Incoherent Mode. This is the backup incoherent mode, where the incoherent clock is sent though the core logic to B_PTX_TEST_DATA of the PHY.

7.2.1 DDI Programming Sequence

The DDI programming sequence can be divided into 3 parts:

- Part 1: Power On Sequence

  To power on the PLL according to the lanes being used program the following:

Table 7-24 Power On Sequence - PLL

Register Settings	Function/Comment
PCIE_NBCFG_REG16 -- NBMISCIND: 0x2E B_PPLL_PDNB_FDIS_GFX_A Set bit[4] to 0	Power on PLL A for Lanes 0-3
PCIE_NBCFG_REG16 -- NBMISCIND: 0x2E B_PPLL_PDNB_FDIS_GFX_B Set bit[5] to 0	Power on PLL B for Lanes 4-7
PCIE_NBCFG_REG16 -- NBMISCIND: 0x2E B_PPLL_PDNB_FDIS_GFX_C Set bit[6] to 0	Power on PLL C for Lanes 8-15

To power on the TX lanes program the following:

Table 7-25 Power On Sequence - TX Lanes

PCIE_NBCFG_REG17 - NBMICSCIND:0x2F	Function/Comment
B_PTX_PDNB_0 Set bit[16] to 1	Power on TX lane 0
B_PTX_PDNB_1 Set bit[17] to 1	Power on TX lane 1
B_PTX_PDNB_2 Set bit[18] to 1	Power on TX lane 2
B_PTX_PDNB_3 Set bit[19] to 1	Power on TX lane 3
B_PTX_PDNB_4 Set bit[20] to 1	Power on TX lane 4
B_PTX_PDNB_5 Set bit[21] to 1	Power on TX lane 5
B_PTX_PDNB_6 Set bit[22] to 1	Power on TX lane 6
B_PTX_PDNB_7 Set bit[23] to 1	Power on TX lane 7
B_PTX_PDNB_8 Set bit[24] to 1	Power on TX lane 8
B_PTX_PDNB_9 Set bit[25] to 1	Power on TX lane 9
B_PTX_PDNB_10 Set bit[26] to 1	Power on TX lane 10
B_PTX_PDNB_11 Set bit[27] to 1	Power on TX lane 11
B_PTX_PDNB_12 Set bit[28] to 1	Power on TX lane 12
B_PTX_PDNB_13 Set bit[29] to 1	Power on TX lane 13
B_PTX_PDNB_14 Set bit[30] to 1	Power on TX lane 14
B_PTX_PDNB_15 Set bit[31] to 1	Power on TX lane 15

- Part 2: Initialization Sequence

  The initialization sequence is broken down into a case by case basis below.

- Part 3: Power Down Sequence

  Before undocking, the PLL should be powered down in order to save power.

  To power down the PLL according to the lanes being used program the following:

Table 7-26 Power Down Sequence - PLL

Register Settings	Function/Comment
PCIE_NBCFG_REG16 -- NBMISCIND: 0x2E B_PPLL_PDNB_FDIS_GFX_A Set bit[4] to 1	Power down PLL A for Lanes 0-3
PCIE_NBCFG_REG16 -- NBMISCIND: 0x2E B_PPLL_PDNB_FDIS_GFX_B Set bit[5] to 1	Power down PLL B for Lanes 4-7
PCIE_NBCFG_REG16 -- NBMISCIND: 0x2E B_PPLL_PDNB_FDIS_GFX_C Set bit[6] to 1	Power down PLL C for Lanes 8-15

Before undocking, the TX lanes should be disabled and powered down in order to save power.

To disable the TX lanes program the following:

Table 7-27 Power Down Sequence - TX Lanes

Register Settings	Function/Comment
PCIE_NBCFG_REG11 -- NBMISCIND: 0x29 REG_B_PTX_EN_A Set bit[4] to 0	Disables TX Lanes 0-3
PCIE_NBCFG_REG12 -- NBMISCIND: 0x2A REG_B_PTX_EN_B Set bit[4] to 0	Disables TX Lanes 4-7
PCIE_NBCFG_REG13 -- NBMISCIND: 0x2B REG_B_PTX_EN_C Set bit[4] to 0	Disables TX Lanes 8-15

Note: To power down the TX lanes use the information in *Table 7-25* for B_PTX_PDNB in the power on sequence above. Set the corresponding bit to 0 to power down a specific lane. The same table can be used to power down unused lanes in Display Port configuration.

7.2.2 Initialization Sequence

The following information in *Figure 7-1* specifies all the different frequencies that are being supported. It will be referenced in the programming sequence.

Standard	Bit Rate		PLL Reference Clock Freq		B_PPLL_CLKR[4:0] (PLL reference divide ratio)		B_PPLL_CLKF[6:0] (PLL feedback divide ratio)		B_PG2PLL_CLKFRAC[2:0] (PLL feedback fractional divide ratio)		B_P90PLL_IBIAS[7:0]	B_PTX_CLK_DIV[2:0]	B_PRX_CLK_DIV[1:0]	B_PG2PLL_1X_CLK_DIV[2:0]	Tx Divide Ratio		Rx Divide Ratio		VCO Frequency Required		B_PPLL_CLKP[2:0] (PLL post divide ratio)		PLL Output Frequency Required		P_B1X Output Divide	P_B1X Frequency [MHz]		TMDS Incoherent Clock Frequency (B_P90TX_TEST_DATA[0])		TMDS Incoherent to Ref Clock Ratio	DPLL Incoherent Clk Post Divider	B_PG2PLL_VCO_MODE	Bit rate check	
	Low Range (Gb/s)	High Range (Gb/s)	Low Range (MHz)	High Range (MHz)	Ctl Value (Bin)	Divide Value (Dec)	(Hex)	(Dec)	(Bin)	(Dec)	(Hex)	(Bin)	(Bin)	(Bin)	Front End (Dec)	Digital (Dec)	Front End (Dec)	Digital (Dec)	Low Range (GH	High Range (GH	Ctl Value (Bin)	Divide Value	Low Range (GHz	High Range (GHz	(Dec)	Low Range (MHz	High Range (MHz	Low Range (MHz	High Range (MHz	(Dec)	(Dec)	(bin)	Low	Hi
PCIe Gen2	5	5	100	100	0xxxx	1	0x32	50			0xC6	000	00	000	1	2	1	2					5	5	1	500	500	x	x	x	x	0	OK	OK
PCIe Gen1	2.5	2.5	100	100	0xxxx	1	0x32	50			0xC6	001	01	000	2	4	2	4					5	5	1	500	500	x	x	x	x	0	OK	OK
PCIe Gen1	2.5	2.5	100	100	0xxxx	1	0x19	25			0xAD	000	00	000	1	2	1	2					2.5	2.5	1	250	250	x	x	x	x	1	OK	OK
SDVO(Tx)	2	2.5	100	125	0xxxx	1	0x14	20			0x8C	000	xx	000	1	2	x	x					2	2.5	1	200	250	x	x	x	x	1	OK	OK
	1.65	2	82.5	100	0xxxx	1	0x14	20			0x5B	000	xx	000	1	2	x	x					1.65	2	1	165	200	x	x	x	x	1	OK	OK
	1.25	1.65	62.5	82.5	0xxxx	1	0x14	20			0x4B	000	xx	000	1	2	x	x					1.25	1.65	1	125	165	x	x	x	x	1	OK	OK
	1	1.25	100	125	0xxxx	1	0x14	20			0x8C	001	xx	001	2	4	x	x					2	2.5	2	100	125	x	x	x	x	1	OK	OK
TMDS (Tx)	1.25	1.65	125	165	10000	2	0x14	20			0x19	000	xx	000	1	2	x	x					1.25	1.65	1	125	165	125	165	1	1	1	OK	OK
	0.625	1.25	62.5	125	0xxxx	1	0x14	20			0x19	001	xx	001	2	4	x	x					1.25	2.5	2	62.5	125	62.5	125	1	1	1	OK	OK
	0.3125	0.625	62.5	125	0xxxx	1	0x14	20			0x19	010	xx	010	4	8	x	x					1.25	2.5	4	31.3	63	31.25	62.5	2	2	1	OK	OK
	0.25	0.3125	50	62.5	0xxxx	1	0x32	50			0x3A	100	xx	100	10	20	x	x					2.5	3.13	10	25	31	25	31.25	2	2	0	OK	OK
DP (Tx)	2.7	2.7	270	270	10000	2	0x14	20			0x84	000	00	000	1	2	1	2					2.7	2.7	1	270	270	x	x	x	1	1	OK	OK
	2.7	2.7	100	100	0xxxx	1	0x1B	27			0x94	000	00	000	1	2	1	2					2.7	2.7	1	270	270	x	x	x	1	1	OK	OK
	1.62	1.62	270	270	10000	2	0x0C	12			0x2B	000	00	000	1	2	1	2					1.62	1.62	1	162	162	x	x	x	1	1	OK	OK
	1.62	1.62	162	162	10000	2	0x14	20			0x5B	000	00	000	1	2	1	2					1.62	1.62	1	162	162	x	x	x	1	1	OK	OK
HT3	5.2	5.2	100	100	0xxxx	1	0x34	52			0xC6	000	00	000	1	2	1	2					5.2	5.2	1	520	520	x	x	x	x	0	OK	OK
GDDR5	4	5	100	100	0xxxx	1	0x28-32	40-50	000-111	0-0.875	0x9D-0xC6	000	00	000	1	2	1	2	4	5	000	1	4	5	1	400	500	x	x	x	x	0	OK	OK
	3.3	4	100	100	0xxxx	1	0x21-28	33-40	000-111	0-0.875	0x6C-0x9D	000	00	000	1	2	1	2	3.3	4	000	1	3.3	4	1	330	400	x	x	x	x	0	OK	OK
	2.5	3.3	100	100	0xxxx	1	0x19-21	25-33	000-111	0-0.875	0x2B-0x6C	000	00	000	1	2	1	2	2.5	3.3	000	1	2.5	3.3	1	250	330	x	x	x	x	0	OK	OK
	1.6	2.5	100	100	0xxxx	1	0x10-19	16-25	000-111	0-0.875	0x5B-0x8C	000	00	000	1	2	1	2	1.6	2.5	000	1	1.6	2.5	1	160	250	x	x	x	x	1	OK	OK
	1.2	1.6	100	100	0xxxx	1	0x0C-10	12-16	000-111	0-0.875	0x4B-0x5B	000	00	000	1	2	1	2	1.2	1.6	000	1	1.2	1.6	1	120	160	x	x	x	x	1	OK	OK
Note:																																			
(1) For other freq in HT3 mode, please use the same setting as in GDDR5 mode except the fractional part.																																			
(2) For freq lower than 1.2Gin GDDR5 mode, use post divider.																																			

Figure 7-1 Supported Frequencies

7.2.2.1 Case 1: DDI_SL on Lanes 0-3

- Step 0a: To enable PHY incoherent mode program the following:

Table 7-28 PHY Incoherent Mode

Step	Register Settings	Function/Comment
1	PCIE_NBCFG_REG11 -- NBMISCIND: 0x29 REG_B_PTX_BYPASS_EN_A Set bit[28] to 1	Asserts bypass enable
2	PCIE_NBCFG_REG7 -- NBMISCIND: 0x37 B_P90PLL_BACKUP_A Set bits[31:29] to 3'b100	Selects display clock to be the incoherent clock

- Note: For frequencies below 62.5MHz, set the following register:

Table 7-29 Settings for Frequencies Below 62.5MHz

Step	Register Settings	Function/Comment
1	PCIE_NBCFG_REG11 -- NBMISCIND: 0x29 REG_B_PG2PLL_TMDS_MODE_A Set bit[27] to 1	Divides the incoherent clock by 2

- Step 0b: To enable Core incoherent mode program the following:

Table 7-30 Core Incoherent Mode

Step	Register Settings	Function/Comment
1	PCIE_NBCFG_REG11 -- NBMISCIND: 0x29 REG_B_PTX_BYPASS_EN_A Set bit[28] to 1	Asserts bypass enable
2	PCIE_NBCFG_REG7 -- NBMISCIND: 0x37 B_P90PLL_BACKUP_A Set bits[31:29] to 3'b000	Selects B_P90TX_TEST_DATA[0] to be the incoherent clock
3	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 SDVO_CLK_lane3_SEL Set bit[24] to 0	Selects B_PG2PLL_DISPLAY_CLK[0] to be B_P90TX_TEST_DATA[0]

- Step 1: Clock Muxing Control
  - TXCLK

Table 7-31 TXCLK

Step	Register Settings	Function/Comment
1	PCIE_PDNB_CNTL -- NBMISCIND: 0x7 IO_TXCLK_A_SEL Set bits[21:20] to 2'b00	Selects PLL A to be the source of B_PTX_DATA_CLK for Lanes 0-3
2	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 Reg_Turn_Off_Both_PLLs Set bit[0] to 0	Allows PLL A to be powered down using register control

  - RXCLK

    There is no receiver data path in DDI mode, so RXCLK is not required.

- Step 2: Lane Muxing Control
  - TX Lane Muxing

Table 7-32 TX Lane Muxing

Step	Register Settings	Function/Comment
1	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 PCIE_DDI_MUX_LEVEL0_1 Set bit[16] to 0	Selects channel 1 data from display
2	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 PCIE_DDI_MUX_LEVEL1_0 Set bit[18] to 0	Passes channel 1 data to Lanes 0-3

3	PCIE_NBCFG_REGF -- NBMISCIND: 0x27 PCIE_TX_MUX_LEVEL2_0 Set bit[4] to 1	Selects DDI traffic for PHY Lanes 0-3

- RX Lane Muxing

  There is no receiver data path in DDI mode, so RX lane muxing is not required.

- Step 3: Initialization Sequence for PCIE PHY (only required for serial simulations)

Table 7-33 PCIE PHY Initialization Sequence

Step	Register Settings	Function/Comment
1	PCIE_NBCFG_REG11 -- NBMISCIND: 0x29 REG_B_PTX_EN_A Set bit[4] to 0	Deasserts TX_EN to shut down the transmitter for Lanes 0-3
2	PCIE_NBCFG_REG11 -- NBMISCIND: 0x29 REG_B_PPLL_PDNB_A Set bit[26] to 0	Asserts PLL_PDNB to shut down PLL A
3	Wait for at least 320ns	
4	PCIE_NBCFG_REGE -- NBMISCIND: 0x26 P_BDISPLAY_CLK_PDNB[0] Set bit[22] to 1	Powers on the display clock branch 0 in PCIE PLL
5a	PCIE_NBCFG_REGE -- NBMISCIND: 0x26 B_PG2PLL_VCO_MODE_A Set bit[26] according to frequency table	Controls the VCO frequency mode
5b	PCIE_NBCFG_REG17 -- NBMISCIND: 0x2F REG_B_P90PLL_CLKR_A Set bits[4:0] according to the frequency table	Sets reference clock divider
5c	PCIE_NBCFG_REG11 -- NBMISCIND: 0x29 REG_B_P90PLL_CLKF_A Set bits[13:7] according to the frequency table	Sets PLL feedback divide ratio
6	PCIE_NBCFG_REG11 -- NBMISCIND: 0x29 REG_B_P90PLL_IBIAS_A Set bits[25:16] according to frequency table	Sets PLL bandwidth control
7	PCIE_NBCFG_REGF -- NBMISCIND: 0x27 REG_B_PG2TX_CLK_DIV_A Set bits[24:22] according to frequency table	Sets TXCLK divider
8	PCIE_NBCFG_REG11 -- NBMISCIND: 0x29 REG_B_PG2RX_CLK_DIV_A Set bits[15:14] according to frequency table	Sets RXCLK divider
9a	PCIE_NBCFG_REG11 -- NBMISCIND: 0x29 REG_B_PG2PLL_1X_CLK_DIV_A Set bits[31:29] according to frequency table	Sets second divider on P_B1X
9b	PCIE_NBCFG_REGE -- NBMISCIND: 0x26 B_PG2PLL_VCO_MODE_A Set bit[26] according to frequency table	Controls VCO frequency mode
10	PCIE_NBCFG_REG10 -- NBMISCIND: 0x29 B_PREFCLK_SEL_A Set bits[7:6] to 2'b10	Selects B_PG2PLL_DISPLAY_CLK[0] to be the REFCLK for PLL A

11	Wait for at least 320ns	
12	PCIE_NBCFG_REG11 -- NBMISCIND: 0x29 REG_B_PPLL_PDNB_A Set bit[26] to 1	Deasserts PLL_PDNB to power on PLL A
13	Wait for at least 30us	Takes 30us for PLL to lock
14	PCIE_NBCFG_REG11 -- NBMISCIND: 0x29 REG_B_PTX_CM_HIGHI_A[0] Set bit[0] to 1	Asserts B_PTX_CM_HIGHI for Lane 0
15	Wait for at least 4ns	
16	PCIE_NBCFG_REG11 -- NBMISCIND: 0x29 REG_B_PTX_CM_HIGHI_A[1] Set bit[1] to 1	Asserts B_PTX_CM_HIGHI for Lane 1
17	Wait for at least 4ns	
18	PCIE_NBCFG_REG11 -- NBMISCIND: 0x29 REG_B_PTX_CM_HIGHI_A[2] Set bit[2] to 1	Asserts B_PTX_CM_HIGHI for Lane 2
19	Wait for at least 4ns	
20	PCIE_NBCFG_REG11 -- NBMISCIND: 0x29 REG_B_PTX_CM_HIGHI_A[3] Set bit[3] to 1	Asserts B_PTX_CM_HIGHI for Lane 3
21	Wait for at least 36ns	
22	PCIE_NBCFG_REG11 -- NBMISCIND: 0x29 REG_B_PTX_EN_A Set bit[4] to 1	Asserts TX_EN to power on the transmitter for Lanes 0-3

- Step 4: Clock Selection for Display FIFO in PCIE

Table 7-34 Clock Selection for Display FIFO in PCIE

Step	Register Settings	Function/Comment
1	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 DISP_LINK_CLK_SEL[0] Set bit[12] to 1	Selects iDISP_LINK_CLK[0] to be the write clock of FIFO 1
2	PCIE_PDNB_CNTL – NBMISCIND: 0x7 DISP_FIFO_RCLK0_SEL Set bits[28:26] to 3'b100	Selects the TXCLK output from PLL A to be the read clock of FIFO 1

- Step 5: Reset Display FIFO in PCIE

Table 7-35 Reset Display FIFO in PCIE

Step	Register Settings	Function/Comment
1	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 WRESET_FIFO1 RRESET_FIFO1 Set bit[3:2] to 2'b11	Asserts write and read resets for FIFO1
2	Wait for at least 100ns	
3	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 WRESET_FIFO1 RRESET_FIFO1 Set bit[3:2] to 2'b00	De-asserts resets for FIFO1

- Interrupt Detection Check. Make sure /top/ig/upcie_top/INT0_DET_RX_ASSERTED is 1

7.2.2.2 Case 2: DDI_SL on Lanes 4-7

- Step 0a: To enable PHY incoherent mode program the following:

Table 7-36 PHY Incoherent Mode

Step	Register Settings	Function/Comment
1	PCIE_NBCFG_REG12 -- NBMISCIND: 0x2A REG_B_PTX_BYPASS_EN_B Set bit[28] to 1	Asserts bypass enable
2	PCIE_NBCFG_REGE -- NBMISCIND: 0x26 B_P90PLL_BACKUP_B Set bits[30:28] to 3'b100	Selects display clock to be the incoherent clock

- Note: For frequencies below 62.5MHz set the following register:

Table 7-37 Settings For Frequencies Below 62.5MHz

Step	Register Settings	Function/Comment
1	PCIE_NBCFG_REG12 -- NBMISCIND: 0x2A REG_B_PG2PLL_TMDS_MODE_B Set bit[27] to 1	Divides the incoherent clock by 2

- Step 0b: To enable Core incoherent mode program the following:

Table 7-38 Core Incoherent Mode

Step	Register Settings	Function/Comment
1	PCIE_NBCFG_REG12 -- NBMISCIND: 0x2A REG_B_PTX_BYPASS_EN_B Set bit[28] to 1	Asserts bypass enable
2	PCIE_NBCFG_REGE -- NBMISCIND: 0x26 B_P90PLL_BACKUP_B Set bits[30:28] to 3'b000	Selects B_P90TX_TEST_DATA[0] to be the incoherent clock
3	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 SDVO_CLK_lane7_SEL Set bit[25] to 0	Selects PCIE_SDVO_CLK_LINK1 to be B_P90TX_TEST_DATA[0]

- Note: If in Case 5, add the following step:

Table 7-39 Core Incoherent Mode

Step	Register Settings	Function/Comment
4	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 SDVO_CLK_lane7_SEL Set bit[25] to 1	Selects PCIE_SDVO_CLK_LINK2 to be B_P90TX_TEST_DATA[0]

- Step 1: Clock Muxing Control
  - TXCLK

Table 7-40 TXCLK

Step	Register Settings	Function/Comment
1	PCIE_PDNB_CNTL -- NBMISCIND: 0x7 IO_TXCLK_B_SEL Set bits[23:22] to 2'b11	Selects PLL B to be the source of B_PTX_DATA_CLK for Lanes 4-7

2	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 Reg_Turn_Off_Both_PLLs Set bit[0] to 0	Allows PLL B to be powered down using register control

- RXCLK

  There is no receiver data path in DDI mode, so RXCLK is not required.

- Step 2: Lane Muxing Control
  - TX Lane Muxing

Table 7-41 TX Lane Muxing

Step	Register Settings	Function/Comment
1	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 PCIE_DDI_MUX_LEVEL0_1 Set bit[16] to 0	Selects channel 1 data from display
2	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 PCIE_DDI_MUX_LEVEL1_1 Set bit[19] to 1	Passes channel 1 data to Lanes 4-7
3	PCIE_NBCFG_REGF -- NBMISCIND: 0x27 PCIE_TX_MUX_LEVEL2_1 Set bit[5] to 1	Selects DDI traffic for PHY Lanes 4-7

- RX Lane Muxing

  There is no receiver data path in DDI mode, so RX lane muxing is not required.

- Step 3: Initialization Sequence for PCIE PHY (only required for serial simulations)

Table 7-42 PCIE PHY Initialization Sequence

Step	Register Settings	Function/Comment
1	PCIE_NBCFG_REG12 -- NBMISCIND: 0x2A REG_B_PTX_EN_B Set bit[4] to 0	De-asserts TX_EN to shut down the transmitter for Lanes 4-7
2	PCIE_NBCFG_REG12 -- NBMISCIND: 0x2A REG_B_PPLL_PDNB_B Set bit[26] to 0	Asserts PLL_PDNB to shut down PLL B
3	Wait for at least 320ns	
4	PCIE_NBCFG_REGE -- NBMISCIND: 0x26 P_BDISPLAY_CLK_PDNB[0] Set bit[22] to 1	Powers on the display clock branch 0 in PCIE PLL.
5a	PCIE_NBCFG_REGE -- NBMISCIND: 0X26 B_PG2PLL_VCO_MODE_B Set bit[25] according to frequency table	Controls VCO frequency mode
5b	PCIE_NBCFG_REG17 -- NBMISCIND: 0X2F REG_B_P90PLL_CLKR_B Set bits[13:7] according to frequency table	Sets reference clock divider
5c	PCIE_NBCFG_REG12 -- NBMISCIND: 0x2A REG_B_P90PLL_CLKF_B Set bits[13:7] according to frequency table	Sets PLL feedback divide ratio

6	PCIE_NBCFG_REG12 -- NBMISCIND: 0x2A REG_B_P90PLL_IBIAS_B Set bits[25:16] according to frequency table	Sets PLL bandwidth control
7	PCIE_NBCFG_REGF -- NBMISCIND: 0x27 REG_B_PG2TX_CLK_DIV_B Set bits[27:25] according to frequency table	Sets TXCLK divider
8	PCIE_NBCFG_REG12 -- NBMISCIND: 0x2A REG_B_PG2RX_CLK_DIV_B Set bits[15:14] according to frequency table	Sets RXCLK divider
9a	PCIE_NBCFG_REG12 -- NBMISCIND: 0x2A REG_B_PG2PLL_1X_CLK_DIV_B Set bits[31:29] according to frequency table	Sets second divider on P_B1X
9b	PCIE_NBCFG_REGE -- NBMISCIND: 0x26 B_PG2PLL_VCO_MODE_B Set bit[25] according to frequency table	Controls VCO frequency mode
10	PCIE_NBCFG_REG10 -- NBMISCIND: 0x29 B_PREFCLK_SEL_B Set bits[9:8] to 2'b10 If in Case 5: Set bits[9:8] to 2'b11	Selects B_PG2PLL_DISPLAY_CLK[0] to be the REFCLK for PLL B Selects B_PG2PLL_DISPLAY_CLK[0] to be the REFCLK for PLL B
11	Wait for at least 320ns	
12	PCIE_NBCFG_REG12 -- NBMISCIND: 0x2A REG_B_PPLL_PDNB_B Set bit[26] to 1	Deasserts PLL_PDNB to power on PLL B
13	Wait for at least 30us	Takes 30us for PLL to lock
14	PCIE_NBCFG_REG12 -- NBMISCIND: 0x2A REG_B_PTX_CM_HIGHI_B[0] Set bit[0] to 1	Asserts B_PTX_CM_HIGHI for Lane 4
15	Wait for at least 4ns	
16	PCIE_NBCFG_REG12 -- NBMISCIND: 0x2A REG_B_PTX_CM_HIGHI_B[1] Set bit[1] to 1	Asserts B_PTX_CM_HIGHI for Lane 5
17	Wait for at least 4ns	
18	PCIE_NBCFG_REG12 -- NBMISCIND: 0x2A REG_B_PTX_CM_HIGHI_B[2] Set bit[2] to 1	Asserts B_PTX_CM_HIGHI for Lane 6
19	Wait for at least 4ns	
20	PCIE_NBCFG_REG12 -- NBMISCIND: 0x2A REG_B_PTX_CM_HIGHI_B[3] Set bit[3] to 1	Asserts B_PTX_CM_HIGHI for Lane 7
21	Wait for at least 36ns	
22	PCIE_NBCFG_REG12 -- NBMISCIND: 0x2A REG_B_PTX_EN_B Set bit[4] to 1	Asserts TX_EN to power on the transmitter for Lanes 4-7

- Step 4: Clock Selection for Display FIFO in PCIE

Table 7-43 Clock Selection for Display FIFO in PCIE

Step	Register Settings	Function/Comment
1	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 DISP_LINK_CLK_SEL[0] Set bit[12] to 1	Selects iDISP_LINK_CLK[0] to be the write clock of FIFO 1
2	PCIE_PDNB_CNTL – NBMISCIND: 0x7 DISP_FIFO_RCLK0_SEL Set bits[28:26] to 3'b010	Selects the TXCLK output from PLL B to be the read clock of FIFO 1

- Step 5: Reset Display FIFO in PCIE

Table 7-44 Reset Display FIFO in PCIE

Step	Register Settings	Function/Comment
1	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 WRESET_FIFO1 RRESET_FIFO1 Set bit[3:2] to 2'b11	Asserts write and read resets for FIFO1
2	Wait for at least 100ns	
3	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 WRESET_FIFO1 RRESET_FIFO1 Set bit[3:2] to 2'b00	De-asserts resets for FIFO1

- Interrupt Detection Check: Make sure /top/ig/upcie_top/INT0_DET_RX_ASSERTED is 1

7.2.2.3 Case 3: DDI_SL on Lanes 8-11

- Step 0a: To enable PHY incoherent mode program the following:

Table 7-45 PHY Incoherent Mode

Step	Register Settings	Function/Comment
1	PCIE_NBCFG_REG13 -- NBMISCIND: 0x2B REG_B_PTX_BYPASS_EN_C Set bit[28] to 1	Asserts bypass enable
2	PCIE_NBCFG_REGB -- NBMISCIND: 0x23 B_P90PLL_BACKUP_C Set bits[16:14] to 3'b100	Selects display clock to be the incoherent clock

- Note: For frequencies below 62.5MHz, set the following register:

Table 7-46 Settings For Frequencies Below 62.5MHz

Step	Register Settings	Function/Comment
1	PCIE_NBCFG_REG12 -- NBMISCIND: 0x2B REG_B_PG2PLL_TMDS_MODE_C Set bit[27] to 1	Divides the incoherent clock by 2

- Step 0b: To enable Core incoherent mode program the following:

Table 7-47 Core Incoherent Mode

Step	Register Settings	Function/Comment
1	PCIE_NBCFG_REG13 -- NBMISCIND: 0x2B REG_B_PTX_BYPASS_EN_C Set bit[28] to 1	Asserts bypass enable
2	PCIE_NBCFG_REGE -- NBMISCIND: 0x26 B_P90PLL_BACKUP_C Set bits[16:14] to 3'b000	Selects B_P90TX_TEST_DATA[0] to be the incoherent clock
3	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 SDVO_CLK_lane11_SEL Set bit[26] to 0	Selects PCIE_SDVO_CLK_LINK1 to be B_P90TX_TEST_DATA[0]

- Note: If in Case 7 or in Case 9, add the following step:

Table 7-48 Core Incoherent Mode

Step	Register Settings	Function/Comment
4	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 SDVO_CLK_lane11_SEL Set bit[26] to 1	Selects PCIE_SDVO_CLK_LINK2 to be B_P90TX_TEST_DATA[0]

- Step 1: Clock Muxing Control
  - TXCLK

Table 7-49 TXCLK

Step	Register Settings	Function/Comment
1	PCIE_PDNB_CNTL -- NBMISCIND: 0x7 IO_TXCLK_C_SEL Set bits[25:24] to 2'b10	Selects PLL C to be the source of B_PTX_DATA_CLK for Lanes 8-11
2	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 Reg_Turn_Off_Both_PLLs Set bit[1] to 0	Allows PLL C to be powered down using register control

  - RXCLK

    There is no receiver data path in DDI mode, so RXCLK is not required.

- Step 2: Lane Muxing Control
  - TX Lane Muxing

Table 7-50 TX Lane Muxing

Step	Register Settings	Function/Comment
1	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 PCIE_DDI_MUX_LEVEL0_1 Set bit[16] to 0	Selects channel 1 data from display
2	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 PCIE_DDI_MUX_LEVEL1_2 Set bit[20] to 1	Passes channel 1 data to Lanes 8-11

3	PCIE_NBCFG_REGF -- NBMISCIND: 0x27 PCIE_TX_MUX_LEVEL2_2 Set bit[6] to 1	Selects DDI traffic for PHY Lanes 8-11
4	PCIE_NBCFG_REGF -- NBMISCIND: 0x27 PCIE_TX_MUX_LEVEL2_3 Set bit[7] to 1	Disables PCIE traffic for PHY Lanes 12-15 since PLL is shared with Lanes 8-11

- RX Lane Muxing

  There is no receiver data path in DDI mode, so RX lane muxing is not required.

- Step 3: Initialization Sequence for PCIE PHY (only required for serial simulations)

Table 7-51 PCIE PHY Initialization Sequence

Step	Register Settings	Function/Comment
1	PCIE_NBCFG_REG13 -- NBMISCIND: 0x2B REG_B_PTX_EN_C Set bit[4] to 0	Deasserts TX_EN to shut down the transmitter for Lanes 8-11
2	PCIE_NBCFG_REG13 -- NBMISCIND: 0x2B REG_B_PPLL_PDNB_C Set bit[26] to 0	Asserts PLL_PDNB to shut down PLL C
3	Wait for at least 320ns	
4	PCIE_NBCFG_REGE -- NBMISCIND: 0x26 If in Case 3: P_BDISPLAY_CLK_PDNB[0] Set bit[22] to 1 If in Case 7 or in Case 9: P_BDISPLAY_CLK_PDNB[1] Set bit[23] to 1	 Powers on the display clock branch 0 in PCIE PLL Powers on the display clock branch 1 in PCIE PLL
5a	PCIE_NBCFG_REGE -- NBMISCIND: 0x26 B_PG2PLL_VCO_MODE_C Set bit[24] according to frequency table	Controls VCO frequency mode
5b	PCIE_NBCFG_REG17 -- NBMISCIND: 0x2F REG_B_P90PLL_CLKR_C Set bits[14:10] according to frequency table	Sets reference clock divider
5c	PCIE_NBCFG_REG13 -- NBMISCIND: 0x2B REG_B_P90PLL_CLKF_C Set bits[13:7] according to frequency table	Sets PLL feedback divide ratio
6	PCIE_NBCFG_REG13 -- NBMISCIND: 0x2B REG_B_P90PLL_IBIAS_C Set bits[25:16] according to frequency table	Sets PLL bandwidth control
7	PCIE_NBCFG_REGF -- NBMISCIND: 0x27 REG_B_PG2TX_CLK_DIV_C Set bits[30:28] according to frequency table	Sets TXCLK divider
8	PCIE_NBCFG_REG13 -- NBMISCIND: 0x2B REG_B_PG2RX_CLK_DIV_C Set bits[15:14] according to frequency table	Sets RXCLK divider

9a	PCIE_NBCFG_REG13 -- NBMISCIND: 0x2B REG_B_PG2PLL_1X_CLK_DIV_C Set bits[31:29] according to frequency table	Sets second divider on P_B1X
9b	PCIE_NBCFG_REGE -- NBMISCIND: 0x26 B_PG2PLL_VCO_MODE_C Set bit[24] according to frequency table	Controls VCO frequency mode
10	PCIE_NBCFG_REG10 -- NBMISCIND: 0x29 B_PREFCLK_SEL_B Set bits[11:10] to 2'b10 If in Case 7 or in Case 9: Set bits[11:10] to 2'b11	Selects B_PG2PLL_DISPLAY_CLK[0] to be the REFCLK for PLL C Selects B_PG2PLL_DISPLAY_CLK[1] to be the REFCLK for PLL C
11	Wait for at least 320ns	
12	PCIE_NBCFG_REG13 -- NBMISCIND: 0x2B REG_B_PPLL_PDNB_C Set bit[26] to 1	Deasserts PLL_PDNB to power on PLL C
13	Wait for at least 30us	Waits for PLL to lock
14	PCIE_NBCFG_REG13 -- NBMISCIND: 0x2B REG_B_PTX_CM_HIGHI_C[0] Set bit[0] to 1	Asserts B_PTX_CM_HIGHI for Lane 8
15	Wait for at least 4ns	
16	PCIE_NBCFG_REG13 -- NBMISCIND: 0x2B REG_B_PTX_CM_HIGHI_C[1] Set bit[1] to 1	Asserts B_PTX_CM_HIGHI for Lane 9
17	Wait for at least 4ns	
18	PCIE_NBCFG_REG13 -- NBMISCIND: 0x2B REG_B_PTX_CM_HIGHI_C[2] Set bit[2] to 1	Asserts B_PTX_CM_HIGHI for Lane 10
19	Wait for at least 4ns	
20	PCIE_NBCFG_REG13 -- NBMISCIND: 0x2B REG_B_PTX_CM_HIGHI_C[3] Set bit[3] to 1	Asserts B_PTX_CM_HIGHI for Lane 11
21	Wait for at least 36ns	
22	PCIE_NBCFG_REG13 -- NBMISCIND: 0x2B REG_B_PTX_EN_C Set bit[4] to 1	Asserts TX_EN to power on the transmitter for Lanes 8-11

- Step 4: Clock Selection for Display FIFO in PCIE

Table 7-52 Clock Selection for Display FIFO in PCIE

Step	Register Settings	Function/Comment
1	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 DISP_LINK_CLK_SEL[0] Set bit[12] to 1	Selects iDISP_LINK_CLK[0] to be the write clock of FIFO 1
2	PCIE_PDNB_CNTL – NBMISCIND: 0x7 DISP_FIFO_RCLK0_SEL Set bits[28:26] to 3'b001	Selects the TXCLK output from PLL C to be the read clock of FIFO 1

- Step 5: Reset Display FIFO in PCIE

Table 7-53 Reset Display FIFO in PCIE

Step	Register Settings	Function/Comment
1	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 WRESET_FIFO1 RRESET_FIFO1 Set bit[3:2] to 2'b11	Asserts write and read resets for FIFO1
2	Wait for at least 100ns	
3	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 WRESET_FIFO1 RRESET_FIFO1 Set bit[3:2] to 2'b00	De-asserts resets for FIFO1

- Interrupt Detection Check: Make sure /top/ig/upcie_top/INT0_DET_RX_ASSERTED is 1

7.2.2.4 Case 4: DDI_SL on Lanes 12-15

- Step 0a: To enable PHY incoherent mode program the following:

Table 7-54 PHY Incoherent Mode

Step	Register Settings	Function/Comment
1	PCIE_NBCFG_REG14 -- NBMISCIND: 0x2C REG_B_PTX_BYPASS_EN_D Set bit[29] to 1	Asserts bypass enable
2	PCIE_NBCFG_REGB -- NBMISCIND: 0x23 B_P90PLL_BACKUP_C Set bits[16:14] to 3'b100	Selects display clock to be the incoherent clock

- Note: For frequencies below 62.5MHz, set the following register:

Table 7-55 Settings For Frequencies Below 62.5MHz

Step	Register Settings	Function/Comment
1	PCIE_NBCFG_REG12 -- NBMISCIND: 0x2B REG_B_PG2PLL_TMDS_MODE_C Set bit[27] to 1	Divides the incoherent clock by 2

- Step 0b: To enable Core incoherent mode program the following:

Table 7-56 Core Incoherent Mode

Step	Register Settings	Function/Comment
1	PCIE_NBCFG_REG14 -- NBMISCIND: 0x2C REG_B_PTX_BYPASS_EN_D Set bit[29] to 1	Asserts bypass enable
2	PCIE_NBCFG_REGE -- NBMISCIND: 0x26 B_P90PLL_BACKUP_C Set bits[16:14] to 3'b000	Selects B_P90TX_TEST_DATA[0] to be the incoherent clock
3	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 SDVO_CLK_lane12_SEL Set bit[27] to 0	Selects PCIE_SDVO_CLK_LINK1 to be B_P90TX_TEST_DATA[0]

- Note: If in Case 8 or in Case 10, add the following step:

Table 7-57 Core Incoherent Mode

Step	Register Settings	Function/Comment
3	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 SDVO_CLK_lane12_SEL Set bit[27] to 1	Selects PCIE_SDVO_CLK_LINK2 to be B_P90TX_TEST_DATA[0]

- Step 1: Clock Muxing Control
  - TXCLK

Table 7-58 TXCLK

Step	Register Settings	Function/Comment
1	PCIE_PDNB_CNTL -- NBMISCIND: 0x7 IO_TXCLK_C_SEL Set bits[25:24] to 2'b10	Selects PLL C to be the source of B_PTX_DATA_CLK for Lanes 8-11
2	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 Reg_Turn_Off_Both_PLLs Set bit[1] to 0	Allows PLL C to be powered down using register control

  - RXCLK

    There is no receiver data path in DDI mode, so RXCLK is not required.

- Step 2: Lane Muxing Control
  - For COHERENT mode: TX Lane Muxing

Table 7-59 TX Lane Muxing (COHERENT Mode)

Step	Register Settings	Function/Comment
1	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 PCIE_DDI_MUX_LEVEL0_1 Set bit[16] to 0	Selects channel 1 data from display
2	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 PCIE_DDI_MUX_LEVEL1_3 Set bit[21] to 1	Passes channel 1 data to Lanes 12-15
3	PCIE_NBCFG_REGF -- NBMISCIND: 0x27 PCIE_TX_MUX_LEVEL2_3 Set bit[7] to 1	Selects DDI traffic for PHY Lanes 12-15
4	PCIE_NBCFG_REGF -- NBMISCIND: 0x27 PCIE_TX_MUX_LEVEL2_2 Set bit[6] to 1	Disables PCIE traffic for PHY Lanes 8-11

  - For INCOHERENT mode: TX Lane Muxing for Mirroring. In this case, incoherent mode is only supported through mirroring because the distribution of the incoherent clock is only available on Lane 12 (Lane 4 of the x8 macro). Therefore, special TX lane muxing is required for mirroring.

Table 7-60 TX Lane Muxing (INCOHERENT Mode)

Step	Register Settings	Function/Comment
1	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 PCIE_DDI_MUX_LEVEL0_2 Set bit[17] to 1	Selects the mirrored version of channel 1 data from display

2	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 PCIE_DDI_MUX_LEVEL1_3 Set bit[21] to 0	Passes the mirrored version of channel 1 data to Lanes 12-15
3	PCIE_NBCFG_REGF -- NBMISCIND: 0x27 PCIE_TX_MUX_LEVEL2_3 Set bit[7] to 1	Selects DDI traffic for PHY Lanes 12-15
4	PCIE_NBCFG_REGF -- NBMISCIND: 0x27 PCIE_TX_MUX_LEVEL2_2 Set bit[6] to 1	Disables PCIE traffic for PHY Lanes 8-11

- RX Lane Muxing

  There is no receiver data path in DDI mode, so RX lane muxing is not required.

- Step 3: Initialization Sequence for PCIE PHY (only required for serial simulations)

Table 7-61 PCIE PHY Initialization Sequence

Step	Register Settings	Function/Comment
1	PCIE_NBCFG_REG13 -- NBMISCIND: 0x2B REG_B_PTX_EN_C Set bit[4] to 0	Deasserts TX_EN to shut down the transmitter for Lanes 12-15
2	PCIE_NBCFG_REG13 -- NBMISCIND: 0x2B REG_B_PPLL_PDNB_C Set bit[26] to 0	Asserts PLL_PDNB to shut down PLL C
3	Wait for at least 320ns	
4	PCIE_NBCFG_REGE -- NBMISCIND: 0x26 If in Case 3: P_BDISPLAY_CLK_PDNB[0] Set bit[22] to 1 If in Case 8 or in Case 10: P_BDISPLAY_CLK_PDNB[1] Set bit[23] to 1	 Powers on the display clock branch 0 in PCIE PLL Powers on the display clock branch 1 in PCIE PLL.
5a	PCIE_NBCFG_REGE -- NBMISCIND: 0x26 B_PG2PLL_VCO_MODE_C Set bit[24] according to frequency table	Controls VCO frequency mode
5b	PCIE_NBCFG_REG17 -- NBMISCIND: 0x2F REG_B_P90PLL_CLKR_C Set bits[14:10] according to frequency table	Sets reference clock divider
5c	PCIE_NBCFG_REG13 -- NBMISCIND: 0x2B REG_B_P90PLL_CLKF_C Set bits[13:7] according to frequency table	Sets PLL feedback divide ratio
6	PCIE_NBCFG_REG13 -- NBMISCIND: 0x2B REG_B_P90PLL_IBIAS_C Set bits[25:16] according to frequency table	Sets PLL bandwidth control
7	PCIE_NBCFG_REGF -- NBMISCIND: 0x27 REG_B_PG2TX_CLK_DIV_C Set bits[30:28] according to frequency table	Sets TXCLK divider

8	PCIE_NBCFG_REG13 -- NBMISCIND: 0x2B REG_B_PG2RX_CLK_DIV_C Set bits[15:14] according to frequency table	Sets RXCLK divider
9a	PCIE_NBCFG_REG13 -- NBMISCIND: 0x2B REG_B_PG2PLL_1X_CLK_DIV_C Set bits[31:29] according to frequency table	Sets second divider on P_B1X
9b	PCIE_NBCFG_REGE -- NBMISCIND: 0x26 B_PG2PLL_VCO_MODE_C Set bit[24] according to frequency table	Controls VCO frequency mode
10	PCIE_NBCFG_REG10 -- NBMISCIND: 0x29 B_PREFCLK_SEL_C Set bits[11:10] to 2'b10 If in Case 8 or Case 10: Set bits[11:10] to 2'b11	Selects B_PG2PLL_DISPLAY_CLK[0] to be the REFCLK for PLL C Selects B_PG2PLL_DISPLAY_CLK[1] to be the REFCLK for PLL C
11	Wait for at least 320ns	
12	PCIE_NBCFG_REG13 -- NBMISCIND: 0x2B REG_B_PPLL_PDNB_C Set bit[26] to 1	Deasserts PLL_PDNB to power on PLL C
13	Wait for at least 30us	Waits for PLL to lock
14	PCIE_NBCFG_REG13 -- NBMISCIND: 0x2B REG_B_PTX_CM_HIGHI_C[0] Set bit[0] to 1	Asserts B_PTX_CM_HIGHI for Lane 12
15	Wait for at least 4ns	
16	PCIE_NBCFG_REG13 -- NBMISCIND: 0x2B REG_B_PTX_CM_HIGHI_C[1] Set bit[1] to 1	Asserts B_PTX_CM_HIGHI for Lane 13
17	Wait for at least 4ns	
18	PCIE_NBCFG_REG13 -- NBMISCIND: 0x2B REG_B_PTX_CM_HIGHI_C[2] Set bit[2] to 1	Asserts B_PTX_CM_HIGHI for Lane 14
19	Wait for at least 4ns	
20	PCIE_NBCFG_REG13 -- NBMISCIND: 0x2B REG_B_PTX_CM_HIGHI_C[3] Set bit[3] to 1	Asserts B_PTX_CM_HIGHI for Lane 15
21	Wait for at least 36ns	
22	PCIE_NBCFG_REG13 -- NBMISCIND: 0x2B REG_B_PTX_EN_C Set bit[4] to 1	Asserts TX_EN to power on the transmitter for Lanes 12-15

- Step 4: Clock Selection for Display FIFO in PCIE

Table 7-62 Clock Selection for Display FIFO in PCIE

Step	Register Settings	Function/Comment
1	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 DISP_LINK_CLK_SEL[0] Set bit[12] to 1	Selects iDISP_LINK_CLK[0] to be the write clock of FIFO 1
2	PCIE_PDNB_CNTL – NBMISCIND: 0x7 DISP_FIFO_RCLK0_SEL Set bits[28:26] to 3'b001	Selects the TXCLK output from PLL C to be the read clock of FIFO 1

- Step 5: Reset Display FIFO in PCIE

Table 7-63 Reset Display FIFO in PCIE

Step	Register Settings	Function/Comment
1	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 WRESET_FIFO1 RRESET_FIFO1 Set bit[3:2] to 2'b11	Asserts write and read resets for FIFO1
2	Wait for at least 100ns	
3	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 WRESET_FIFO1 RRESET_FIFO1 Set bit[3:2] to 2'b00	De-asserts resets for FIFO1

- Interrupt Detection Check: Make sure /top/ig/upcie_top/INT0_DET_RX_ASSERTED is 1

7.2.2.5 Case 5: DDI_SL on Lanes 0-3 + DDL_SL on Lanes 4-7

If incoherent mode is enabled for DDI_SL on Lanes 0-3, see incoherent mode programming in Case 1 (section *7.2.2.1*).

If incoherent mode is enabled for DDI_SL on Lanes 4-7, see incoherent mode programming in Case 2 (section *7.2.2.2*).

- To initialize DDI on Lanes 0-3
  - Step 1: Clock Muxing Control
    - TXCLK

Table 7-64 TXCLK

Step	Register Settings	Function/Comment
1	PCIE_PDNB_CNTL -- NBMISCIND: 0x7 IO_TXCLK_A_SEL Set bits[21:20] to 2'b00	Selects PLL A to be the source of B_PTX_DATA_CLK for Lanes 0-3
2	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 Reg_Turn_Off_Both_PLLs Set bit[0] to 0	Allows PLL A and PLL B to be powered down using register control

    - RXCLK

      There is no receiver data path in DDI mode, so RXCLK is not required.

- Step 2: Lane Muxing Control
  - TX Lane Muxing

Table 7-65 TX Lane Muxing

Step	Register Settings	Function/Comment
1	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 PCIE_DDI_MUX_LEVEL0_1 Set bit[16] to 0	Selects channel 1 data from display
2	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 PCIE_DDI_MUX_LEVEL1_0 Set bit[18] to 0	Passes channel 1 data to Lanes 0-3
3	PCIE_NBCFG_REGF -- NBMISCIND: 0x27 PCIE_TX_MUX_LEVEL2_0 Set bit[4] to 1	Selects DDI traffic for PHY Lanes 0-3

  - RX Lane Muxing

    There is no receiver data path in DDI mode, so RX lane muxing is not required.

- Step 3: Initialization Sequence for PCIE PHY (only required for serial simulations)

  Execute Step 3 in Case 1 (section *7.2.2.1*), DDI_SL on Lanes 0-3.

- Step 4: Clock Selection for Display FIFO in PCIE

Table 7-66 Clock Selection for Display FIFO in PCIE

Step	Register Settings	Function/Comment
1	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 DISP_LINK_CLK_SEL[0] Set bit[12] to 1	Selects iDISP_LINK_CLK[0] to be the write clock of FIFO 1
2	PCIE_PDNB_CNTL – NBMISCIND: 0x7 DISP_FIFO_RCLK0_SEL Set bits[28:26] to 3'b100	Selects the TXCLK output from PLL A to be the read clock of FIFO 1

- Step 5: Reset Display FIFO in PCIE

Table 7-67 Reset Display FIFO in PCIE

Step	Register Settings	Function/Comment
1	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 WRESET_FIFO1, RRESET_FIFO1 Set bit[3:2] to 2'b11	Asserts write and read resets for FIFO1
2	Wait for at least 100ns	
3	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 WRESET_FIFO1, RRESET_FIFO1 WRESET_FIFO2, RRESET_FIFO2 Set bit[3:2] to 2'b0	De-asserts resets for FIFO1

- To initialize DDI on Lanes 4-7
  - Step 1: Clock Muxing Control
    - TXCLK

Table 7-68 TXCLK

Step	Register Settings	Function/Comment
1	PCIE_PDNB_CNTL -- NBMISCIND: 0x7 IO_TXCLK_B_SEL Set bits[23:22] to 2'b11	Selects PLL B to be the source of B_PTX_DATA_CLK for Lanes 4-7
2	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 Reg_Turn_Off_Both_PLLs Set bit[0] to 0	Allows PLL A and PLL B to be powered down using register control

    - RXCLK

      There is no receiver data path in DDI mode, so RXCLK is not required.
  - Step 2: Lane Muxing Control
    - TX Lane Muxing

Table 7-69 TX Lane Muxing

Step	Register Settings	Function/Comment
1	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 PCIE_DDI_MUX_LEVEL0_2 Set bit[17] to 0	Selects channel 2 data from display
2	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 PCIE_DDI_MUX_LEVEL1_1 Set bit[19] to 0	Passes channel 2 data to Lanes 4-7
3	PCIE_NBCFG_REGF -- NBMISCIND: 0x27 PCIE_TX_MUX_LEVEL2_1 Set bit[5] to 1	Selects DDI traffic for PHY Lanes 4-7

    - RX Lane Muxing

      There is no receiver data path in DDI mode, so RX lane muxing is not required.
  - Step 3: Initialization Sequence for PCIE PHY (only required for serial simulations)

    Execute Step 3 in Case 2 (section *7.2.2.2*), DDI_SL on Lanes 4-7.
  - Step 4: Clock Selection for Display FIFO in PCIE

Table 7-70 TX Clock Selection for Display FIFO in PCIE

Step	Register Settings	Function/Comment
1	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 DISP_LINK_CLK_SEL[1] Set bit[13] to 1	Selects iDISP_LINK_CLK[1] to be the write clock of FIFO 2
3	PCIE_PDNB_CNTL – NBMISCIND: 0x7 DISP_FIFO_RCLK1_SEL Set bits[31:29] to 3'b010	Selects the TXCLK output from PLL B to be the read clock of FIFO 2

- Step 5: Reset Display FIFO in PCIE

Table 7-71 Reset Display FIFO in PCIE

Step	Register Settings	Function/Comment
1	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 WRESET_FIFO2, RRESET_FIFO2 Set bit[5:4] to 2'b11	Asserts write and read resets for FIFO2
2	Wait for at least 100ns	
3	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 WRESET_FIFO2, RRESET_FIFO2 Set bit[5:4] to 2'b0	De-asserts resets for FIFO2

- Interrupt Detection Check:
  - Make sure /top/ig/upcie_top/INT0_DET_RX_ASSERTED is 1
  - Make sure /top/ig/upcie_top/INT1_DET_RX_ASSERTED is 1

7.2.2.6 Case 6: DDI_DL on Lanes 0-7

If incoherent mode is enabled, refer to incoherent mode programming in Case 1 (section *7.2.2.1*), DDI_SL on Lanes 0-3.

- Step 1: Clock Muxing Control
  - TXCLK

Table 7-72 TXCLK

Step	Register Settings	Function/Comment
1	PCIE_PDNB_CNTL -- NBMISCIND: 0x7 IO_TXCLK_A_SEL Set bits[21:20] to 2'b00	Selects PLL A to be the source of B_PTX_DATA_CLK for Lanes 0-3
2	PCIE_PDNB_CNTL -- NBMISCIND: 0x7 IO_TXCLK_B_SEL Set bits[23:22] to 2'b00	Selects PLL A to be the source of B_PTX_DATA_CLK for Lanes 4-7
3	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 Reg_Turn_Off_Both_PLLs Set bit[0] to 0	Allows PLL A and PLL B to be powered down using register control

  - RXCLK

    There is no receiver data path in DDI mode, so RXCLK is not required.

- Step 2: Lane Muxing Control
  - TX Lane Muxing

Table 7-73 TX Lane Muxing

Step	Register Settings	Function/Comment
1	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 PCIE_DISP_FIFO_CfgDualLink Set bit[23] to 1	Enables dual link mode
2	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 PCIE_DDI_MUX_LEVEL0_1 Set bit[16] to 0	Selects channel 1 data from display
3	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 PCIE_DDI_MUX_LEVEL0_2 Set bit[17] to 0	Selects channel 2 data from display

4	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 PCIE_DDI_MUX_LEVEL1_0 Set bit[18] to 0	Passes channel 1 data to Lanes 0-3
5	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 PCIE_DDI_MUX_LEVEL1_1 Set bit[19] to 0	Passes channel 2 data to Lanes 4-7
6	PCIE_NBCFG_REGF -- NBMISCIND: 0x27 PCIE_TX_MUX_LEVEL2_0 Set bit[4] to 1	Selects DDI traffic for PHY Lanes 0-3
7	PCIE_NBCFG_REGF -- NBMISCIND: 0x27 PCIE_TX_MUX_LEVEL2_1 Set bit[5] to 1	Selects DDI traffic for PHY Lanes 4-7

- RX Lane Muxing

  There is no receiver data path in DDI mode, so RX lane muxing is not required.

- Step 3: Initialization Sequence for PCIE PHY (only required for serial simulations)

Table 7-74 PCIE PHY Initialization Sequence

Step	Register Settings	Function/Comment
1	PCIE_NBCFG_REG11 -- NBMISCIND: 0x29 REG_B_PTX_EN_A Set bit[4] to 0	Deasserts TX_EN to shut down the transmitter for Lanes 0-7
2	PCIE_NBCFG_REG11 -- NBMISCIND: 0x29 REG_B_PPLL_PDNB_A Set bit[26] to 0	Asserts PLL_PDNB to shut down PLL A
3	PCIE_NBCFG_REG12 -- NBMISCIND: 0x2A REG_B_PPLL_PDNB_B Set bit[26] to 0	Asserts PLL_PDNB to shut down PLL B
4	Wait for at least 320ns	
5a	PCIE_NBCFG_REGE -- NBMISCIND: 0x26 P_BDISPLAY_CLK_PDNB[1:0] Set bits[23:22] to 2'b11	Powers on the display clock branch 0 and branch 1 in PCIE PLL
5b	PCIE_NBCFG_REGE -- NBMISCIND: 0x26 B_PG2PLL_VCO_MODE_A Set b[26] according to frequency table	Controls VCO frequency mode
5c	PCIE_NBCFG_REG17 -- NBMISCIND: 0x2F REG_B_P90PLL_CLKR_A Set bits[4:0] according to frequency table	Sets reference clock divider
6	PCIE_NBCFG_REG11 -- NBMISCIND: 0x29 REG_B_P90PLL_CLKF_A Set bits[13:7] according to frequency table	Sets PLL feedback divide ratio
7	PCIE_NBCFG_REG11 -- NBMISCIND: 0x29 REG_B_P90PLL_IBIAS_A Set bits[25:16] according to frequency table	Sets PLL bandwidth control
8	PCIE_NBCFG_REGF -- NBMISCIND: 0x27 REG_B_PG2TX_CLK_DIV_A Set bits[24:22] according to frequency table	Sets TXCLK divider

9a	PCIE_NBCFG_REG11 -- NBMISCIND: 0x29 REG_B_PG2RX_CLK_DIV_A Set bits[15:14] according to frequency table	Sets RXCLK divider
9b	PCIE_NBCFG_REGE -- NBMISCIND: 0x26 B_PG2PLL_VCO_MODE_A Set bit[26] according to frequency table	Controls VCO frequency mode
10	PCIE_NBCFG_REG11 -- NBMISCIND: 0x29 REG_B_PG2PLL_1X_CLK_DIV_A Set bits[31:29] according to frequency table	Sets second divider on P_B1X
11	PCIE_NBCFG_REG10 -- NBMISCIND: 0x29 B_PREFCLK_SEL_A Set bits[7:6] to 2'b10	Selects B_PG2PLL_DISPLAY_CLK[0] to be the REFCLK for PLL A
12a	PCIE_NBCFG_REGE -- NBMISCIND: 0x26 B_PG2PLL_VCO_MODE_B Set bit[25] according to frequency table	Controls VCO frequency mode
12b	PCIE_NBCFG_REG17 -- NBMISCIND: 0x2F REG_B_P90PLL_CLKR_B Set bits[9:5] according to frequency table	Sets reference clock divider
13	PCIE_NBCFG_REG12 -- NBMISCIND: 0x2A REG_B_P90PLL_CLKF_B Set bits[13:7] according to frequency table	Sets PLL feedback divide ratio
14	PCIE_NBCFG_REG12 -- NBMISCIND: 0x2A REG_B_P90PLL_IBIAS_B Set bits[25:16] according to frequency table	Sets PLL bandwidth control
15	PCIE_NBCFG_REGF -- NBMISCIND: 0x27 REG_B_PG2TX_CLK_DIV_B Set bits[27:25] according to frequency table	Sets TXCLK divider
16	PCIE_NBCFG_REG12 -- NBMISCIND: 0x2A REG_B_PG2RX_CLK_DIV_B Set bits[15:14] according to frequency table	Sets RXCLK divider
17	PCIE_NBCFG_REG12 -- NBMISCIND: 0x2A REG_B_PG2PLL_1X_CLK_DIV_B Set bits[31:29] according to frequency table	Sets second divider on P_B1X
18	PCIE_NBCFG_REG10 -- NBMISCIND: 0x29 B_PREFCLK_SEL_B Set bits[9:8] to 2'b10	Selects B_PG2PLL_DISPLAY_CLK[0] to be the REFCLK for PLL B
19	Wait for at least 320ns	
20	PCIE_NBCFG_REG11 -- NBMISCIND: 0x29 REG_B_PPLL_PDNB_A Set bit[26] to 1	Deasserts PLL_PDNB to power on PLL A
21	Wait for at least 30us	Waits for PLL to lock
22	PCIE_NBCFG_REG12 -- NBMISCIND: 0x2A REG_B_PPLL_PDNB_B Set bit[26] to 1	Deasserts PLL_PDNB to power on PLL B

23	PCIE_NBCFG_REG11 -- NBMISCIND: 0x29 REG_B_PTX_CM_HIGHI_A[0] Set bit[0] to 1	Asserts B_PTX_CM_HIGHI for Lane 0, 4
24	Wait for at least 4ns	
25	PCIE_NBCFG_REG11 -- NBMISCIND: 0x29 REG_B_PTX_CM_HIGHI_A[1] Set bit[1] to 1	Asserts B_PTX_CM_HIGHI for Lane 1, 5
26	Wait for at least 4ns	
27	PCIE_NBCFG_REG11 -- NBMISCIND: 0x29 REG_B_PTX_CM_HIGHI_A[2] Set bit[2] to 1	Asserts B_PTX_CM_HIGHI for Lane 2, 6
28	Wait for at least 4ns	
29	PCIE_NBCFG_REG11 -- NBMISCIND: 0x29 REG_B_PTX_CM_HIGHI_A[3] Set bit[3] to 1	Asserts B_PTX_CM_HIGHI for Lane 3, 7
30	Wait for at least 36ns	
31	PCIE_NBCFG_REG11 -- NBMISCIND: 0x29 REG_B_PTX_EN_A Set bit[4] to 1	Asserts TX_EN to power on the transmitter for Lanes 0-3

- Step 4: Clock Selection for Display FIFO in PCIE

Table 7-75 Clock Selection for Display FIFO in PCIE

Step	Register Settings	Function/Comment
1	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 DISP_LINK_CLK_SEL Set bit[13:12] to 2'b11	Selects iDISP_LINK_CLK[0] and iDISP_LINK_CLK[1] to be the write clock of FIFO 1 and FIFO 2 respectively
2	PCIE_PDNB_CNTL – NBMISCIND: 0x7 DISP_FIFO_RCLK0_SEL Set bits[28:26] to 3'b100	Selects the TXCLK output from PLL A to be the read clock of FIFO 1
3	PCIE_PDNB_CNTL – NBMISCIND: 0x7 DISP_FIFO_RCLK1_SEL Set bits[31:29] to 3'b100	Selects the TXCLK output from PLL A to be the read clock of FIFO 2

- Step 5: Reset Display FIFO in PCIE

Table 7-76 Reset Display FIFO in PCIE

Step	Register Settings	Function/Comment
1	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 WRESET_FIFO1, RRESET_FIFO1 WRESET_FIFO2, RRESET_FIFO2 Set bit[5:2] to 4'b1111	Asserts write and read resets for FIFO1 and FIFO2
2	Wait for at least 100ns	
3	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 WRESET_FIFO1, RRESET_FIFO1 WRESET_FIFO2, RRESET_FIFO2 Set bit[5:2] to 4'b0	De-asserts resets for FIFO1 and FIFO2

- Interrupt Detection Check:
  - Make sure /top/ig/upcie_top/INT0_DET_RX_ASSERTED is 1
  - Make sure /top/ig/upcie_top/INT1_DET_RX_ASSERTED is 1

7.2.2.7 Case 7: DDI_SL on Lanes 0-3 + DDL_SL on Lanes 8-11

If incoherent mode is enabled for DDI_SL on Lanes 0-3, see incoherent mode programming in Case 1 (section *7.2.2.1*).

If incoherent mode is enabled for DDI_SL on Lanes 8-11, see incoherent mode programming in Case 3 (section *7.2.2.3*).

- To initialize DDI on Lanes 0-3

  See Case 5 (section *7.2.2.5*).
- To initialize DDI on Lanes 8-11
  - Step 1: Clock Muxing Control
    - TXCLK

Table 7-77 TXCLK

Step	Register Settings	Function/Comment
1	PCIE_PDNB_CNTL -- NBMISCIND: 0x7 IO_TXCLK_C_SEL Set bits[25:24] to 2'b10	Selects PLL C to be the source of B_PTX_DATA_CLK for Lanes 8-11
2	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 Reg_Turn_Off_Both_PLLs Set bit[1:0] to 2'00	Allows PLL A to be powered down using register control

    - RXCLK

      There is no receiver data path in DDI mode, so RXCLK is not required.
  - Step 2: Lane Muxing Control
    - TX Lane Muxing

Table 7-78 TX Lane Muxing

Step	Register Settings	Function/Comment
1	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 PCIE_DDI_MUX_LEVEL0_2 Set bit[17] to 0	Selects channel 2 data from display
2	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 PCIE_DDI_MUX_LEVEL1_2 Set bit[20] to 0	Passes channel 2 data to Lanes 8-11
3	PCIE_NBCFG_REGF -- NBMISCIND: 0x27 PCIE_TX_MUX_LEVEL2_2 Set bit[6] to 1	Selects DDI traffic for PHY Lanes 8-11
4	PCIE_NBCFG_REGF -- NBMISCIND: 0x27 PCIE_TX_MUX_LEVEL2_3 Set bit[7] to 1	Disables PCIE traffic for PHY Lanes 12-15 since PLL is shared with Lanes 8-11

    - RX Lane Muxing

      There is no receiver data path in DDI mode, so RX lane muxing is not required.
  - Step 3: Initialization Sequence for PCIE PHY (only required for serial simulations)

    Execute Step 3 in Case 3 (section *7.2.2.3*), DDI_SL on Lanes 8-11.

- Step 4: Clock Selection for Display FIFO in PCIE

Table 7-79 Clock Selection for Display FIFO in PCIE

Step	Register Settings	Function/Comment
1	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 DISP_LINK_CLK_SEL Set bit[13] to 1	Selects iDISP_LINK_CLK[1] to be the write clock of FIFO 2
2	PCIE_PDNB_CNTL – NBMISCIND: 0x7 DISP_FIFO_RCLK1_SEL Set bits[31:29] to 3'b001	Selects the TXCLK output from PLL C to be the read clock of FIFO 2

- Step 5: Reset Display FIFO in PCIE

Table 7-80 Reset Display FIFO in PCIE

Step	Register Settings	Function/Comment
1	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 WRESET_FIFO2, RRESET_FIFO2 Set bit[5:4] to 2'b11	Asserts write and read resets for FIFO2
2	Wait for at least 100ns	
3	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 WRESET_FIFO2, RRESET_FIFO2 Set bit[5:4] to 2'b0	De-asserts resets for FIFO2

  - Interrupt Detection Check:
    - Make sure /top/ig/upcie_top/INT0_DET_RX_ASSERTED is 1
    - Make sure /top/ig/upcie_top/INT1_DET_RX_ASSERTED is 1

7.2.2.8 Case 8: DDI_SL on Lanes 0-3 + DDL_SL on Lanes 12-15

If incoherent mode is enabled for DDI_SL on Lanes 0-3, see incoherent mode programming in Case 1 (section *7.2.2.1*).

Incoherent mode for DDL_SL on Lanes 12-15 is not supported in this case.

- To initialize DDI on Lanes 0-3

  See Case 5 (section *7.2.2.5*).
- To initialize DDI on Lanes 12-15
  - Step 1: Clock Muxing Control
    - TXCLK

Table 7-81 TXCLK

Step	Register Settings	Function/Comment
1	PCIE_PDNB_CNTL -- NBMISCIND: 0x7 IO_TXCLK_C_SEL Set bits[25:24] to 2'b10	Selects PLL C to be the source of B_PTX_DATA_CLK for Lanes 12-15
2	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 Reg_Turn_Off_Both_PLLs Set bit[1:0] to 2'00	Allows PLL A to be powered down using register control

    - RXCLK

      There is no receiver data path in DDI mode, so RXCLK is not required.

- Step 2: Lane Muxing Control
  - TX Lane Muxing

Table 7-82 TX Lane Muxing

Step	Register Settings	Function/Comment
1	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 PCIE_DDI_MUX_LEVEL0_2 Set bit[17] to 0	Selects channel 2 data from display
2	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 PCIE_DDI_MUX_LEVEL1_3 Set bit[21] to 0	Passes channel 2 data to Lanes 12-15
3	PCIE_NBCFG_REGF -- NBMISCIND: 0x27 PCIE_TX_MUX_LEVEL2_3 Set bit[7] to 1	Selects DDI traffic for PHY Lanes 12-15
4	PCIE_NBCFG_REGF -- NBMISCIND: 0x27 PCIE_TX_MUX_LEVEL2_2 Set bit[6] to 1	Disables PCIE traffic for PHY Lanes 8-11 since PLL is shared with Lanes 12-15

  - RX Lane Muxing

  There is no receiver data path in DDI mode, so RX lane muxing is not required.

- Step 3: Initialization Sequence for PCIE PHY (only required for serial simulations)

  Execute Step 3 in Case 4 (section *7.2.2.4*), DDI_SL on Lanes 12-15.

- Step 4: Clock Selection for Display FIFO in PCIE

Table 7-83 Clock Selection for Display FIFO in PCIE

Step	Register Settings	Function/Comment
1	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 DISP_LINK_CLK_SEL Set bit[13] to 1	Selects iDISP_LINK_CLK[1] to be the write clock of FIFO 2
2	PCIE_PDNB_CNTL – NBMISCIND: 0x7 DISP_FIFO_RCLK1_SEL Set bits[31:29] to 3'b001	Selects the TXCLK output from PLL C to be the read clock of FIFO 2

- Step 5: Reset Display FIFO in PCIE

Table 7-84 Reset Display FIFO in PCIE

Step	Register Settings	Function/Comment
1	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 WRESET_FIFO2, RRESET_FIFO2 Set bit[5:4] to 2'b11	Asserts write and read resets for FIFO2
2	Wait for at least 100ns	
3	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 WRESET_FIFO2, RRESET_FIFO2 Set bit[5:4] to 2'b0	De-asserts resets for FIFO2

  - Interrupt Detection Check:
    - Make sure /top/ig/upcie_top/INT0_DET_RX_ASSERTED is 1
    - Make sure /top/ig/upcie_top/INT1_DET_RX_ASSERTED is 1

7.2.2.9 Case 9: DDI_SL on Lanes 4-7 + DDL_SL on Lanes 8-11

If incoherent mode is enabled for DDI_SL on Lanes 4-7, see incoherent mode programming in Case 2 (section 7.2.2.2).

If incoherent mode is enabled for DDI_SL on Lanes 8-11, see incoherent mode programming in Case 3 (section 7.2.2.3).

- To initialize DDI on Lanes 4-7

  See Case 2 (section 7.2.2.2).

- To initialize DDI on Lanes 8-11

  See Case 7 (section 7.2.2.7).

  - Interrupt Detection Check:
    - Make sure /top/ig/upcie_top/INT0_DET_RX_ASSERTED is 1
    - Make sure /top/ig/upcie_top/INT1_DET_RX_ASSERTED is 1

7.2.2.10 Case 10: DDI_SL on Lanes 4-7 + DDL_SL on Lanes 12-15

If incoherent mode is enabled for DDI_SL on Lanes 4-7, see incoherent mode programming in Case 2 (section 7.2.2.2).

Incoherent mode for DDL_SL on Lanes 12-15 is not supported in this case.

- To initialize DDI on Lanes 4-7

  See Case 2 (section 7.2.2.2).

- To initialize DDI on Lanes 12-15

  See Case 8 (section 7.2.2.8).

7.2.2.11 Case 11: DDI_DL on Lanes 8-15

If incoherent mode is enabled, refer to incoherent mode programming in Case 3 (section 7.2.2.3), DDI_SL on Lanes 8-11.

- Step 1: Clock Muxing Control
  - TXCLK

Table 7-85 TXCLK

Step	Register Settings	Function/Comment
1	PCIE_PDNB_CNTL -- NBMISCIND: 0x7 IO_TXCLK_C_SEL Set bits[25:24] to 2'b10	Selects PLL C to be the source of B_PTX_DATA_CLK for Lanes 8-15
2	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 Reg_Turn_Off_Both_PLLs Set bit[1] to 0	Allows PLL C to be powered down using register control

  - RXCLK

  There is no receiver data path in DDI mode, so RXCLK is not required.

- Step 2: Lane Muxing Control
  - TX Lane Muxing

Table 7-86 TX Lane Muxing

Step	Register Settings	Function/Comment
1	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 PCIE_DISP_FIFO_CfgDualLink Set bit[23] to 1	Enables dual link mode
2	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 PCIE_DDI_MUX_LEVEL0_1 Set bit[16] to 0	Selects channel 1 data from display

3	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 PCIE_DDI_MUX_LEVEL0_2 Set bit[17] to 0	Selects channel 2 data from display
4	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 PCIE_DDI_MUX_LEVEL1_2 Set bit[20] to 1	Passes channel 1 data to Lanes 8-11
5	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 PCIE_DDI_MUX_LEVEL1_3 Set bit[21] to 0	Passes channel 2 data to Lanes 12-15
6	PCIE_NBCFG_REGF -- NBMISCIND: 0x27 PCIE_TX_MUX_LEVEL2_2 Set bit[6] to 1	Selects DDI traffic for PHY Lanes 8-11
7	PCIE_NBCFG_REGF -- NBMISCIND: 0x27 PCIE_TX_MUX_LEVEL2_3 Set bit[7] to 1	Selects DDI traffic for PHY Lanes 12-15

- RX Lane Muxing

  There is no receiver data path in DDI mode, so RX lane muxing is not required.

- Step 3: Initialization Sequence for PCIE PHY (only required for serial simulations)

  Execute Step 3 in Case 3 (section *7.2.2.3*), DDL_SL on Lanes 8-11.

- Step 4: Clock Selection for Display FIFO in PCIE

Table 7-87 Clock Selection for Display FIFO in PCIE

Step	Register Settings	Function/Comment
1	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 DISP_LINK_CLK_SEL Set bit[13:12] to 2'b11	Selects iDISP_LINK_CLK[0] and iDISP_LINK_CLK[1] to be the write clock of FIFO 1 and FIFO 2 respectively
2	PCIE_PDNB_CNTL – NBMISCIND: 0x7 DISP_FIFO_RCLK0_SEL Set bits[28:26] to 3'b001	Selects the TXCLK output from PLL C to be the read clock of FIFO 1
3	PCIE_PDNB_CNTL – NBMISCIND: 0x7 DISP_FIFO_RCLK1_SEL Set bits[31:29] to 3'b001	Selects the TXCLK output from PLL C to be the read clock of FIFO 2

- Step 5: Reset Display FIFO in PCIE

Table 7-88 Reset Display FIFO in PCIE

Step	Register Settings	Function/Comment
1	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 WRESET_FIFO1, RRESET_FIFO1 WRESET_FIFO2, RRESET_FIFO2 Set bit[5:2] to 4'b1111	Asserts write and read resets for FIFO1 and FIFO2
2	Wait for at least 100ns	
3	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 WRESET_FIFO1, RRESET_FIFO1 WRESET_FIFO2, RRESET_FIFO2 Set bit[5:2] to 4'b0	De-asserts resets for FIFO1 and FIFO2

- Interrupt Detection Check:
  - Make sure /top/ig/upcie_top/INT0_DET_RX_ASSERTED is 1
  - Make sure /top/ig/upcie_top/INT1_DET_RX_ASSERTED is 1

7.2.2.12 Case 12: DDI_DL on Lanes 8-15 with Mirroring

If incoherent mode is enabled, refer to incoherent mode programming in Case 4 (section *7.2.2.4*), DDI_SL on Lanes 12-15.

- Step 1: Clock Muxing Control
  - TXCLK

Table 7-89 TXCLK

Step	Register Settings	Function/Comment
1	PCIE_PDNB_CNTL -- NBMISCIND: 0x7 IO_TXCLK_C_SEL Set bits[25:24] to 2'b10	Selects PLL C to be the source of B_PTX_DATA_CLK for Lanes 8-15
2	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 Reg_Turn_Off_Both_PLLs Set bit[1] to 0	Allows PLL C to be powered down using register control

  - RXCLK

    There is no receiver data path in DDI mode, so RXCLK is not required.

- Step 2: Lane Muxing Control
  - TX Lane Muxing

Table 7-90 TX Lane Muxing

Step	Register Settings	Function/Comment
1	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 PCIE_DISP_FIFO_CfgDualLink Set bit[23] to 1	Enables dual link mode
2	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 PCIE_DDI_MUX_LEVEL0_1 Set bit[16] to 1	Selects mirrored version of channel 2 data from display
3	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 PCIE_DDI_MUX_LEVEL0_2 Set bit[17] to 1	Selects mirrored version of channel 1 data from display
4	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 PCIE_DDI_MUX_LEVEL1_2 Set bit[20] to 1	Passes mirrored version of channel 2 data to Lanes 8-11
5	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 PCIE_DDI_MUX_LEVEL1_3 Set bit[21] to 0	Passes mirrored version of channel 1 data to Lanes 12-15
6	PCIE_NBCFG_REGF -- NBMISCIND: 0x27 PCIE_TX_MUX_LEVEL2_2 Set bit[6] to 1	Selects DDI traffic for PHY Lanes 8-11
7	PCIE_NBCFG_REGF -- NBMISCIND: 0x27 PCIE_TX_MUX_LEVEL2_3 Set bit[7] to 1	Selects DDI traffic for PHY Lanes 12-15

- RX Lane Muxing

- Step 3: Initialization Sequence for PCIE PHY (only required for serial simulations)

  Execute Step 3 in Case 3 (section *7.2.2.3*), DDL_SL on Lanes 8-11.

- Step 4: Clock Selection for Display FIFO in PCIE

Table 7-91 Clock Selection for Display FIFO in PCIE

Step	Register Settings	Function/Comment
1	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 DISP_LINK_CLK_SEL Set bit[13:12] to 2'b11	Selects iDISP_LINK_CLK[0] and iDISP_LINK_CLK[1] to be the write clock of FIFO 1 and FIFO 2 respectively
2	PCIE_PDNB_CNTL – NBMISCIND: 0x7 DISP_FIFO_RCLK0_SEL Set bits[28:26] to 3'b001	Selects the TXCLK output from PLL C to be the read clock of FIFO 1
3	PCIE_PDNB_CNTL – NBMISCIND: 0x7 DISP_FIFO_RCLK1_SEL Set bits[31:29] to 3'b001	Selects the TXCLK output from PLL C to be the read clock of FIFO 2

- Step 5: Reset Display FIFO in PCIE

Table 7-92 Reset Display FIFO in PCIE

Step	Register Settings	Function/Comment
1	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 WRESET_FIFO1, RRESET_FIFO1 WRESET_FIFO2, RRESET_FIFO2 Set bit[5:2] to 4'b1111	Asserts write and read resets for FIFO1 and FIFO2
2	Wait for at least 100ns	
3	PCIE_NBCFG_REG10 -- NBMISCIND: 0x28 WRESET_FIFO1, RRESET_FIFO1 WRESET_FIFO2, RRESET_FIFO2 Set bit[5:2] to 4'b0	De-asserts resets for FIFO1 and FIFO2

  - Interrupt Detection Check:
    - Make sure /top/ig/upcie_top/INT0_DET_RX_ASSERTED is 1
    - Make sure /top/ig/upcie_top/INT1_DET_RX_ASSERTED is 1

7.2.3 Adjustable PHY Parameters for Better Quality Display

7.2.3.1 TMDS/HDMI

For TMDS/HDMI perform the following:

- Step 1: Set the Transmitter Driving Strength to strongest
- Step 2: Disable De-Emphasis

Table 7-93 Pre-Emphasis Level (dB)

Pre-Emphasis Level (dB)			
Required			Optional
0dB (1x)	3.5dB (1.5x)	6dB (2x)	9.5dB (3x)
Vdiff_pre_pp	Vdiff_pre_pp	Vdiff_pre_pp	Vdiff_pre_pp
0.4 B_PTX_DEEMPH_EN Set to 0 B_P90TX_DEEMPH_STR[7:0] Set to 8'b11000000 B_P90TX_DRV_STR[1:0] Set to 2'b11	0.6 B_PTX_DEEMPH_EN Set to 1 B_P90TX_DEEMPH_STR[7:0] Set to 8'b10000001 B_P90TX_DRV_STR[1:0] Set to 2'b11	0.8 B_PTX_DEEMPH_EN Set to 1 B_P90TX_DEEMPH_STR[7:0] Set to 8'b01000011 B_P90TX_DRV_STR[1:0] Set to 2'b11	1.2 B_PTX_DEEMPH_EN Set to 1 B_P90TX_DEEMPH_STR[7:0] Set to 8'b00000100 B_P90TX_DRV_STR[1:0] Set to 2'b11
0.6 B_PTX_DEEMPH_EN Set to 0 B_P90TX_DEEMPH_STR[7:0] Set to 8'b10000000 B_P90TX_DRV_STR[1:0] Set to 2'b11	0.9 B_PTX_DEEMPH_EN Set to 1 B_P90TX_DEEMPH_STR[7:0] Set to 8'b01000001 B_P90TX_DRV_STR[1:0] Set to 2'b11	1.2 B_PTX_DEEMPH_EN Set to 1 B_P90TX_DEEMPH_STR[7:0] Set to 8'b00000011 B_P90TX_DRV_STR[1:0] Set to 2'b11	Not allowed
0.8 B_PTX_DEEMPH_EN Set to 0 B_P90TX_DEEMPH_STR[7:0] Set to 8'b01000000 B_P90TX_DRV_STR[1:0] Set to 2'b11	1.2 B_PTX_DEEMPH_EN Set to 1 B_P90TX_DEEMPH_STR[7:0] Set to 8'b00000001 B_P90TX_DRV_STR[1:0] Set to 2'b11	Not allowed	Not allowed
1.2 B_PTX_DEEMPH_EN Set to 0 B_P90TX_DEEMPH_STR[7:0] Set to 8'b00000000 B_P90TX_DRV_STR[1:0] Set to 2'b11	Not allowed	Not allowed	Not allowed

7.2.3.2 Transmitter Driving Strength

Table 7-94 Transmitter Driving Strength

Register Settings	Function/Comment
PCIE_NBCFG_REGF -- NBMISCIND: 0x27 REG_B_P90TX_DRV_STR_A Bits[17:16]	Sets driving strength for Lane 0 2'b11 = strongest driving strength
DCIO_UNIPHY: UNIPHY_MACRO_CONTORL1 - GpuF0MMReg: 0x7EC0 Bits[15:4]	Sets driving strength for Lane 1
DCIO_UNIPHY: UNIPHY_MACRO_CONTORL2 - GpuF0MMReg: 0x7EC4 Bits[17:16]	Sets driving strength for Lane 2
DCIO_UNIPHY: UNIPHY_MACRO_CONTORL3 - GpuF0MMReg: 0x7EC8 Bits[17:16]	Sets driving strength for Lane 3
PCIE_NBCFG_REGF -- NBMISCIND: 0x27 REG_B_P90TX_DRV_STR_B Bits[19:18]	Sets driving strength for Lanes 4-7 2'b11 = strongest driving strength
PCIE_NBCFG_REGF -- NBMISCIND: 0x27 REG_B_P90TX_DRV_STR_C Bits[21:20]	Sets driving strength for Lanes 8-15 2'b11 = strongest driving strength

7.2.3.3 De-Emphasis Enable

Table 7-95 De-Emphasis Enable

Register Settings	Function/Comment
PCIE_NBCFG_REG14 -- NBMISCIND: 0x2C REG_B_PTX_DEEMPH_EN_A Bit[24]	Enables de-emphasis for Lane 0
DCIO_UNIPHY: UNIPHY_MACRO_CONTORL1 - GpuF0MMReg: 0x7EC0 Bit[20]	Enables de-emphasis for Lane 1
DCIO_UNIPHY: UNIPHY_MACRO_CONTORL2 - GpuF0MMReg: 0x7EC8 Bit[28]	Enables de-emphasis for Lane 2
DCIO_UNIPHY: UNIPHY_MACRO_CONTORL3 - GpuF0MMReg: 0x7EC0 Bit[20]	Enables de-emphasis for Lane 3
PCIE_NBCFG_REG14 -- NBMISCIND: 0x2C REG_B_PTX_DEEMPH_EN_B Bit[25]	Enables de-emphasis for Lanes 4-7
PCIE_NBCFG_REG14 -- NBMISCIND: 0x2C REG_B_PTX_DEEMPH_EN_C Bit[26]	Enables de-emphasis for Lanes 8-15

7.2.3.4 De-Emphasis Driving Strength

Table 7-96 De-Emphasis Driving Strength

Register Settings	Function/Comment
PCIE_NBCFG_REG14 -- NBMISCIND: 0x2C REG_B_P90TX_DEEMPH_STR_A Bits[7:0]	Sets de-emphasis strength for Lane 0
DCIO_UNIPHY: UNIPHY_MACRO_CONTORL1 - GpuF0MMReg: 0x7EC0 Bits[13:12] Bits[9:8] Bits[3:0]	Sets de-emphasis strength for Lane 1. The 8-bit field is split into two 2-bit fields and one 4-bit field.
DCIO_UNIPHY: UNIPHY_MACRO_CONTORL2 - GpuF0MMReg: 0x7EC4 Bits[15:12] Bits[7:4]	Sets de-emphasis strength for Lane 2. The 8-bit field is split into two 4-bit fields.
DCIO_UNIPHY: UNIPHY_MACRO_CONTROR3 - GpuF0MMReg: 0x7EC8 Bits[14:8] Bit[0]	Sets de-emphasis strength for Lane 3. The 8-bit field is split into 7-bit and 1-bit fields.
PCIE_NBCFG_REG14 -- NBMISCIND: 0x2C REG_B_PTX_DEEMPH_STR_B Bits[15:8]	Sets de-emphasis strength for Lanes 4-7
PCIE_NBCFG_REG14 -- NBMISCIND: 0x2C REG_B_PTX_DEEMPH_STR_C Bits[23:16]	Sets de-emphasis strength for Lanes 8-15

7.3 PCIE + DDI Modes

Table 7-97 shows the PCIE and DDI combined modes that need to be tested:

Table 7-97 PCIE + DDI Modes

Lanes	0 to 3	4 to 7	8 to 11	12 to 15
1	GFX x8 A		DDI_SL	
2	GFX x8 A			DDI_SL
3	GFX x8 A		DDI_DL	
4	DDI_SL		GFX x8 A	
5		DDI_SL	GFX x8 A	
6	DDI_SL	DDI_SL	GFX x8 A	
7	DDI_DL		GFX x8 A	
8	GPP x4 A	GPP x4 B	DDI_SL	
9	GPP x4 A	GPP x4 B		DDI_SL
10	GPP x4 A	GPP x4 B	DDI_DL	
11	DDI_SL		GPP x4 A	GPP x4 B
12		DDI_SL	GPP x4 A	GPP x4 B
13	DDI_SL	DDI_SL	GPP x4 A	GPP x4 B
14	DDI_DL		GPP x4 A	GPP x4 B

The programming in section *7.1* (PCIE Modes) and section *7.2* (DDI Modes) can be combined accordingly.

This page intentionally left blank.

Chapter 8
HTIU Settings

8.1 HT Link Initialization

The RS780 northbridge is expected to be the only HT device in the system other than the CPU. Moving the chipset off of unit ID 0 to look for additional devices, as described in the AMD BKDG and HyperTransport specification, is neither necessary, nor is it supported. All devices on PCI bus 0 are expected to be located within the northbridge and the southbridge. The use of a HyperTransport tunnel device between the CPU and the RS780 is not supported.

Note: If multiple northbridges are used, then the ones that are connected to the southbride must be placed on separate logical PCI buses.

8.2 HTIU Indirect Register Space

Many non-standard HTIU related registers are located in an indirect register space HTIUNBIND. This is accessed through HTIU_NB_INDEX (with write-enable in bit [8], and index in bits [6:0]), and HTIU_NB_DATA in the NBCFG space. Note that the location of the index/data pair is different than in previous chipsets (such as the RS690). They are now located in bus 0 device 0 function 0 registers 0x94 and 0x98.

8.3 CPU Register Access

8.3.1 Normal Registers

For a single-socket platform, the CPU will be accessible through bus 0x0 device 0x18 with various function numbers. Note: In this document, these registers will be referenced as CPU_F(function number) x offset.

8.3.2 PHY Dataport Register Access

Refer to the section entitled “Link Phy Offset Registers (Function 4 registers 0x180, 188, 190, 198)” in the *BIOS and Kernel Developers’ Guide (BKDG) for AMD Family 10h Processors* to see how to program the CPU HyperTransport PHY registers. Note: In the HTIU section of this document, these registers will be referenced as CPU_PHY 0xYY, where YY is the register offset. Also refer to “Link Phy Offset Registers (Function 4 registers 0x180, 188, 190, 198)” in the *BIOS and Kernel Developers’ Guide (BKDG) for AMD Family 11h Processors*.

The current assumption in this document is that the northbridge will always be connected to Link 0 so that CPU function 4 registers 0x180 and 0x184 will be used to control the PHY. For multi-northbridge configurations this may not be true.

8.4 Changing to High-Speed Mode

The HT link starts at 200MHz/8-bit mode on initial power-up. Software is responsible for reprogramming both the CPU(s) and northbridge(s) for higher speed operation. Generating a warm reset will cause the system to restart using the new higher speeds.

One can also change the link speed using LDTSTOP. However, this operation is only valid for HT3 frequencies and can not be use to change between HT1 and HT3 mode. The sequence to trigger an LDTSTOP event is as follows:

- Step 1: Save PMREG 0x8A
- Step 2: Program PMREG 0x8A to 0xF0 (set maximum LDTSTOP time)
- Step 3: Write 1 to PMREG 0x87 (toggle LDTSTOP)
- Step 4: Restore PMREG 0x8A value

8.4.1 Identifying Supported HT Frequencies

The HT frequency capability register located in NBCFG x d0[31:16] determines which HT frequencies are supported by the northbridge. The bits in this 16-bit register denote which frequency codes are valid. For example, a value of 0x25 indicates that frequency codes 0x0, 0x2, and 0x5 are valid indicating 200Mhz, 400Mhz, and 800Mhz support. Similarly, the HT frequency capability register in the CPU is located in CPU_F0 x88 bits [31:16] and are encoded in the same manner. Both the northbridge and the CPU must be set to the same frequency. Note that the lower frequencies require

HyperTransport 1 mode to operate while higher frequencies require HyperTransport 3 mode to operate. Switching between HT1 and HT3 mode requires a warm reset, while changing between HT3 frequencies can be done using both warm reset and LDTSTOP.

Table 8-1 Identifying Supported HT Frequencies

HT Frequency	Frequency Code	IBIAS	Function/Comments
200Mhz	0x0	0x4C	HyperTransport 1 only
400Mhz	0x2	0x4C	HyperTransport 1 only
600Mhz	0x4	0xB6	HyperTransport 1 only
800Mhz	0x5	0x4C	HyperTransport 1 only
1Ghz	0x6	0x9D	HyperTransport 1 only
1.2Ghz	0x7	0xB6	HyperTransport 3 only
1.4Ghz	0x8	0x2B	HyperTransport 3 only
1.6Ghz	0x9	0x4C	HyperTransport 3 only
1.8Ghz	0xa	0x6C	HyperTransport 3 only
2.0Ghz	0xb	0x9D	HyperTransport 3 only
2.2Ghz	0xc	0xAD	HyperTransport 3 only
2.4Ghz	0xd	0xB6	HyperTransport 3 only
2.6Ghz	0xe	0xC6	HyperTransport 3 only

8.4.2 Changing to High-Speed HyperTransport 1 Mode

The following registers in *Table 8-2* must be programmed to enable one of the high-speed HT1 modes.

Table 8-2 Enabling High-Speed HT1 Modes

ASIC Rev	Register	Setting	Function/Comment
RS780 All Revs	NBCFG x d0 [11:8]	HT1 Frequency Code from *Table 8-1*	Sets HT Frequency in the chipset
	NBCFG x c8 [26:24]	0x1	Sets input HT link width on the chipset side to be 16-bits. Only 8-bit and 16-bit widths are supported
	NBCFG x c8 [30:28]	0x1	Sets output HT link width on the chipset side to be 16-bits. Only 8-bit and 16-bit widths are supported
	CLKCFG x d8 [9:0]	IBIAS Code from *Table 8-1*	Sets the chipset HT PLL ibias setting
	HTIUNBIND x 2A [1:0]	0x1	Optimize chipset HT transmitter drive strength
	CPU_F0 R88[11:8]	HT1 Frequency Code from *Table 8-1*	Sets HT Frequency in the CPU
	CPU_F0 R84[26:24]	0x1	Sets input HT link width on the CPU side to be 16-bits
	CPU_F0 R84[30:28]	0x1	Sets output HT link width on the CPU side to be 16-bits
Note: Once these registers are programmed, a warm-reset must be performed to switch to high-speed mode.			

8.4.3 Changing to HyperTransport 3 Mode

HyperTransport 3 capability in the northbridge can be detected by searching the PCI capability linked list for a header with Capability_ID 0x8 and Capability_Type 0xD0. Assuming the CPU is HyperTransport 3 capable, HyperTransport 3 modes can be enabled by programming the following registers in *Table 8-3*. This can be done directly from the 200Mhz/8-bit bootup mode. Please note that to program HT3 capability registers properly, NBCFG x 9c[17:16] must be set to 0x0.

Note: Once the registers are programmed, a warm reset is needed if the system is switching from HT1 to HT3 mode.

Table 8-3 RS780 Register Settings for HyperTransport 3 Mode

ASIC Rev	Register	Setting	Function/Comment
RS780 All Revs	NBCFG x d0 [11:8]	HT3 Frequency Code from *Table 8-1*	Sets HT Frequency in the chipset
	NBCFG x c8 [26:24]	0x1	Sets input HT link width on the chipset side to be 16-bits. Only 8-bit and 16-bit widths are supported
	NBCFG x c8 [30:28]	0x1	Sets output HT link width on the chipset side to be 16-bits. Only 8-bit and 16-bit widths are supported
	CLKCFG x d8 [9:0]	IBIAS Code from *Table 8-1*	Sets the chipset HT PLL ibias setting
	NBCFG x 44 [0]	0x1	Enables error-retry mode
	NBCFG x ac [3]	0x1	Enables scrambling
	NBCFG x a4 [31]	0x1	Enables transmitter de-emphasis
	NBCFG x a4 [26:24]	See notes.	Enables transmitter de-emphasis level This depends on the PCB design and the trace length between NB and CPU Use 0x4 for bringup. See the Transmitter Deemphasis section for programming guidelines.
	NBCFG x a0 [5:0]	LS0 = 1us LS1 = 2us LS2 = CPU_PHY x [530A,520A] [Ls2ExitTime] + 2us	Sets Training 0 time. These settings are chosen to match the processor requirements. See T0Time table for encodings
	NBCFG x ac [14]	0x1	Disables command throttling
	HTIUNBIND x 15 [22]	0x1	Enables strict TM4 detection
	HTIUNBIND x 2A [1:0]	0x1	Optimizes chipset HT transmitter drive strength
	HTIUNBIND x 016[10]	0x1	Enables proper DLL reset sequence

Table 8-4 HyperTransport 3 Processor Register Settings

ASIC Rev	Register	Setting	Function/Comment
RS780 All Revs	CPU_F0 x88[11:8]	HT3 Frequency Code from *Table 8-1*	Sets HT Frequency in the CPU
	CPU_F0 x84[26:24]	0x1	Sets input HT link width on the CPU side to be 16-bits
	CPU_F0 x84[30:28]	0x1	Sets output HT link width on the CPU side to be 16-bits
	CPU_F0 x130[0]	0x1	Enables error-retry mode
	CPU_F0 x170[3]	0x1	Enables scrambling
	CPU_PHY x C5	See notes	Enables transmitter de-emphasis This depends on the PCB design and the trace length between NB and CPU For bringup, use the -3dB settings described in the appropriate processor BKDG (for family 10h and 11h, this is located under “Link PHY De-emphasis Value Registers”).
	CPU_PHY x D5	See notes	Enables transmitter de-emphasis This depends on the PCB design and the trace length between NB and CPU For bringup, use the -3dB settings described in the appropriate processor BKDG (for family 10h and 11h this is located under “Link PHY De-emphasis Value Registers”).
	CPU_F0 x168[10]	0x1	Disables command throttling
	CPU_F0 x16C[5:0]	LS0 = 1us LS1 = 2us LS2 = CPU_PHY x [530A,520A] [Ls2ExitTime] + 2us	Sets Training 0 Time. See T0Time table for encodings
	CPU_F0 x16C[5:0]	LS0 = 1us LS1 = 2us LS2 = CPU_PHY x [530A,520A] [Ls2ExitTime] + 2us	Sets Training 0 Time. See T0Time table for encodings

Table 8-5 T0Time Settings From the HyperTransport 3 Specification

T0Tme [3:0]	T0Time [5:4]			
	0x0	0x1	0x2	0x3
0x 0	0.0us	0.0 us	0 us	0.0us
0x 1	0.1us	0.5 us	2 us	20 us
0x 2	0.2 us	1.0 us	4 us	40 us
0x 3	0.3 us	1.5 us	6 us	60 us
0x 4	0.4 us	2.0 us	8 us	80 us
0x 5	0.5 us	2.5 us	10 us	100 us
0x 6	0.6 us	3.0 us	12 us	120 us
0x 7	0.7 us	3.5 us	14 us	140 us
0x 8	0.8 us	4.0 us	16 us	160 us
0x 9	0.9 us	4.5 us	18 us	180 us
0xA	1.0 us	5.0 us	20 us	200 us
0xB	1.1 us	5.5 us	22 us	Reserved
0xC	1.2 us	6.0 us	24 us	Reserved
0xD	1.3 us	6.5 us	26 us	Reserved
0xE	1.4 us	7.0 us	28 us	Reserved
0xF	1.5 us	7.5 us	30 us	Reserved

8.4.4 HT Link Width + LVM Support

In the RS780, 2-bit and 4-bit link width support was added in HT3 mode (not available in HT1 mode). The 4-bit link can be enabled at any time. However for 2-bit mode, Low Voltage support must be enabled as well. Otherwise, hardware will not permit 2-bit width change.

Table 8-6 RS780 Register Settings for HyperTransport 3 Mode

ASIC Rev	Register	Setting	Function/Comment
RS780 All Revs	HTIUNBIND x 46 [16]	0	Disable Low-Voltage Mode

LVM should be disabled by default.

8.5 Workarounds

8.5.1 LPC DMA Deadlock

To avoid a deadlock scenario with LPC DMA transactions, program the following registers in the order they are listed. Also, the LPC memory-mapped registers must not be covered by one of the CPU's MMIO ranges.

Table 8-7 LPC DMA Deadlock

ASIC Rev	Setting	Function/Comment
RS780 All Revs	D24 F0 R68[22:21] = 0x1 for internal graphics D24 F0 R68[22:21] = 0x0 for no internal graphics	If set to 1, this limits the number of outstanding CPU non-posted transactions to 1. This register is located in the CPU and may cause the AMD BIOS checker to issue a warning. This is expected.
	SB PCIEIND x 10 [9] = 0x1	Enables the special NP protocol in the southbridge PCIE.
	GPP PCIEIND x 10 [9] = 0x1	Enables the special NP protocol in the northbridge PCIE.
	HTIUNBIND x 06 [26] = 1	Enables the special NP protocol in HTIU.

8.5.2 CPU Access To UMA Memory Deadlock

The following mechanism applies to all systems which use internal graphics with UMA memory. This works around a deadlock scenario involving CPU access to the UMA frame buffer which is sent to the internal graphics and is reflected back to the CPU. In the CPU, a read response from a reflected DMA read can get stuck behind a large number of CPU writes which in turn may be stuck waiting for the reflected read to finish.

Table 8-8 CPU Access To UMA Memory Deadlock Settings

ASIC Rev	Setting	Function/Comment
RS780 All Revs	The system BIOS must modify the processor MMIO configuration to mark the internal graphics prefetchable and non-prefetchable ranges with the non-posted property. MMIO is controlled through base/limit pairs at d24, f1, and from reg 0x80 to reg 0xBC. The non-posted property is located in bit 7 of each limit register. The internal graphics frame-buffer location is available through B0/D1/F0 reg 0x24. Memory-mapped I/O base/limit pairs are not allowed to overlap, so additional base/limit pairs may need to be used to properly cover all of the memory-mapped space, including the non-posted range for the frame-buffer. As previously stated in the LPC DMA Deadlock scenario, the LPC memory-mapped space should not be covered by an MMIO base/limit pair.	Prevents a deadlock scenario.

8.6 HT Register Settings

8.6.1 HT General Register Settings

Table 8-9 HT General Register Settings

ASIC Rev	Setting	Function/Comment
RS780 All Revs	HTIUNBIND x 1C [17] = 1	Prevents AllowLdtStop from being asserted during HT link recovery
	HTIUNBIND x 06 [0] = 0x0	Enables writes to pass in-progress reads
	HTIUNBIND x 06 [1] = 0x1	Enables streaming of CPU writes
	HTIUNBIND x 06 [9] = 0x1	Enables extended write buffer for CPU writes
	HTIUNBIND x 06 [13] = 0x1	Enables additional response buffers
	HTIUNBIND x 06 [17] = 0x1	Enables special reads to pass writes
	HTIUNBIND x 06 [16:15] = 0x3	Enables decoding of C1e/C3 and FID cycles
	HTIUNBIND x 06 [25] = 0x1	Enables HTIU-display handshake bypass.
	HTIUNBIND x 06 [30] = 0x1	Enables tagging fix
	HTIUNBIND x 07 [0] = 0x1	Enables byte-write optimization for IOC requests
	HTIUNBIND x 07 [1] = 0x0	Disables delaying STPCLK de-assert during FID sequence.
	HTIUNBIND x 07 [2] = 0x0	Disables upstream system-management delay
	HTIUNBIND x 07 [15] = 0x1	Enables drop zero mask request
	HTIUNBIND x C [1:0] = 0x1	Priority for Stage 1a of posted upstream arbitration
	HTIUNBIND x C [3:2] = 0x2	Priority for Stage 1b of posted upstream arbitration
	HTIUNBIND x C [5:4] = 0x0	Set CLMC client to highest priority in posted upstream arbitration
	HTIUNBIND x 23 [28] = 0x1	Enable transmitter reset when exiting disconnect state in LS1 mode
RS780 A12	HTIUNBIND x 2D [4] = 0x1	Enables write combine workaround
	HTIUNBIND x 2D [6] = 0x1	Enables DLL reset on link retrain
	HTIUNBIND x 05 [2] = 0x1	Enables weight round robin between read/write
	HTIUNBIND x 1E [31:0] = 0xFFFF_FFFF	Enables all HT protocol checking
RS780 A13	HTIUNBIND x 05 [9] = 0x1	Enables the write ACK FIFO workaround. Program this register **before** any GFX traffic (prior to VBIOS)
	HTIUNBIND x 05 [10] = 0x1	Enables the Special Target Done workaround
	HTIUNBIND x 17 [30] = 0x1	Enable GSM DMA detection in C3 only

8.6.2 UnitID Clumping

The RS780 supports UnitID clumping to increase the number of outstanding requests supported by a single device. Clumping is only supported with family 10h and 11h processors. It may be enabled for PCI-Express GFX links in certain configurations. Clumping may be enabled when using only the lower number bridge within each PCI-Express GFX core. For example, when using bridge 2, and not bridge 3 (UnitIDs 0x2 and 0x3), to support a single x8 or x16 PCI-Express link, the UnitIDs reserved for both bridges may be clumped together. The clumping register must be enabled the same in both the chipset and processor.

Table 8-10 Register Settings for UnitID Clumping

ASIC Rev	Register	Setting	Comments
RS780 All Revs	NBCFG x 5C [3]	0x1	Enables clumping of UnitIDs 2 and 3.
	CPU_F0 x 110 [3]	0x1	Enables clumping of UnitIDs 2 and 3 in family 10h processors. No processor registers need to be programmed for family 11h processors.
	NBCFG x 5C [16]	0x1	Enables clumping of UnitIDs 0x10 and 0xF (Internal GFX).
	CPU_F0 x 110 [16]	0x1	Enables clumping of UnitIDs 0x10 and 0xF (Internal GFX) in family 10h processors. No processor registers need to be programmed for family 11h processors
	NBCFG x 5C [18]	0x1	Enables clumping of UnitIDs 0x12 and 0x11 (Internal DSP).
	CPU_F0 x 110 [18]	0x1	Enables clumping of UnitIDs 0x12 and 0x11 (Internal DSP) in family 10h processors. No processor registers need to be programmed for family 11h processors
Note: These registers take effect immediately. Clumping registers are cleared on warm-reset and should be programmed before DMA traffic is enabled.			

Note for SBIOS: For RS780 bringup, ensure that UnitID clumping is added as a SBIOS menu option. The options should be as follows:

- UnitID Clumping Disabled (default)
- Clump UnitIDs 0x2 + 0x3 and 0xF + 0x10 and 0x11 + 0x12
- Clump UnitIDs 0x2 + 0x3
- Clump UnitIDs 0xF + 0x10
- Clump UnitIDs 0x11+ 0x12
- Clump UnitIDs 0xF + 0x10 and 0x11 + 0x12

If a particular pair of UnitIDs cannot be clumped because they do not meet the criteria described above, then clumping should not be enabled, regardless of the menu selection.

8.6.3 Isochronous Flow-Control Mode

Although not recommended for integrated graphics performance, the RS780 supports the use of the Isochronous Flow-Control Mode (IFCM) function to provide reduced latency for certain classes of traffic. In this mode, dedicated flow-control buffering must be reserved inside of the processor. IFCM is only supported in AMD Family 10h, AMD Family 11h, and later processors, and is not supported by K8 generation processors. Refer to the register settings in sections *8.6.4* and *8.6.5* to reserve the isochronous flow-control buffers. In addition to this, the isochronous mode must be enabled. IFMC must be enabled to support internal graphics with AMD Family 10h and AMD Family 11h processors.

Table 8-11 Enabling Isochronous Flow-Control Mode Settings

ASIC Rev	Register	Setting	Comments
RS780 All Revs	NBCFG x c8 [12]	0x1	Enables IFCM in the chipset.
	CPU_F0 x 84 [12]	0x1	Enables IFCM in the processor. This setting applies to AMD Family 10h and AMD Family 11h processors.
	CPU F0 x 1D0 [5]	0x1	Enabled Hi-Priority Mode in AMD Family 11h rev B processors. This setting is in addition to enabling IFCM. This setting applies to AMD Family 11h rev B processors.

Note: These registers require a warm reset to take effect.

If isochronous flow-control mode is not enabled, then the legacy display refresh mode should be enabled instead. Display-refresh mode is automatically enabled in K8 generation processors.

Note for SBIOS: Ensure that the display-refresh mode is the default mode for internal graphics.

Table 8-12 Register Settings for Enabling Display-Refresh Mode

ASIC Rev	Register	Setting	Comments
RS780 All Revs	CPU F0 x 68 [24]	0x1	Enables Display Refresh Mode in the processor. This setting applies to family 10h processors as well as family 11h rev A processors. This bit takes effect immediately. Please set it after the warm reset is used to make changes to HyperTransport and processor XBAR/XCS buffer allocations take effect.
	CPU F0 x 1D0 [5]	0x1	Enabled Hi-Priority Mode. This setting applies to family 11h rev B processors. This bit takes effect after warm reset.

8.6.4 AMD Family 10h Processor Buffer Allocation Settings

8.6.4.1 Recommended Buffer Allocation For Internal Graphics (RS780 - All Revs, 10h CPUs)

The recommended settings in this section are tuned to optimize the performances of the RS780. Consult the CPU BKDG for further information. Note that the registers in this section require a warm reset to take effect.

Table 8-13 F0x90 Data Buffer Counter Register - 808502D0

F0x90 Data Buffer Count Register			808502D0
msb	lsb		
31	31	LockBc	1
30	28	Reserved	0
27	25	FreeData	0
24	20	FreeCmd	16
19	18	RspData	2
17	16	NpReqData	2
15	12	ProbeCmd	0
11	8	RspCmd	4
7	5	Preq	12
4	0	NpReqCmd	32

Table 8-14 F0x94 Data Buffer Counter Register - 00000000

F0x94 Data Buffer Count Register			00000000
msb	lsb		
31	29	Reserved	0
28	27	IsocRspData	0
26	25	IsocNpReqData	0
24	22	IsocRspCmd	0
21	19	IsocPReq	0
18	16	IsocNpReqCmd	0
15	8	SeqBusNum	0
7	0	Reserved	0

Table 8-15 F0x68 Link Transaction Control Register

F0x68 Link Transaction Control Register			
msb	lsb		
24		DispRefModeEn	1

Table 8-16 F0x84 Link Control Register

F0x84 Link Control Register			
msb	lsb		
12		IsocEn	0

Table 8-17 F2x90 Link Control Register

F2x90 Link Control Register			
msb	lsb		
10		BurstLength32	0

Table 8-18 F3x6C Data Buffer Count Register

F3x6C Data Buffer Count Register			60018051
msb	lsb		
30	28	IsocrspDbc	6
18	16	UpRspDbc	1
15	15	DatBuf24	1
7	6	DnRspDbc	1
5	4	DnReqDbc	1
2	0	UpReqDbc	1

Table 8-19 F3x70 SRI-To-Xbar Command Buffer Count Register

F3x70 SRI-To-Xbar Command Buffer Count Register			60321151
msb	lsb		
30	28	Isocrspcbc	6
26	24	Isocpreqcbc	0
22	20	Isocreqcbc	3
18	16	Uprspcbc	2
14	12	Dnpreqcbc	1
10	8	Uppreqcbc	1
7	6	Dnrspcbc	1
5	4	Dnreqcbc	1
2	0	Upreqcbc	1

Table 8-20 F3x74 Xbar-To-SRI Command Buffer Count Register

F3x74 Xbar-To-SRI Command Buffer Count Register			00980101
msb	lsb		
31	28	DRReqCBC	0
26	24	Isocpreqcbc	0
23	20	Isocreqcbc	9
19	16	Probecbc	8
14	12	Dnpreqcbc	0
10	8	Uppreqcbc	1
6	4	Dnreqcbc	0
2	0	Upreqcbc	1

Table 8-21 F3x78 MCT-To-Xbar Buffer Count Register

F3x78 MCT-To-Xbar Buffer Count Register			00200C14
msb	lsb		
21	16	RspDBC	32
12	8	ProbeCBC	12
4	0	RspCBC	20

Table 8-22 F3x7C Free List Buffer Count Register

F3x7C Free List Buffer Count Register			L3 Cache Disabled	L3 Cache Enabled
			0007080D	00070811
msb	lsb			
30	28	Xbar2SriFreeListCBInc	0	0
22	20	Sri2XbarFreeRspDBC	0	0
19	16	Sri2XbarFreeXreqDBC	7	7
15	12	Sri2XbarFreeRspCBC	0	0
11	8	Sri2XbarFreeXreqCBC	8	8
4	0	Xbar2SriFreeListCBC	13	17

Table 8-23 F3x140 SRI-To-XCS Token Count Register

F3x140 SRI-To-XCS Token Count Register			00D33656
msb	lsb		
23	20	FreeTok	13
17	16	IsocRspTok	3
15	14	IsocPreqTok	0
13	12	IsocReqTok	3
11	10	DnRspTok	1
9	8	UpRspTok	2
7	6	DnPreqTok	1
5	4	UpPreqTok	1
3	2	DnReqTok	1
1	0	UpReqTok	2

Table 8-24 F3x144 MCT-To-XCS Token Register

F3x144 MCT-To-XCS Token Register			00000036
msb	lsb		
3	0	RspTok	6
7	4	Probe0Tok	3

Table 8-25 F3x148 Link to XCS Token Count Register

F3x148 Link to XCS Token Count Register			8000832A
msb	lsb		
31	30	FreeTok[3:2]	2
15	14	FreeTok[1:0]	2
13	12	IsocRsp0Tok	0
11	10	IsocPreqoTok	0
9	8	IsocReq0Tok	3
7	6	Probe0Tok	0
5	4	Rsp0Tok	2
3	2	PReq0Tok	2
1	0	Req0Tok	2

Table 8-26 F3x158

F3x158			00000000
msb	lsb	LnkToXcsDRToken	
3	0		0

Table 8-27 F3x1A0

F3x1A0			L3 Cash Disabled			L3 Cash Enabled		
			Nunber of CPU Cores			Nunber of CPU Cores		
			2	3	4	2	3	4
msb	lsb		000041C2	000041A2	00004182	00004184	00004144	00004104
14	12	L3ToSriReqCBC	4	4	4	4	4	4
11	9	Reserved	0	0	0	0	0	0
8	4	L3FreeListCBC	28	26	24	24	20	16
3	3	Reserved	0	0	0	0	0	0
2	0	CpuCmdBufCnt	2	2	2	4	4	4

8.6.4.2 Recommended Buffer Allocation For NO Internal Graphics (RS780 - All Revs, 10h CPUs)

The recommended settings in this section are tuned to optimize the performance of the RS780. Consult the CPU BKDG for further information. Note that the registers in this section require a warm reset to take effect.

Table 8-28 F0x90 Data Buffer Counter Register

F0x90 Data Buffer Count Register			808502CF
msb	lsb		
31	31	LockBc	1
30	28	Reserved	0
27	25	FreeData	0
24	20	FreeCmd	16
19	18	RspData	2
17	16	NpReqData	2
15	12	ProbeCmd	0
11	8	RspCmd	4
7	5	Preq	12
4	0	NpReqCmd	30

Table 8-29 F0x94 Data Buffer Counter Register - 00020000

F0x94 Data Buffer Count Register			00020000
msb	lsb		
31	29	Reserved	0
28	27	IsocRspData	0
26	25	IsocNpReqData	0
24	22	IsocRspCmd	0
21	19	IsocPReq	0
18	16	IsocNpReqCmd	2
15	8	SeqBusNum	0
7	0	Reserved	0

Table 8-30 F0x68 Link Transaction Control Register

F0x68 Link Transaction Control Register			
msb	lsb		
24		DispRefModeEn	0

Table 8-31 F0x84 Link Control Register

F0x84 Link Control Register			
msb	lsb		
12		IsocEn	1

Table 8-32 F3x6C Data Buffer Count Register

F3x6C Data Buffer Count Register			60018051
msb	lsb		
30	28	IsocrspDbc	6
18	16	UpRspDbc	1
15	15	DatBuf24	1
7	6	DnRspDbc	1
5	4	DnReqDbc	1
2	0	UpReqDbc	1

Table 8-33 F3x70 SRI-To-Xbar Command Buffer Count Register

F3x70 SRI-To-Xbar Command Buffer Count Register			60321151
msb	lsb		
30	28	Isocrspcbc	6
26	24	Isocpreqcbc	0
22	20	Isocreqcbc	3
18	16	Uprspcbc	2
14	12	Dnpreqcbc	1
10	8	Uppreqcbc	1
7	6	Dnrspcbc	1
5	4	Dnreqcbc	1
2	0	Upreqcbc	1

Table 8-34 F3x74 Xbar-To-SRI Command Buffer Count Register

F3x74 Xbar-To-SRI Command Buffer Count Register			00980101
msb	lsb		
31	28	DRReqCBC	0
26	24	Isocpreqcbc	0
23	20	Isocreqcbc	9
19	16	Probecbc	8
14	12	Dnpreqcbc	0
10	8	Uppreqcbc	1
6	4	Dnreqcbc	0
2	0	Upreqcbc	1

Table 8-35 F3x78 MCT-To-Xbar Buffer Count Register

F3x78 MCT-To-Xbar Buffer Count Register			00200C14
msb	lsb		
21	16	RspDBC	32
12	8	ProbeCBC	12
4	0	RspCBC	20

Table 8-36 F3x7C Free List Buffer Count Register

F3x7C Free List Buffer Count Register			L3 Cache Disabled	L3 Cache Enabled
			0007080D	00070811
msb	lsb			
30	28	Xbar2SriFreeListCBInc	0	0
22	20	Sri2XbarFreeRspDBC	0	0
19	16	Sri2XbarFreeXreqDBC	7	7
15	12	Sri2XbarFreeRspCBC	0	0
11	8	Sri2XbarFreeXreqCBC	8	8
4	0	Xbar2SriFreeListCBC	13	17

Table 8-37 F3x140 SRI-To-XCS Token Count Register

F3x140 SRI-To-XCS Token Count Register			00D33656
msb	lsb		
23	20	FreeTok	13
17	16	IsocRspTok	3
15	14	IsocPreqTok	0
13	12	IsocReqTok	3
11	10	DnRspTok	1
9	8	UpRspTok	2
7	6	DnPreqTok	1
5	4	UpPreqTok	1
3	2	DnReqTok	1
1	0	UpReqTok	2

Table 8-38 F3x144 MCT-To-XCS Token Register

F3x144 MCT-To-XCS Token Register			00000036
msb	lsb		
3	0	RspTok	6
7	4	Probe0Tok	3

Table 8-39 F3x148 Link-To-XCS Token Count Register

F3x148 Link-To-XCS Token Count REgister			8000832A
msb	lsb		
31	30	FreeTok[3:2]	2
15	14	FreeTok[1:0]	2
13	12	IsocRsp0Tok	0
11	10	IsocPreqoTok	0
9	8	IsocReq0Tok	3
7	6	Probe0Tok	0
5	4	Rsp0Tok	2
3	2	PReq0Tok	2
1	0	Req0Tok	2

Table 8-40 F3x158

F3x158			00000000
msb	lsb	LnkToXcsDRToken	
3	0		0

Table 8-41 F3x1A0

F3x1A0			Disabled			Enabled		
			Nunber of CPU Cores			Nunber of CPU Cores		
			2	3	4	2	3	4
msb	lsb		000041C2	000041A2	00004182	00004184	00004144	00004104
14	12	L3ToSriReqCBC	4	4	4	4	4	4
11	9	Reserved	0	0	0	0	0	0
8	4	L3FreeListCBC	28	26	24	24	20	16
3	3	Reserved	0	0	0	0	0	0
2	0	CpuCmdBufCnt	2	2	2	4	4	4

8.6.4.3 Buffer Allocation For Isochronous Flow Control Mode (Not Recommended For Optimal Performance)

The following tables in the section are provided for completeness. For optimal performance the settings in the previous sections should be used.

Table 8-42 HT Buffer Allocation For Non-coherent Links And Isochronous Flow-Control Mode And No Internal Graphics (Not Recommended)

ASIC Rev	Register	Setting	Comments
RS780 All Revs	CPU_F0 x 90/B0/D0 [27:25]	0x1	2 Free Data
	CPU_F0 x 90/B0/D0 [24:20]	0x8	16 Free Command
	CPU_F0 x 90/B0/D0 [19:18]	0x1	2 Response Data
	CPU_F0 x 90/B0/D0 [17:16]	0x0	0 Non-posted Data
	CPU_F0 x 90/B0/D0 [15:12]	0x0	0 Probe
	CPU_F0 x 90/B0/D0 [11:8]	0x1	2 Response Command
	CPU_F0 x 90/B0/D0 [7:5]	0x6	12 Posted Command and 12 Posted Data
	CPU_F0 x 90/B0/D0 [4:0]	0x10	32 Non-posted Command
	CPU_F0 x 94/B4/D4 [18:16]	0x1	2 Isochronous Non-posted Command
Note: These registers require a warm reset to take effect.			

Table 8-43 HT Buffer Allocation For Non-coherent Links And Isochronous Flow-Control Mode And Internal Graphics (Not Recommended)

ASIC Rev	Register	Setting	Comments
RS780 All Revs	CPU_F0 x 90/B0/D0 [27:25]	0x1	2 Free Data
	CPU_F0 x 90/B0/D0 [24:20]	0x8	16 Free Command
	CPU_F0 x 90/B0/D0 [19:18]	0x1	2 Response Data
	CPU_F0 x 90/B0/D0 [17:16]	0x0	0 Non-posted Data
	CPU_F0 x 90/B0/D0 [15:12]	0x0	0 Probe
	CPU_F0 x 90/B0/D0 [11:8]	0x1	2 Response Command
	CPU_F0 x 90/B0/D0 [7:5]	0x6	12 Posted Command and 12 Posted Data
	CPU_F0 x 90/B0/D0 [4:0]	0xA	20 Non-posted Command
	CPU_F0 x 94/B4/D4 [18:16]	0x7	14 Isochronous Non-posted Command
Note: These registers require a warm reset to take effect.			

Table 8-44 AMD Family 10h Processor XBAR/XCS Buffer Allocation For Isochronous Flow Control Mode and Internal Graphics

ASIC Rev	Register	Setting	Comments
RS780 All Revs	CPU_F3 x 70 [30:28]	0x6	IsocRspCBC
	CPU_F3 x 70 [26:24]	0x0	IsocPreqCBC
	CPU_F3 x 70 [22:20]	0x3	IsocReqCBC
	CPU_F3 x 70 [18:16]	0x2	UpRspCBC
	CPU_F3 x 70 [14:12]	0x1	DnPreqCBC
	CPU_F3 x 70 [10:8]	0x1	UpPreqCBC
	CPU_F3 x 70 [7:6]	0x1	DnRspCBC
	CPU_F3 x 70 [5:4]	0x1	DnReqCBC
	CPU_F3 x 70 [2:0]	0x1	UpReqCBC
	CPU_F3 x 74 [31:28]	0xC	DRReqCBC
	CPU_F3 x 74 [26:24]	0x0	IsocPReq
	CPU_F3 x 74 [23:20]	0x1	IsocReqCBC
	CPU_F3 x 74 [19:16]	0x8	ProbeCBC
	CPU_F3 x 74 [14:12]	0x0	DnPreqCBC
	CPU_F3 x 74 [10:8]	0x1	UpPreqCBC
	CPU_F3 x 74 [6:4]	0x0	DnReqCBC
	CPU_F3 x 74 [2:0]	0x1	UpReqCBC
	CPU_F3 x 7C [4:0]	0xC for processors with L3 cache 0x9 for quad-core processors without L3 cache 0xD for dual-core processors without L3 cache 0xF for single-core processors without L3 cache	Xbar2SriFreeListCBC
	CPU_F3 x 7C [22:20]	0x0	Sri2XbarFreeRspDBC
	CPU_F3 x 7C [19:16]	0x7	Sri2XbarFreeXreqDBC
	CPU_F3 x 7C [15:12]	0x0	Sri2XbarFreeRspCBC
	CPU_F3 x 7C [11:8]	0x8	Sri2XbarFreeXreqCBC
	CPU F3 x 6C [30:28]	0x6	IsocRspDBC
	CPU F3 x 6C [18:16]	0x1	UpRspDBC
	CPU F3 x 6C [15]	0x1	DatBuf24
	CPU F3 x 6C [7:6]	0x1	DnRspDBC
	CPU F3 x 6C [5:4]	0x1	DnReqDBC
	CPU F3 x 6C [2:0]	0x1	UpReqDBC
	CPU_F3 x 140 [23:20]	0xC	FreeTok
	CPU_F3 x 140 [17:16]	0x4	IsocRspTok
	CPU_F3 x 140 [15:14]	0x0	IsocPreqTok
	CPU_F3 x 140 [13:12]	0x3	IsocReqTok
	CPU_F3 x 140 [11:10]	0x1	DnRspTok
	CPU_F3 x 140 [9:8]	0x3	UpRspTok
	CPU_F3 x 140 [7:6]	0x1	DnPreqTok
	CPU_F3 x 140 [5:4]	0x1	UpPreqTok
	CPU_F3 x 140 [3:2]	0x1	DnReqTok
	CPU_F3 x 140 [1:0]	0x2	UpReqTok
	CPU_F3 x 144 [7:4]	0x3	ProbeTok
	CPU_F3 x 144 [3:0]	0x6	RspTok
	CPU_F3 x 148 [31:30]	0x2	FreeTok[3:2]
	CPU_F3 x 148 [28]	0x0	IsocRspTok1
	CPU_F3 x 148 [26]	0x0	IsocPreqTok1
	CPU_F3 x 148 [24]	0x0	IsocReqTok1

	CPU_F3 x 148 [23:22]	0x0	ProbeTok1
	CPU_F3 x 148 [21:20]	0x0	RspTok1
	CPU_F3 x 148 [19:18]	0x0	PReqTok1
	CPU_F3 x 148 [17:16]	0x0	ReqTok1
	CPU_F3 x 148 [15:14]	0x1	FreeTok[1:0]
	CPU_F3 x 148 [13:12]	0x0	IsocRspTok0
	CPU_F3 x 148 [11:10]	0x0	IsocPreqTok0
	CPU_F3 x 148 [9:8]	0x1	IsocReqTok0
	CPU_F3 x 148 [7:6]	0x0	ProbeTok0
	CPU_F3 x 148 [5:4]	0x2	RspTok0
	CPU_F3 x 148 [3:2]	0x2	PReqTok0
	CPU_F3 x 148 [1:0]	0x2	ReqTok0
	CPU_F3 x 158 [3:0]	0x3	LnkToXcsDRToken
	CPU_F3 x 14C	0x0	Link1 to XCS Token Count Register
	CPU_F3 x 150	0x0	Link2 to XCS Token Count Register
	CPU_F3 x 154	0x0	Link3 to XCS Token Count Register
	CPU F2 x 118[13:12]	0x3	MctPriIsoc
	CPU F2 x 118[31:28]	0x1	MctVarPriCntLmt
Note: These registers require a warm reset to take effect.			

Note: The RS780 does not issue upstream isoc writes so IsocPreq related fields can be set to 0x0. For further information please refer to the AMD Family 10h Processor BKDG.

8.6.5 AMD Family 11h Buffer Allocation Settings

8.6.5.1 Recommend Buffer Allocation (RS780 - All Revs, 11h CPUs, Rev B+ Only)

The recommended settings in this section are tuned to optimizing the performance of the RS780. Consult the CPU BKDG for further information. Note that the registers in this section require a warm reset to take effect.

Table 8-45 F0x90 Data Buffer Count Register

F0x90 Data Buffer Count Register			UMA	Non-UMA
			001005B6	001005B5
msb	lsb			
31	24	Reserved	0	0
23	20	RspData	1	1
19	19	Reserved	0	0
18	15	NPReqData	0	0
14	10	RspCmd	1	1
9	5	Preq	13	13
4	0	NpReqCmd	22	21

Table 8-46 F0x94 Data Buffer Count Register

F0x94 Data Buffer Count Register			UMA	Non-UMA
			0000000	00000001
msb	lsb			
31	24	Reserved	0	0
23	20	IsocRspData	0	0
19	19	Reserved	0	0
18	15	IsocNpReqData	0	0
14	10	IsocRspCmd	0	0
9	9	Reserved	0	0
8	5	IsocPreq	0	0
4	0	IsocNpReqCmd	0	1

Table 8-47 F0x68 Link Transaction Control Register

F0x68 Link Transaction Control Register			UMA	Non-UMA
msb	lsb			
24	24	DispRefModeEn	1	0

Table 8-48 F0x84 Link Control Register

F0x84 Link Control Register			UMA	Non-UMA
msb	lsb			
12	12	IsocEn	0	1

Table 8-49 F0x1D0 Extended LInk Buffer Count Register

F0x1D0 Extended Link Buffer Count Register			UMA	Non-UMA
msb	lsb			
5	5	HiPriModeEn	0	1

Table 8-50 F0x1A4 Downstream ONION Buffer Count Register

F0x1A4 Downstream ONION Buffer Count Register			UMA	Non-UMA
			00002821	04212824
msb	lsb			
31	31	Reserved	0	0
30	26	DnHiRespBC	0	1
25	21	DnHiNpReqBC	0	1
20	16	DnHiPReqBC	0	1
15	15	Reserved	0	0
14	10	DnLoRespBC	10	10
9	5	DnLoNpReqBC	1	1
4	0	DnLoPReqBC	1	4

Table 8-51 F0x1D4 Downstream ONION Buffer Count Register 2

F0x1D4 Downstream ONION Buffer Count Register 2			UMA	Non-UMA
			000000D0	00000010
msb	lsb			
31	10	Reserved	0	0
9	5	DnDRRespBC	6	0
4	0	DnPoolBC	16	16

Table 8-52 F2x90 Link Control Register

F2x90 Link Control Register			
msb	lsb		
10	10	BurstLength32	0

Table 8-53 F3x6C Upstream ONION Buffer Count Register

F3x6C Upstream ONION Buffer Count Register			UMA	Non-UMA
			08000808	08110815
msb	lsb			
31	28	Reserved	0	0
27	24	UpHiRespDBC	8	8
23	20	UpHiNpreqDBC	0	1
19	16	UpHiPreqDBC	0	1
15	12	Reserved	0	0
11	8	UpLoRespDBC	8	8
7	4	UpLoNpreqDBC	0	1
3	0	UpLoPreqDBC	8	5

Table 8-54 F3x7C In-Flight Queue Buffer Allocation Register

F3x7C In-Flight Queue Buffer Allocation Register			UMA	Non-UMA
			85001101	B0111101
msb	lsb			
31	28	FreePoolBC	8	11
27	24	DispRefrBC	5	0
23	20	HiPriNPBC	0	1
19	16	HiPriPBC	0	1
15	12	LoPriNPBC	1	1
11	8	LoPriPBC	1	1
7	4	Reserved	0	0
3	0	Cpu0BC	1	1

Table 8-55 NB_HTIUNBCFG:NB_HT_ARB_II

NB_HTIUNBCFG:NB_HT_ARB_II			00001000
msb	lsb		
24	24		1

Table 8-56 MSR C001_001F

MSR C001_001F			UMA	Non-UMA
msb	lsb			
58	58	EnConvertToNonSync	1	1

Table 8-57 F3x74 Upstream ONION Command Buffer Count Register

F3x74 Upstream ONION Command Buffer Count Register			UMA HT, & MC SLOW (EMP | DDR400 | HT<=1400)	UMA HT & MC Fast ~(EMP | DDR400 | HT<=1400)	Non-UMA
			40000866	30000876	02110695
msb	lsb				
31	28	UpDRReqCBC	4	3	0
27	24	UpHiRespCBC	0	0	2
23	20	UpHiNpreqCBC	0	0	1
19	16	UpHiPreqCBC	0	0	1
15	12	Reserved	0	0	0
11	8	UpLoRespCBC	8	8	6
7	4	UpLoNpreqCBC	6	7	9
3	0	UpLoPreqCBC	6	6	5

8.6.5.2 Buffer Allocation For Other Modes And Rev A CPUs (Legacy Modes, Non-optimal For Performance)

The following tables in this section are provided for completeness. For optimal performance the settings in the previous sections should be used.

Refer to the Bios Kernel Developer's Guide (BKDG) for AMD Family 11h Processors for specific details on how to program the processors buffer allocations settings. In particular, refer to the section entitled "Display Refresh and IFCM".

- For internal graphics and AMD Family 11h revision A processors set F0x68[DispRefModeEn]=0 and F0x84[IsocEn]=1 but use the "Lo, Hi(DR)" table columns to program register fields in F0x1A4, F0x1D4, F3x6C, F3x74 and F3x7C.
- For internal graphics and AMD Family 11h revision B processors set F0x68[DispRefModeEn]=1,F0x84[IsocEn]=1, F0x1D0[HiPriModeEn]=1 and MSRC001_001F[EnConvertToNonIsoc]=1 but use the "Lo, DR" table columns to program register fields in F0x1A4, F0x1D4, F3x6C, F3x74 and F3x7C.
- For non-internal graphics and AMD Family 11h revision A processors set F0x68[DispRefModeEn]=0 and F0x84[IsocEn]=1 and use the programming tables listed below for register fields in F0x1A4, F0x1D4, F3x6C, F3x74 and F3x7C instead of the BKDG values.
- For non=internal graphics and AMD Family 11h revision B processors set F0x68[DispRefModeEn]=1,F0x84[IsocEn]=1, F0x1D0[HiPriModeEn]=1 and MSRC001_001F[EnConvertToNonIsoc]=1 and use the programming tables listed below for register fields in F0x1A4, F0x1D4, F3x6C, F3x74 and F3x7C instead of the BKDG values.
- For F0x94 and F0x90, please use the programming information found in *Table 8-58* below, rather than the one listed in the BKDG.

Table 8-58 HT Buffer Allocation for Isochronous Flow-Control Mode

Processor	Register	Setting
Revision A	F0x90[RspData]	0x1
	F0x90[NpReqData]	0x0
	F0x90[RspCmd]	0x1
	F0x90[PReq]	0x6 for UMA mode 0x7 for non-UMA mode
	F0x90[NpReqCmd]	0x13 for UMA mode 0x17 for non-UMA mode
	F0x94[IsocRspData]	0x0
	F0x94[IsocNpReqData]	0x0
	F0x94[IsocRspCmd]	0x0
	F0x94[IsocPReq]	0x0
	F0x94[IsocNpReqCmd]	0x6 for UMA mode 0x1 for non-UMA mode
Revision B	F0x90[RspData]	0x1
	F0x90[NpReqData]	0x0
	F0x90[RspCmd]	0x1
	F0x90[PReq]	0xF
	F0x90[NpReqCmd]	0xD for UMA mode 0x13 for non-UMA mode
	F0x94[IsocRspData]	0x0
	F0x94[IsocNpReqData]	0x0
	F0x94[IsocRspCmd]	0x0
	F0x94[PReq]	0x0
	F0x94[IsocNpReqCmd]	0x7 for UMA mode 0x1 for non-UMA mode

Table 8-59 Onion Buffer Allocation for Non-internal Graphics – Family 11h Processor rev A

ASIC Rev	Register	Setting	Comments
RS780 All Revs	CPU_F3 x 1A4 [30:26]	0x1	DnHiRespBC
	CPU_F3 x 1A4 [25:21]	0x0	DnHiNpreqBC
	CPU_F3 x 1A4 [20:16]	0x0	DnHiPreqBC
	CPU_F3 x 1A4 [14:10]	0x12	DnLoRespBC
	CPU_F3 x 1A4 [9:5]	0x1	DnLoNpreqBC
	CPU_F3 x 1A4 [4:0]	0x4	DnLoPreqBC
	CPU_F3 x6C [27:24]	0x8	UpHiRespDBC
	CPU_F3 x6C [23;20]	0x0	UpHiNpreqDBC
	CPU_F3 x6C [19:16]	0x0	UpHiPreqDBC
	CPU_F3 x6C [11:8]	0x8	UpLoRespDBC
	CPU_F3 x6C [7:4]	0x1	UpLoNpreqDBC
	CPU_F3 x6C [3:0]	0x7	UpLoPreqDBC
	CPU_F3 x74 [27:24]	0x0	UpHiRespCBC
	CPU_F3 x74 [23;20]	0x1	UpHiNpreqCBC
	CPU_F3 x74 [19:16]	0x0	UpHiPreqCBC
	CPU_F3 x74 [11:8]	0x8	UpLoRespCBC
	CPU_F3 x74 [7:4]	0x7	UpLoNpreqCBC
	CPU_F3 x74 [3:0]	0x8	UpLoPreqCBC
	CPU_F3 x7C [31:28]	0xA	FreePoolBC
	CPU_F3 x7C [24]	0x1	DevTWBC
	CPU_F3 x7C [23;20]	0x1	HiPriNPBC
	CPU_F3 x7C [19:16]	0x0	HiPriPBC
	CPU_F3 x7C [15:12]	0x1	LowPriNPBC
	CPU_F3 x7C [11:8]	0x1	LowPriPBC
	CPU_F3 x 174 [7:4]	0x1	Cpu1BC
	CPU_F3 x 174 [3:0]	0x1	Cpu0BC
Note: These registers require a warm reset to take effect.			

Table 8-60 Onion Buffer Allocation for Non-internal Graphics – Family 11h Processor rev B (preliminary)

ASIC Rev	Register	Setting	Comments
RS780 All Revs	CPU_F3 x 1A4 [30:26]	0x1	DnHiRespBC
	CPU_F3 x 1A4 [25:21]	0x0	DnHiNpreqBC
	CPU_F3 x 1A4 [20:16]	0x0	DnHiPreqBC
	CPU_F3 x 1A4 [14:10]	0x12	DnLoRespBC
	CPU_F3 x 1A4 [9:5]	0x1	DnLoNpreqBC
	CPU_F3 x 1A4 [4:0]	0x4	DnLoPreqBC
	CPU_F3 x 1D4 [9:5]	0x0	DnDRRReqBC
	CPU_F3 x 1D4 [4:0]	0x8	DnPoolBC
	CPU_F3 x6C [27:24]	0x8	UpHiRespDBC
	CPU_F3 x6C [23;20]	0x0	UpHiNpreqDBC
	CPU_F3 x6C [19:16]	0x0	UpHiPreqDBC
	CPU_F3 x6C [11:8]	0x8	UpLoRespDBC
	CPU_F3 x6C [7:4]	0x1	UpLoNpreqDBC
	CPU_F3 x6C [3:0]	0x7	UpLoPreqDBC
	CPU_F3 x74 [31:28]	0x0	UpDRReqCBC
	CPU_F3 x74 [27:24]	0x0	UpHiRespCBC
	CPU_F3 x74 [23;20]	0x1	UpHiNpreqCBC
	CPU_F3 x74 [19:16]	0x0	UpHiPreqCBC
	CPU_F3 x74 [11:8]	0x8	UpLoRespCBC
	CPU_F3 x74 [7:4]	0x7	UpLoNpreqCBC
	CPU_F3 x74 [3:0]	0x8	UpLoPreqCBC
	CPU_F3 x7C [31:28]	0xC	FreePoolBC
	CPU_F3 x7C [27:24]	0x0	DispRefBC
	CPU_F3 x7C [23;20]	0x1	HiPriBC
	CPU_F3 x7C [19:16]	0x0	HiPriPBC
	CPU_F3 x7C [15:12]	0x1	LowPriBC
	CPU_F3 x7C [11:8]	0x1	LowPriPBC
	CPU_F3 x 174 [3:0]	0x1	CpuBC
Note: These registers require a warm reset to take effect.			

8.6.6 K8 Buffer Allocation Settings (Special Settings For UMA Mode)

The K8 processor uses an internal packet-based switch to route requests from HyperTransport and the CPU core to the memory controller. Buffers must be allocated for various request types, especially when running a UMA configuration, in order to support the display. The registers are described in the AMD BIOS and Kernel Developers Guide in section 3.6.8 – XBAR Flow Control Buffers, and section 13.6.9 – XBAR-to-SRI Buffer Counter Register. *Table 8-61* shows the AMD recommended buffer allocation. These setting are designed to provide maximum performance with AMD RS780 chipsets. The AMD BIOS checker may flag these registers as failing. This behavior is expected as long as they match the values in *Table 8-61* below.

Note: These registers require a warm reset to take effect.

Table 8-61 K8 Buffer Allocation Settings

Register	UMA Systems (UMA internal gfx, UMA internal gfx + external gfx, UMA + sideport internal gfx, UMA + sideport internal gfx + external gfx)	Non-UMA Systems (sideport internal gfx, external gfx, sideport internal gfx + external gfx)
D24 F0 x90	0x01700169	0x01700178
D24 F3 x70	0x51220111	0x51020111
D24 F3 x74	0x50404021	0x50008011
D24 F3 x78	0x08002A00	0x08002A00
D24 F3 x7C	0x0000212B (Single core CPU) 0x0000212A (Dual core CPU)	0x0000232B (Single core CPU) 0x0000232A (Dual-core CPU)
Note: These registers require a warm reset to take effect.		

8.6.7 Additional UMA Settings

AMD recommends additional register programming in their processor to support UMA systems, as documented in section 13.11 of their BIOS and Kernel Developer's Guide. These settings are repeated here for completeness and should be implemented if UMA graphics is being used. Many of these settings are common to non-UMA systems as well.

Table 8-62 Settings for Revision E and Earlier Processors

Register	Setting
D24 F0 x68 [28]	0x0
D24 F0 x68 [27:26]	0x3
D24 F0 x68 [11]	0x1
D24 F2 x94 [19]	0x1
D24 F2 x90 [27:25]	0x7
D24 F2 x90 [19]	0x1
D24 F2 x90 [15:14]	0x3
MSR C001_001F[36]	0x0
MSR C001_001F[9]	For revision CG processors: 0x1 For later revision processors: 0x0

Table 8-63 Settings for Revision F and G Processors

Register	Setting
D24 F0 x68 [28]	0x0
D24 F0 x68 [27:26]	0x3
D24 F0 x68 [11]	0x1
D24 F2 xA0 [5]	0x1
D24 F2 x94[27:24]	0x7
D24 F2 x90 [10]	0x0
D24 F2 xA0 [3:2]	0x3
MSR C001_001F[36]	0x0
MSR C001_001F[9]	0x0

8.6.8 Transmitter Deemphasis

The RS780 transmitter deemphasis setting that is used should be based upon the PCB trace length between the chipset and the processor. These settings will be programmed in NBCFG x a4 [26:24]. *Table 8-64* may be used as a guideline.

Table 8-64 Transmitter Deemphasis Versus Trace Length

ASIC Rev	Trace Length	Deemphasis Level	Deemphasis Code
RS780 All Revs	0 to 4.5”	-2 dB	0x1
	4.5” to 11”	-3 dB	0x2
	11” to 18”	-6 dB	0x3
	18+”	-9 dB	0x4

8.7 Power Management Settings

8.7.1 AMD Family 10h PMM Programming

Refer to the AMD Family 10h Processor BKDG for details on how to program the processor PMM registers. Specifically, refer to the programming tables “ACPI Power State Control Register SMAF Settings” with the descriptions for registers F3x[84:80].

8.7.2 AMD Family 11h PMM Programming

Refer to the AMD Family 11h Processor BKDG for details on how to program the processor PMM registers. Specifically, refer to the programming tables “ACPI Power State Control High” and “ACPI Power State Control Low” listed with the descriptions for registers F3x[84:80].

8.8 K8 PMM Programming

This section describes how to program the PMM registers in the K8. These are located in device 24, function 3, registers 80 and 84. “Mobile Setting” refers to any mobile platform that requires maximum power management support, regardless of whether the actual CPU is a mobile or a desktop variant. Generally, the mobile setting should be used if C1e/C3-state is enabled. The desktop setting should be used for all other systems. The “Cool’n’Quiet” feature, and optionally the C2-state, are still supported with the desktop settings.

Table 8-65 Power Management Settings

Register	Mobile Setting	Desktop Setting
PMM0	For rev F non-UMA systems: 0x61 For all other systems (UMA systems, or rev E or earlier non-UMA systems): 0x31	If C2 is enabled: 0x31 If C2 is disabled: 0x00
PMM1	See section *8.8.1*	See section *8.8.1*
PMM2	0x07	0x07
PMM3	For rev E and F processors in non-UMA systems: 0x6B For all other cases (rev CG and rev D processors or any UMA system): 0x33.	0x23
PMM4	0x13	0x13
PMM5	0x31	0x21
PMM6	0x13	0x13
PMM7	For UMA systems: 0x31 For non-UMA systems: 0x61	0x21

8.8.1 K8 PMM1 Programming

The PMM1 register should be programmed according to the information in *Table 8-66* regardless of whether a mobile or desktop CPU is being used.

Table 8-66 PMMI Programming Scenario DDR266 to DDR400

Scenario	DDR266	DDR333	DDR400
C1e/C3 Disabled	0x00	0x00	0x00
UMA GFX + C1e/C3 Enabled	0x03	0x23	0x33
UMA + Sideport GFX + C1e/C3 Enabled	0x03	0x23	0x33
No UMA GFX + C1e/C3 Enabled . Includes: - Sideport internal GFX - External PCIE GFX card and no internal GFX - External PCIE GFX card and sideport internal GFX	For rev CG and rev D processors: 0x63. For rev E and F processors: 0x6B.	For rev CG and rev D processors: 0x63 For rev E and F processors: 0x6B.	For rev CG and rev D processors: 0x63. For rev E and F processors: 0x6B.

Table 8-67 PMMI Programming Scenario DDR2-400 to DDR2-800

Scenario	DDR2-400	DDR2-533	DDR2-667	DDR2-800
C1e/C3 Disabled	0x00	0x00	0x00	0x00
UMA GFX + C1e/C3 Enabled	For rev F or G0: 0x37 For rev G1 or later: 0x33	For rev F or G0: 0x37 For rev G1 or later: 0x33	For rev F or G0: 0x37 For rev G1 or later: 0x33	For rev F or G0: 0x37 For rev G1 or later: 0x33
UMA + Sideport GFX + C1e/C3 Enabled	For rev F or G0: 0x37 For rev G1 or later: 0x33	For rev F or G0: 0x37 For rev G1 or later: 0x33	For rev F or G0: 0x37 For rev G1 or later: 0x33	For rev F or G0: 0x37 For rev G1 or later: 0x33
No UMA GFX + C1e/C3 Enabled Includes: - Sideport internal GFX - External PCIE GFX card and no internal GFX - External PCIE GFX card and sideport internal GFX	For rev F or G0: 0x67 For rev G1 or later: 0x6B	For rev F or G0: 0x67 For rev G1 or later: 0x6B	For rev F or G0: 0x67 For rev G1 or later: 0x6B	For rev F or G0: 0x67 For rev G1 or later: 0x6B

8.8.2 Low-Power HyperTransport Features

Table 8-68 Controlling BIAS Current for Receiver and Transmitter Settings

ASIC Rev	Register	Setting	Comments
RS780 All Revs	NBHTIUIND x 15 [31]	0x1	Receiver VBias is disabled when receiver termination is disabled
	NBHTIUIND x 15 [30]	0x1	Transmitter VBias is disabled when all lanes are in HiZ

8.8.2.1 HyperTransport 1

In HyperTransport 1 mode, the RS780 can tristate parts of the link in order to reduce power consumption. By default, all lanes are tristated. The CAD and CTL lanes must be tristated together, or there can be CAD, CTL, and CLK tristated.

Table 8-69 Tristating CAD and CTL Settings

ASIC Rev	Register	Setting	Comments
RS780 All Revs	NBCFG x C8 [13]	0x1	Enables HT Tristate in the chipset
	NBCFG x AC [8]	0x0	Disables HT CLK Tristate in the chipset
	CPU_F0 x 84 [13]	0x1	Enables HT Tristate in the CPU
	CPU_F0 x 170 [8]	0x0	Disables HT CLK Tristate in the CPU

Table 8-70 Tristating CAD, CTL, and CLK Settings

ASIC Rev	Register	Setting	Comments
RS780 All Revs	NBCFG x C8 [13]	0x1	Enables HT Tristate in the chipset.
	NBCFG x AC [8]	0x1	Enables HT CLK Tristate in the chipset.
	CPU_F0 x 84 [13]	0x1	Enables HT Tristate in the CPU.
	CPU_F0 x 170 [8]	0x1	Enables HT CLK Tristate in the CPU.
	CPU F3 x D8 [27:24]	TBD	Reconnects the Delay Timer in the CPU. Use 0x0 for bringup

Note for the SBIOS: For RS780 bringup, make the choice between Tristate Disable, Tristate CAD and CTL, and Tristate CAD, CTL, and CLK a SBIOS menu option.

8.8.2.2 HyperTransport 3

In HyperTransport 3 mode, the RS780 supports the LS1, the LS2, and the LS3 low-power states when LDTSTOP# is asserted, programmable through the LSSel register. The LS3 state is largely impractical to enable due to long reconnection latency. The system must enable either the LS1 or the LS2 low power states. The LS1 features low reconnection latency but higher power consumption relative to the LS2. In RS780 A11, the LS1 feature is not functional so the LS2 low power state must be used.

Table 8-71 LS1 Settings

ASIC Rev	Register	Setting	Comments
RS780 ASIC Revision A12 and after	NBCFG x AC [8:7]	0x0	Enables LS1 in the chipset.
	NBCFG x AC [22:21]	0x0	Enables LS1 in the chipset transmitter
	CPU_F0 x 170 [8]	0x0 (only for AMD Family 10h CPU)	Enables LS1 in the AMD Family 10h CPU.
	CPU_F0 x 170 [22:21]	0x0 (only for AMD Family 11h CPU)	Enables LS1 in the AMD Family 11h CPU transmitter
	CPU_F0 x 170 [8:7]	0x0 (only for AMD Family 11h CPU)	Enables LS1 in the AMD Family 11h CPU receiver
	PM_IO 0x8B	0x8	Minimum LDSTP# assertion

Table 8-72 LS2 Settings

ASIC Rev	Register	Setting	Comments
RS780 All Revs	NBCFG x AC [8:7]	0x2	Enables LS2 in the chipset.
	NBCFG x AC [22:21]	0x2	Enables LS2 in the chipset transmitter
	HTIUNBIND x 4B [11]	0x1	Gates the clock to the HTPHY during LS2
	HTIUNBIND x 15 [27]	0x1	Powers down the chipset DLLs in the LS2 state.
	CPU_F0 x 170 [8]	0x1(only for AMD Family 10h CPU)	Enables LS2 in the AMD Family 10h CPU.
	CPU_F0 x 170 [22:21]	0x2 (only for AMD Family 11h CPU)	Enables LS2 in the AMD Family 11h CPU transmitter
	CPU_F0 x 170 [8:7]	0x2 (only for AMD Family 11h CPU)	Enables LS2 in the AMD Family 11h CPU receiver

Note for the SBIOS: For RS780 bringup, make the choice between LS1 and LS2 a SBIOS menu option.

8.8.2.3 Inactive Lane State

When the HT link width is reduced below 16 bits in either direction, the electrical state of the inactive lanes will be controlled by the inactive lane state register.

Table 8-73 HyperTransport 1 Inactive Lane State Settings

ASIC Rev	Register	Setting	Comments
RS780 All Revs	NBCFG x AC[26:25]	0x0 for family 10h CPUs 0x1 for other processor families	Sets the chipset Inactive Lanes in the warm reset state for family 10 processors and into the PHY OFF state for other processor families
	NBCFG x AC[24:23]	0x0 for family 10h CPUs 0x1 for other processor families	Sets the chipset Inactive Lanes in the warm reset state for family 10 processors and into the PHY OFF state for other processor families
	CPU_F0 x 16C [7:6]	0x0 (only for family 10h processors)	Sets family 10h CPU Inactive Lanes in the warm reset state.
	CPU_F0 x 170 [26:25]	0x1(only for family 11h processors)	Sets family 11h CPU Transmitter Inactive Lanes in the PHY OFF state.
	CPU_F0 x 170 [24:23]	0x1(only for family 11h processors)	Sets family 11h CPU Receiver Inactive Lanes in the PHY OFF state.

Table 8-74 HyperTransport 3 Inactive Lane State Settings

ASIC Rev	Register	Setting	Comments
RS780 All Revs	NBCFG x AC[26:25]	0x0 for family 10h CPUs 0x1 for other processor families	Sets the chipset Inactive Lanes in the warm reset state for family 10 processors and into the PHY OFF state for other processor families
	NBCFG x AC[24:23]	0x0 for family 10h CPUs 0x1 for other processor families	Sets the chipset Inactive Lanes in the warm reset state for family 10 processors and into the PHY OFF state for other processor families
	CPU_F0 x 16C [7:6]	0x0 (only for family 10h processors)	Sets CPU Inactive Lanes in the warm reset state.
	CPU_F0 x 170 [26:25]	0x1(only for family 11h processors)	Sets family 11h CPU Receiver Inactive Lanes in the PHY OFF state.
	CPU_F0 x 170 [24:23]	0x1(only for family 11h processors)	Sets family 11h CPU Receiver Inactive Lanes in the PHY OFF state.

8.8.3 ATIVumaSysInfoRev3 Programming

The system BIOS is responsible for filling in the ATIVumaSysInfoRev3 table when the video BIOS calls GetIntegratedSystemInformation via INT 15h. The table parameters are used by the video BIOS and the video driver to support the “C1e/C3/Stutter Mode” feature and the “PowerNow!/Cool’nQuiet” feature. This section explains how to program the table entries k8SyncStartDly and k8DataRetTime.

The table parameter k8MclkMhz programming is covered in section *section 8.7“Power Management Settings,” on page 8-25*. The values of k8SyncStartDly and k8DataRetTime, given in *Table 8-75* and *Table 8-76*, are based on the system memory frequency and HyperTransport mode.

Table 8-75 K8 Values

System Memory Frequency	K8SyncStartDelay	K8DataRetTime
DDR266	269d	378d
DDR333	336d	290d
DDR400	394d	232d
DDR2-400	322d	286d
DDR2-533	322d	286d
DDR2-667	322d	286d
DDR2-800	322d	286d
Default	394d	378d

Table 8-76 AMD Family 10h and Family 11h Processors Preliminary Values

HyperTransport Mode	K8SyncStartDelay	K8DataRetTime
HyperTransport 1	100d	300d

HyperTransport Mode	K8SyncStartDelay	K8DataRetTime
HyperTransport 3 LS1	100d	150d
HyperTransport 3 LS2	100d	1300d
Default	100d	1300d

8.8.4 Generalized Stutter Mode

Generalized Stutter Mode (GSM) must be enabled when enabling the C1e low power state in the processor. This feature allows the chipset to reconnect the HyperTransport link when the processor is in the C1e state in order to pass DMA requests to memory.

Table 8-77 Register Settings for Isochronous Flow-Control Mode

ASIC Rev	Register	Setting	Comments
RS780 All Revs	NBMISCIND x C [13]	0x1	Enables GSM Mode.
	HTIUNBIND x 1C [31:17]	0xfff	Enables all traffic to be detected as GSM traffic.
	HTIUNBIND x 16 [15:10]	0x7	Enables all traffic to be detected as GSM traffic

8.9 Programming Guidelines

8.9.1 Debug Menu Features

These features should be exposed in the hidden SBIOS debug menu (even on customer systems). They assist in performing electrical margining to ensure a robust platform. Both AMD Family 10h and AMD Family 11h processors contain the same HyperTransport margining capabilities with the same register locations.

Table 8-78 Voltage Margining

ASIC Rev	Register	Setting	Comments
RS780 All Revs	NBCFG x A4 [18:16]	User Option from 0x0 (default) to 0x7	Enables RS780 transmitter attenuation. 0x0: No attenuation ... 0x4: Maximum attenuation Warm reset or LDTSTOP must be asserted before this setting takes effect.
	CPU_PHY x C5 and D5 [25:21]	User Option from 0x0 (default) to 0x1F	Enables processor transmitter attenuation 0x0: No attenuation … 0x1F: Maximum attenuation Register 0xC5 and 0xD5 should be set to the same value

Table 8-79 Time Margining

ASIC Rev	Register	Setting	Comments
RS780 All Revs	NBCFG x A8 [31]	User Option Enable (0x1) or Disable (0x0 – default)	Enables RS780 receiver time margining 0x0: Time Margining Off 0x1: Time Margining On Warm reset or LDTSTOP must be asserted before this setting takes effect.
	NBCFG x A8 [19:16]	User Option 0x0 (default) to 0xF	Sets RS780 time margining level 0x0: Minimum time margining … 0xF: Maximum time margining Warm reset or LDTSTOP must be asserted before this setting takes effect.
	HTIUNBIND X 15 [28]	(0x1)	HT receiver time margining mode 0x1: Allow margining only in operational, bist and loopback modes 0x0: Enable time margining whenever clock recovery is active, including training states
	CPU_PHY x C3 and D3 [30]	User Option Enable (0x1) or Disable (0x0 – default)	Enables processor time margining 0x0: Time Margining Off 0x1: Time Margining On
	CPU_PHY x C3 and D3 [29]	User Option Late (0x1) or Early (0x0 – default)	Set processor time margining direction 0x0: Early 0x1: Late
	CPU_PHY x C3 and D3 [28:23]	User Option 0x0 (default) to 0x3F	Set processor jitter on time 0x0: Minimum … 0x3F: Maximum
	CPU_PHY x C3 and D3 [15:10]	User Option 0x0 (default) to 0x3F	Sets processor jitter off time 0x0: Minimum … 0x3F: Maximum

Table 8-80 Receiver BIAS Select

ASIC Rev	Register	Setting	Comments
RS780 All Revs	HTIUNBIND x 4B [14:12]	User Option 0x0 (default) to 0x7	Enables RS780 receiver BIAS current control 0x0: Maximum … 0x7: Minimum Warm reset or LDTSTOP must be asserted before this setting takes effect.

Chapter 9
CLMC Programming

9.1 Global CLMC Settings

9.1.1 CLMC Enable

The CLMC feature, as a whole, is controlled by the register setting below. The CLMC is an HT3 feature and thus should be disabled whenever running in HT1. A warm system reset is required in order to enable the CLMC once the global register bit has been programmed. The setting below is required whenever any of the CLMC control features is enabled.

CLMC_I [31:0] NB_HTIUNBCFG:CLMC_I [R/W] 32 bits Access: 8/16/32 HTIUNBIND:0x50			
Field Name	**Bits**	**Setting**	**Description**
CLMC_En	0	1	Global CLMC enable Warm system reset required after programming 0: Disable 1: Enable

9.1.2 Default Inactive Lane State

A default link management mode setting is used by the system when CLMC is enabled but with its CDLC feature disabled. Part of the setting specifies the state of inactive lanes. This should be programmed to "LS2 mode" state. This must be programmed whenever CLMC is enabled.

Table 9-1 Receiver BIAS Select

ASIC Rev	Register	Bits	Setting	Comments
RS780 All Revs	NB_HT3_LINK_TRAINING_0 NBCFG:0xAC	[24:23]	0x3 This value must match the value in TxInLnSt F0 0x170 [26:25] of CPU. If mismatch found, the RS780 value takes precedence and CPU side should be re-programmed.	RxInLnSt: Receiver inactive lane state, set to LS2
	NB_HT3_LINK_TRAINING_0 NBCFG:0xAC	[26:25]	0x3 This value must match the value in RxInLnSt F0 0x170 [24:23] of CPU. If mismatch found, the RS780 value takes precedence and CPU side should be re-programmed.	TxInLnSt: Transmitter inactive lane state, set to LS2

9.2 Capability Registers

Although there exists one global register setting for enabling the CLMC, the CLMC does not exercise any of its features until the corresponding capability register is programmed. There are six main features in total. Below is the complete list of CLMC capability registers.

Note: The CDLR capability has two additional registers. These two registers exist in the same configuration space as the CDLR capability, but do not represent separate features. These correspond to the CDLR feature.

NB_HT3_Power_Management_data_port NB_NBCFG:NB_HT3_Power_management_data_port R/W] 32 bits Access: 8/16/32 nbconfig:0xfc			
Field Name	**Bits**	**Setting**	**Description**
CDLC_En	16	1	Sends the CLMC LMM value. Clears on RESET and POWERGOOD. 0: Disable 1: Enable
CDLD_En	17	1	Enables CLMC link disconnect feature. Clears on RESET and POWERGOOD. 0: Disable 1: Enable
CDLW_En	18	1	Sends the CLMC up/down link width values. Clears on POWERGOOD only. 0: Disable 1: Enable
CDLF_En	19	Program to 1 only if CPU is Turion or Turion Ultra and has F3x1FC[30] =1 Program to 0 otherwise	Sends the CLMC link frequency value. Clears on POWERGOOD only. 0: Disable 1: Enable
CILR_En	20	1	Enables CLMC inactive lane refresh feature. Clears on POWERGOOD only. 0: Disable 1: Enable
CDLR_En	21	1	Enables CLMC link refresh feature. Clears on RESET and POWERGOOD. 0: Disable 1: Enable
CDLRInt	[11:6]	0x35	LDTSTOP# assertion time when CDLR is enabled, 9.6ms Clears on RESET and POWERGOOD.
CDLRLen	[5:0]	0x23	LDTSTOP# de-assertion time when CDLR is enabled, 6us Clears on RESET and POWERGOOD.
CDLFLinkFreqEn	n/a	n/a	Not used in the RS780 Clears on RESET and POWERGOOD.
Note: CDLF and CDLW are only available when an IGP driver is loaded, and as such are not supported on systems which implement discrete GFX since a non-IGP driver is used in these configurations. In addition, CDLF is only available on Turion and Turion Ultra CPUs, which have F3x1FC[30] = 1 as indicated above.			

9.2.1 Programming Sequence

Different fields of the 'NB_HT3_Power_Managment_data_port[31:0]' register are accessible depending on what index value is programmed in the capability register 'NB_HT3_Power_Management_capability [31:0]'. Below is how the capability register should be programmed.

NB_HT3_Power_Management_capability NB_NBCFG:NB_HT3_Power_management_Capability [R/W] 32 bits Access: 8/16/32 nbconfig:0xf8			
Field Name	**Bits**	**Setting**	**Description**
Reg_Ind	19:16	0x0	0x0: Provides access to all the CLMC capability registers 0x2: Provides access to the CDLRInt[5:0] and CDLRLen[5:0] registers

Once the index value has been programmed, the corresponding CLMC capability registers can then be programmed. Field names and bit mapping are as shown in the table in section *9.2*.

9.3 Sub-Feature Registers

Outside of the capability registers, the CLMC has several registers which pertain to each of its six main features. Some of these registers exist in the NBMCIND configuration space and require special consideration.

9.3.1 Programming the NBMCIND Registers

When programming any register field contained in the NBMCIND 0x29 and 0x2A CLMC registers, the following programming sequence below must be followed every time.

- Step 1: Program the desired fields in the CLMC registers NBMCIND 0x29 and/or 0x2A
- Step 2: Program the UpdateMCRegs and MCRegsUpdateQual registers to value 0x1 at the same time (but after Step 1). This will propagate the new values in the updated fields.
- Step 3: Program the UpdateMCRegs and MCRegsUpdateQual registers back to 0x0. Clearing these registers removes the risk of propagating values undesirably.

HT_ARB_II [31:0] NB_MCCFG:HT_ARB_II [R/W] 32 bits Access: 8/16/32 NBMCIND:0x2C			
Field Name	**Bits**	**Setting**	**Description**
UpdateMCRegs	[20]	0	Used to pass new CLMC NBMCIND register values 0: Retain current settings 1: Propagate new programmed values
MCRegsUpdateQual	[21]	0	Extra qualifier used when passing new CLMC NBMCIND register values 0: Retain current settings 1: Propagate new programmed values

9.4 CLMC Control Features

9.4.1 CDLD (Centralized Dynamic Link Disconnection)

The CDLD feature disconnects the HT link whenever it deems the link to be idle, among other conditions. This feature is intended to be enabled along with CDLR, although the flexibility exists for CDLD to run with or without CDLR. The HT link is disconnected by asserting the LDTSTOP# signal, and reconnected by de-asserting the LDTSTOP# signal. A minimum assertion and de-assertion time must be obeyed on every link disconnection and re-connection. These timer values are programmable.

CLMC_CONTROL_II [31:0] NB_HTIUNBCFG:CLMC_CONTROL_II [R/W] 32 bits Access: 8/16/32 HTIUNBIND:0x53			
Field Name	**Bits**	**Setting**	**Description**
MinLdtStopOnTime	[17:0]	0xB	Minimum LDTSTOP assertion time, 1us

CLMC_CONTROL_III [31:0] NB_HTIUNBCFG:CLMC_CONTROL_III [R/W] 32 bits Access: 8/16/32 HTIUNBIND:0x54			
Field Name	**Bits**	**Setting**	**Description**
MinLdtStopOffTime	[17:0]	0x3C	Minimum LDTSTOP de-assertion time, 6us

Note: When CDLD is enabled along with CDLR, the minimum LDTSTOP assertion and de-assertion times are extended by using the CDLR timer registers. However, the registers above remain unchanged since the CLMC will still use them when performing CDLC, CDLW, and CDLF.

9.4.2 CDLC (Centralized Dynamic Link Configuration)

The CDLC feature dynamically changes HT link configuration parameters based on changes in system activity. The various configuration parameters are contained in Link Management Mode (LMM) registers. Each register is 32 bits wide and the CLMC uses seven of these types of registers: LMM0[31:0], and LMM1[31:0] through LMM6[31:0]. All registers are identical in terms of the type of information they hold; corresponding register fields are mapped the same way. The LMM0 register holds the boot-up default configuration parameters; its values are not contained in contiguous register configuration space. The LMM1-LMM6 registers contain the programmable information intended for use by CDLC; the contents of each exist in contiguous register configuration space.

The definition of the RS780 LMM registers is given in *Table 9-2* below.

Table 9-2 LMM Registers

LMM1 - LMM6 [31:0] HTIUNBIND:0x70 - 0x75				
ASIC Rev	**LMMi [31:0] Field Name**	**Bits**	**Default**	**Comments**
RS780 All Revs	LMMi_T0Time	[5:0]	0	T0 training time
	LMMi_FullT0Time	[11:6]	0	Full T0 training time
	LMMi_RxInLnSt	[13:12]	0	RX inactive lane state
	LMMi_TxInLnSt	[15:14]	0	TX inactive lane state
	LMMi_RxLSSel	[17:16]	0	RX link state select
	LMMi_TxLSSel	[19:18]	0	TX link state select
	LMMi_Deemph	[24:20]	0	Deemphasis setting
	LMMi_HiZMode	[25]	0	High impedance mode
	LMMi_TermDis	[26]	0	Termination disable
	LMMi_LS2DLLPwrDn	[27]	0	DLL power down in LS2
	LMMi_RxVBControl	[28]	0	RX vbias control
	LMMi_TxVBControl	[29]	0	TX vbias control
	LMMi_Reserved	[31:30]	0	Reserved

Corresponding LMM registers exist on the CPU core side, although the fields do not map in the exact same way. Each LMM register of the CPU core must be programmed with the same information as its RS780 counterpart. All registers on both sides must be programmed in the SBIOS on boot-up, and cannot be re-programmed thereafter unless a warm or cold

system reset is performed.

The CLMC dynamically switches between different LMM registers by sending an index value ranging from 0x0 to 0x6. This index is used to specify which LMM register should be applied next. The new LMM setting is applied on the subsequent LDTSTOP# assertion.

The recommended programming of the CDLC feature is given in *Table 9-3* below.

Table 9-3 LMM Register Programming

ASIC Rev	Register	Bits	Setting	Comments
RS780 All Revs	HTIUNBIND: 0x70-0x75	**LMM1-6 [31:0]**		
		LMMi_T0Time [5:0]	For register LMM3 (0x72), program to be the same as CPU F0x16C[22:17] For all other registers, program to be the same as NBCFG:0xA0 [5:0]	T0 training time
		LMMi_FullT0Time [11:6]	Program to be the same as NBCFG:0xA0 [22:17]	Full T0 training time
		LMMi_RxInLnSt [13:12]	11	RX inactive lane state
		LMMi_TxInLnSt [15:14]	11	TX inactive lane state
		LMMi_RxLSSel [17:16]	10	RX link state select
		LMMi_TxLSSel [19:18]	10	TX link state select
		LMMi_Deemph [24:20]	Program to be the same as NBCFG: 0xA4 [26:24]	De-emphasis setting
		LMMi_HiZMode [25]	0	High impedance mode
		LMMi_TermDis [26]	0	Termination disable
		LMMi_LS2DLLPwrDn [27]	1	DLL power down in LS2
		LMMi_RxVBControl [28]	1	RX vbias control
		LMMi_TxVBControl [29]	1	TX vbias control
		LMMi_Reserved [31:30]	00	Reserved

Table 9-4 CDLC Setting for CPU F3xD4 Register

Register	Bits	Setting	Comments
F3 0xD4	18	1	Smaf7Dis 0: Set configuration parameters based on SMAF settings 1: Set configuration parameters based on LMM registers (CLMC mode)

Table 9-5 CPU Core LMM Register Programming

Fcm	Offset	Index	Register	Setting	Description
4	Index/Data 0x170/174	1	**LMM1 [31:0]**		
			LmmT0Time [5:0]	Program to be the same as F0x16C[T0Time] [5:0]	T0 training time
			LmmForceFullT0 [8:6]	Program to be the same as F0x16C [ForceFullT0] [15:13]	Force full T0 training time
			LmmRxInLnSt [10:9]	Program to be the same as LMM1.LMMi_TxInLnSt (HTIUNBIND:0x70) [15:14]	Receive inactive lane state
			LmmTxInLnSt [12:11]	Program to be the same as LMM1.LMMi_RxInLnSt (HTIUNBIND:0x70) [13:12]	Transmit inactive lane state
			Reserved [13]	0	Reserved
			LmmCpuPrbEn [14]	Program to be the same as F3x80 [SMAF1.CpuPrbEn] [8]	Memory probe enable
			LmmDramSr [15]	Program to be the same as F3x80 [SMAF1.DramSr] [9]	DRAM self refresh enable
			LmmDramMemClkTri [16]	Program to be the same as F3x80 [SMAF1.DramMemClkTri] [10]	Memory clock tri-state enable
			LmmCpuAltVidEn [17]	0	Alt VID enable
			LmmCpuDid [20:18]	100 for internal graphics 111 for external graphics	CPU DID setting. Should be set to /16 - /1 if Alt-VID is disabled; set to "clocks off" for external graphics
			LmmDeemph [23:21]	001 This value must match the de-emphasis setting which F4x184_x[D5,C5] represents	CPU TX post cursor de-emphasis value 000: No de-emphasis 001: -3 dB 010: -6 dB 011: -8 dB
			LmmRxLSSel [25:24]	Program to be the same as LMM1.LMMi_TxLSSel (HTIUNBIND:0x70) [19:18]	CPU receiver LS state selection
			LmmTxLSSel [27:26]	Program to be the same as LMM1.LMMi_RxLSSel (HTIUNBIND:0x70) [17:16]	CPU transmitter LS state selection
			LmmDfeVoltage [29:28]	Program to be the same as F4x184_x[D4,C4] [DfeVoltage] [6:5]	Decision feedback equalization setting
			LmmDfeEn[30]	Program to be the same as F4x184_x[D4,C4] [DfeEn] [7]	Decision feedback equalization enable
			Reserved [31]	0	Reserved

4	Index/Data 0x170/174	2 - 6	**LMM2-6 [31:0]**		
			LmmT0Time [5:0]	For index value 0x3 (LMM3), program to be the same as CPU F0x16C[22:17] For all other index values, program to be the same as F0x16C[T0Time] [5:0]	T0 training time
			LmmForceFullT0 [8:6]	Program to be the same as F0x16C [ForceFullT0] [15:13]	Force full T0 training time
			LmmRxInLnSt [10:9]	Program to be the same as LMMx.LMMi_TxInLnSt (HTIUNBIND: 0x71-75) [15:14]	Receive inactive lane state
			LmmTxInLnSt [12:11]	Program to be the same as LMMx.LMMi_RxInLnSt (HTIUNBIND:0x71-75) [13:12]	Transmit inactive lane state
			Reserved [13]	0	Reserved
			LmmCpuPrbEn [14]	Program to be the same as F3x80 [SMAF1.CpuPrbEn] [8]	Memory probe enable
			LmmDramSr [15]	Program to be the same as F3x80 [SMAF1.DramSr] [9]	DRAM self refresh enable
			LmmDramMemClkTri [16]	Program to be the same as F3x80 [SMAF1.DramMemClkTri] [10]	Memory clock tri-state enable
			LmmCpuAltVidEn [17]	0	Alt VID enable
			LmmCpuDid [20:18]	111	CPU DID setting. Should be set to /512 or Clks off for Alt-VID enabled; still has effect if Alt-VID disabled.
			LmmDeemph [23:21]	001 This value must match the de-emphasis setting which F4x184_x[D5,C5] represents	CPU TX post cursor de-emphasis value 000: No de-emphasis 001: -3 dB 010: -6 dB 011: -8 dB
			LmmRxLSSel [25:24]	Program to be the same as LMMx.LMMi_TxLSSel (HTIUNBIND:0x71-75) [19:18]	CPU receiver LS state selection
			LmmTxLSSel [27:26]	Program to be the same as LMMx.LMMi_RxLSSel (HTIUNBIND:0x71-75) [17:16]	CPU transmitter LS state selection
			LmmDfeVoltage [29:28]	Program to be the same as F4x184_x[D4,C4] [DfeVoltage] [6:5]	Decision feedback equalization setting
			LmmDfeEn[30]	Program to be the same as F4x184_x[D4,C4] [DfeEn] [7]	Decision feedback equalization enable
			Reserved [31]	0	Reserved

Table 9-6 Additional CDLC Register Programming

ASIC Rev	Register	Bits	Setting	Comments
RS780 All Revs	CLMC_LMM_St1 HTIUNBIND:0x55	[20:3]	0x3E8 for internal graphics 0x3FFFF for external graphics	LMMTimerVal1: State transition timer
		[2:0]	0x3	LMMTrafficSel1: Selects traffic types to monitor
	CLMC_LMM_St2 HTIUNBIND:0x56	[20:3]	0x2000	LMMTimerVal2: State transition timer
		[2:0]	0x7	LMMTrafficSel2: Selects traffic types to monitor
	CLMC_LMM_St3 HTIUNBIND:0x57	[20:3]	0x3E8	LMMTimerVal3: State transition timer
		[2:0]	0x7	LMMTrafficSel3: Selects traffic types to monitor
	CLMC_LMM_St4 HTIUNBIND:0x58	[20:3]	0x1388	LMMTimerVal4: State transition timer
		[2:0]	0x3	LMMTrafficSel4: Selects traffic types to monitor
	CLMC_LMM_St5 HTIUNBIND:0x59	[20:3]	0x3E8	LMMTimerVal5: State transition timer
		[2:0]	0x7	LMMTrafficSel5: Selects traffic types to monitor
	CLMC_LMM_St6 HTIUNBIND:0x5A	[20:3]	0x3E8	LMMTimerVal6: State transition timer
		[2:0]	0x7	LMMTrafficSel6: Selects traffic types to monitor
	NB HT3 Power Management Data Port NBCFG:0xFC	16	1	CDLC_En: Sends the CLMC LMM value. Clears on RESET and POWERGOOD. 0: Disable 1: Enable
	HT_CLMC_I NBMCIND:0x29	[20:19]	0x1 for internal graphics 0x0 for external graphics	LMMSel: Chooses the source of the next LMM value. 0x0: CLMC calculated 0x1: Boot-up configuration (LMM0) 0x2: Software override 0x3: Microcontroller
	HT_CLMC_II NBMCIND:0x2A	10	0	LookAtFBC: Monitors the state of FBC. 0: Ignore FBC status (use for UMA mode) 1: Include FBC status (used for non-UMA mode)
	CLMC_BWESTM_BwRange1 HTIUNBIND:0x65	0	1	CheckLMM: Checks for presence of new LMM value when deciding link disconnect. 0: Do not check 1: Check
	CLMC_BWESTM_Timer1 HTIUNBIND:0x68	[17:0]	0x25	IdleTimerVal: Measures idle periods of GSM traffic.

9.4.3 CDLW (Centralized Dynamic Link Width)

9.4.3.1 CDLW Software Mode

When CDLW is configured to run in software mode, the upstream and downstream link width values applied to the HT link are based on programmable registers. Software (driver) determines the actual link width values to program and when to do so. Once new values are detected by the CLMC hardware, the hardware applies the programmed values to the HT link.

On boot-up, the CLMC link width registers should be initialized to match the default HT configuration link width values. The settings below configure the CDLW software mode of operation.

The software mode of CDLW is the recommended mode of operation.

ASIC Rev	RS780 Register	Bits	Setting	Comments
RS780 All Revs	NB_NBCFG:NB_HT3_Power_management_data_port NBCFG:0xFC	18	1	CDLW_En: Sends the CLMC up/down link width values. Clears on POWERGOOD only.
	HT_CLMC_I NBMCIND:0x29	[22:21]	0x2	LWSel: Makes the source of the next upstream and downstream LW values the register values.
	HT_CLMC_I NBMCIND:0x29	[7:5]	On boot-up, program to match NBCONFIG 0xC8 [30:28]	RegLWup: programmable upstream LW; value adheres to HT specification and is determined by software.
	HT_CLMC_I NBMCIND:0x29	[10:8]	On boot-up, program to match NBCONFIG 0xC8 [26:24]	RegLWdn: programmable downstream LW; value adheres to HT specification and is determined by software.

9.5 CLMC Refresh Features

9.5.1 CDLR (Centralized Disconnected Link Refresh)

The CDLR feature extends the LDTSTOP# assertion and/or de-assertion time by some programmable amount. The programmable amounts are contained in the CDLRInt[5:0] and CDLRLen[5:0] registers. Refer to section *9.2* for details on the CDLR timer registers.

9.5.2 CILR (Centralized Inactive Lane Refresh)

The CILR feature periodically "refreshes" inactive HT lanes by sending T0 packets across these lanes. The frequency of the refreshing is determined by a programmable timer. CILR is performed on the upstream HT link but may also be performed on the downstream link.

A lane is deemed to be "inactive" if it is currently not in use, and is within the link width set at system boot-up. For example, if the boot-up HT link width is 8-bit, and then CLMC changes the link width to 4-bit, then 4 lanes become inactive (not 12, since the boot-up link width was 8-bit, not 16-bit)

NB_HT3_Power_Management_data_port NB_NBCFG:NB_HT3_Power_management_data_port · [R/W] · 32 bits · Access: 8/16/32 · nbconfig:0xfc			
Field Name	**Bits**	**Setting**	**Description**
CILR_En	20	1	Enable CLMC inactive lane refresh feature. Clears on POWERGOOD only. 0: Disable 1: Enable

CLMC_CONTROL_I [31:0] NB_HTIUNBCFG:CLMC_CONTROL_I [R/W] 32 bits Access: 8/16/32 HTIUNBIND:0x52			
Field Name	**Bits**	**Setting**	**Description**
CILRTimerVal	[23:6]	0x186A0	Time to wait before doing next CILR, 10ms

HT_CLMC_il [31:0] NB_MCCFG:HT_CLMC_Il [R/W] 32 bits Access: 8/16/32 NBMCIND:0x2A			
Field Name	**Bits**	**Setting**	**Description**
LookAtInactiveRX	[9]	0	Considers inactive RX lanes during CILR 0: Exclude 1: Include

9.6 CLMC Stutter Mode

When running a system with CDLD and CDLR, the HT link can be disconnected and reconnected as often as possible but as dictated by the system display. This frequent link disconnection is referred to as CLMC stutter mode and is the replacement of the legacy C1e feature.

The settings below are used to configure a system to run with CLMC stutter mode alone or together with the CILR feature.

CLMC_I [31:0] NB_HTIUNBCFG:CLMC_I [R/W] 32 bits Access: 8/16/32 HTIUNBIND:0x50			
Field Name	**Bits**	**Setting**	**Description**
CpuCores	[4:1]	Program to {number of cores minus 1}	Number of CPU cores (program 1 less than actual). For example, for a dual core CPU, program 0x1.
HaltTimerVal	[30:13]	0x15	Time to wait after all CPU cores have halted, 2.1us

NB_HT3_Power_Management_data_port NB_NBCFG:NB_HT3_Power_management_data_port [R/W] 32 bits Access: 8/16/32 nbconfig:0xfc			
Field Name	**Bits**	**Setting**	**Description**
CDLD_En	17	1	Enables CLMC link disconnect feature. Clears on RESET and POWERGOOD. 0: Disable 1: Enable
CDLR_En	21	1	Enables CLMC link refresh feature. Clears on RESET and POWERGOOD. 0: Disable 1: Enable
CDLRInt	[11:6]	0x35	LDTSTOP# assertion time when CDLR is enabled, 9.6ms Clears on RESET and POWERGOOD.
CDLRLen	[5:0]	0x23	LDTSTOP# de-assertion time when CDLR is enabled, 6us Clears on RESET and POWERGOOD.

ASIC Rev	Register	Bits	Setting	Comments
RS780 All Revs	HT_CLMC_I NBMCIND:0x29	[20:19]	0x1	LMMSel: Chooses the source of the next LMM value 0x0: CLMC calculated 0x1: Boot-up configuration (LMM0) 0x2: Software override 0x3: Microcontroller
	HT_CLMC_I NBMCIND:0x29	[22:21]	0x1	LWSel: Chooses the source of the next up/down LW value 0x0: CLMC calculated 0x1: Boot-up configuration 0x2: Software override 0x3: Microcontroller
	HT_CLMC_I NBMCIND:0x29	[24:23]	0x0	FreqSel: Chooses the source of the next frequency value 0x0: Boot-up configuration 0x1: Software override 0x2: Microcontroller 0x3: Microcontroller

ASIC Rev	Register	Bits	Setting	Comments
RS780 All Revs	CLMC_BWESTM_BwRange1 HTIUNBIND:0x65	0	0/1	CheckLMM: Checks for new LMM when deciding link disconnection. 0: For programming stutter mode only—do not check for new incoming LMM value. 1: If CDLC_En=1, check for new incoming LMM value.
	CLMC_BWESTM_BwRange1 HTIUNBIND:0x65	1	0/1	CheckLW: Checks for new LW when deciding link disconnection. 0: For programming stutter mode only—do not check for new incoming LW value. 1: If CDLW_En=1, check for new incoming LW value.
	CLMC_BWESTM_BwRange1 HTIUNBIND:0x65	2	0/1	CheckFreq: Checks for new frequency when deciding link disconnection 0: For programming stutter mode only—do not check for new incoming frequency value. 1: If CDLF_En=1, check for new incoming frequency value.

CLMC_BWESTM_BwRange1 [31:0] NB_HTIUNBCFG:CLMC_BWESTM_BwRange1 · [R/W] · 32 bits · Access: 8/16/32 · HTIUNBIND:0x65			
Field Name	**Bits**	**Settings**	**Description**
CheckLMM	0	0/1	Check for new LMM when deciding link disconnection 0: For programming stutter mode only—do not check for new incoming LMM value. 1: If CDLC_En=1, check for new incoming LMM value.
CheckLW	1	0/1	Check for new LW when deciding link disconnection 0: For programming stutter mode only—do not check for new incoming LW value. 1: If CDLW_En=1, check for new incoming LW value.
CheckFreq	2	0/1	Check for new frequency when deciding link disconnection 0: For programming stutter mode only—do not check for new incoming frequency value. 1: If CDLF_En=1, check for new incoming frequency value.

Appendix A
Revision History

Rev 1.01 (August 2009)

- Modified cover title.
- Added marketing names to the ASIC variants in section 1.1.

Rev 1.00 (July 2009)

- First public release based on OEM rev 1.07.