PEX 8311RDK

Optimizing PEX 8311 PCI Express-to-Local Bus DMA Performance Design Note
Revision 1.1
February, 2007

Optimizing PEX 8311 PCI Express-to-Local Bus DMA Performance PEX 8311RDK Design Note Documentation

A. Affected Silicon Revision

This document details the design note for the following silicon:

Product	Revision	Description	Status
PEX 8311	AA	PCI Express-to-Generic Local Bus Bridge	Production

B. Documentation Revision

The following documentation supports this Design Note:

Document	Version	Description	Publication Date
PEX 8311AA Data Book	0.90	Data Book	April, 2006
PEX 8311 RDK Hardware Reference Manual	0.90	Hardware Reference Manual	December, 2005

C. Design Note Documentation Revision History

Revision	Publication Date	Description
1.0	December, 2005	Baseline.
1.1	February 15, 2007	Rewrote document for clarity. Added last paragraph to Conclusion.

D. Design Note Summary

#	Description
1	Overview
2	PCI Express-to-Local Bus DMA Key Parameters
3	PCI Express-to-Local DMA Performance Measure Example
4	Conclusion

1. Overview

This design note is directed at systems based on the PLX PEX 8311 bridge device. It describes how to modify PEX 8311 registers for optimal PCI Express-to-Local DMA transfer throughput. Examples are provided that demonstrate how the PEX 8311RDK and PCI SDK utilities can be used to measure DMA performance for various combinations of Transfer Sizes and Configuration register settings.

2. PCI Express-to-Local Bus DMA Key Parameters

Description

Three fundamental factors govern PEX 8311 DMA transfer throughput when transferring in the PCI Express-to-Local direction. These factors are:

- Transfer Size
- PCI Express Read Prefetch
- PCI Express Maximum Read Request Size

Transfer Size

The first factor, fundamental for either direction, is Transfer Size. The PEX 8311 DMA channels are equipped with 256-byte-deep FIFOs, which allow fully independent, asynchronous, and concurrent operation of the PCI Express and Local interfaces. Theoretically, maximum throughput is achieved when both interfaces are transferring long, sustained bursts. The larger the Transfer Size, the lower the bus and CPU overhead.

PCI Express Read Prefetch

The second factor, specific to the PCI Express-to-Local direction, is that the PEX 8311 DMA engines initiate Read transactions on the PCI Express interface. Because data is not immediately available, the initial Memory Read is delayed by the latency time required to fetch the data from system memory. Because DMA transfers read from sequential addresses, the PEX 8311's Prefetch feature must be used to prevent Read latency penalties on successive Reads from PCI Express memory.

The PCI Express Read Prefetch Size is specified in the PCI Express Configuration Space **PCI Control** register *Programmed Prefetch Size* field (offset 100Ch, **PCICTL**[29:27]). The *Programmed Prefetch Size* field defines the number of bytes fetched from PCI Express memory, in response to a Direct Master or DMA Read. By default, this field is 000b, meaning that each DMA or Direct Master Read fetches only the data requested. A value of 111b, on the other hand, causes 4-KB of sequential data to be fetched on the first Read. In this way, the initial latency penalty occurs only once for every 4 KB transferred.

Select the Maximum Prefetch Size possible for the given transfer. However, the Prefetch Size should not exceed the Transfer Size. Ideally, Prefetch Size should divide evenly into the Transfer Size, and transfers should be aligned to quad-word (Qword) boundaries. If the Prefetch Size is greater than the number of bytes read, performance is impacted by the reading and discarding of excess Prefetch data.

PCI Express Maximum Read Request Size

The third factor that governs PEX 8311 DMA transfer throughput is that for PCI Express transactions, the largest Payload Size yields the greatest efficiency when moving large blocks of data. The PCI Express Configuration Space **PCI Express Device Control** register *Maximum Read Request Size* field (offset 68h, **DEVCTL**[14:12]) specifies the maximum size of Read requests issued on the PCI Express port.

By default, this field is set to 010b, or 512 bytes. However, Read Request sizes of up to 4-KB are possible, yielding greater efficiency in conjunction with the 4-KB Read Prefetch Size option previously described. For Read Prefetch Sizes greater than 512 bytes, increase the *Maximum Read Request Size* field, accordingly.

3. PCI Express-to-Local DMA Performance Measure Example

This section describes how to use the PEX 8311RDK board and *PLX PCI SDK, Version 4.4* or higher, to estimate PCI Express-to-Local Bus transfer performance. The **PLXMon** utility application (part of the SDK) includes a GUI for measuring transfer throughput for various Transfer Sizes. Before using **PLXMon** to measure performance in the Read direction (PCI Express-to-Local), certain PCI Express Configuration registers must be set manually; otherwise, maximum DMA transfer rates cannot be attained by the **Performance Measure** GUI. In this example, we use the **PLXMon** utility to first manually configure the PCI Express Prefetch and maximum PEX 8311 Read Request Sizes in the PCI Express Configuration Space registers, and then provide instructions for using the GUI to measure performance for 1-MB transfers from PCI Express-to-Local SBSRAM on the PEX 8311 RDK board.

Configuring PCI Express Configuration Space Registers

The PEX 8311 is a single physical device; however, from a software perspective, it is two devices, PEX 8111 and PCI 9056. Setup of PCI Express-to-Local DMA transfers requires writing to the registers in both devices. For optimal throughput for a 1-MB transfer, it is necessary to select maximum Prefetch and Read Request Sizes in the PEX 8311 PCI Express Configuration Space registers.

To configure the Prefetch and Maximum Read Request Sizes

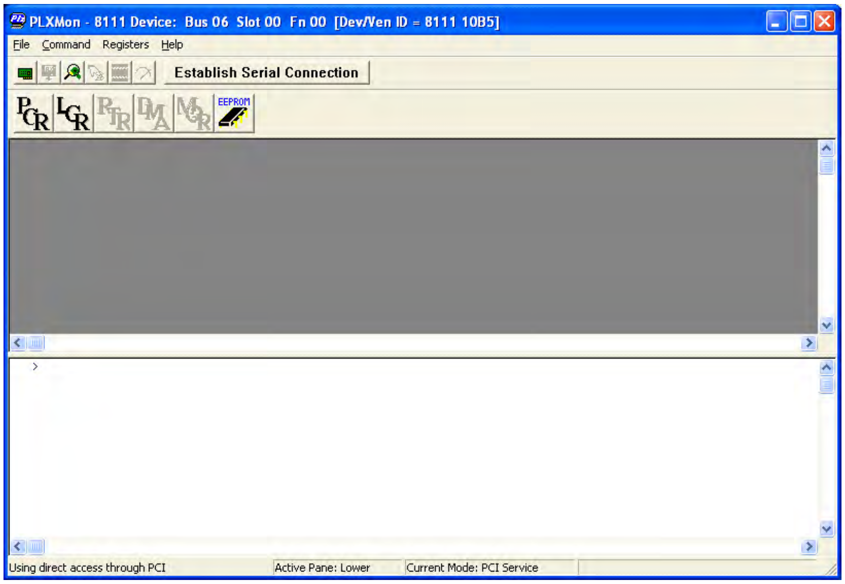
a. Start **PLXMon**.

When **PLXMon** starts, the program window opens, which displays the first PLX device it finds in the system.

There are two GUI screens (upper and lower) for selecting and modifying various Configuration registers. Registers can also be accessed by way of the Command Line interface on the lower half of the program window. (Refer to the *PLX SDK User Manual* for details.)

b. From the **Command** menu, select **Select Device**, then select **PEX 8111**.

PEX 8111 is selected, because the PEX 8311 PCI Express Configuration Space registers appear as a PEX 8111 device in **PLXMon**. When selected, the display should look similar to Figure 1.



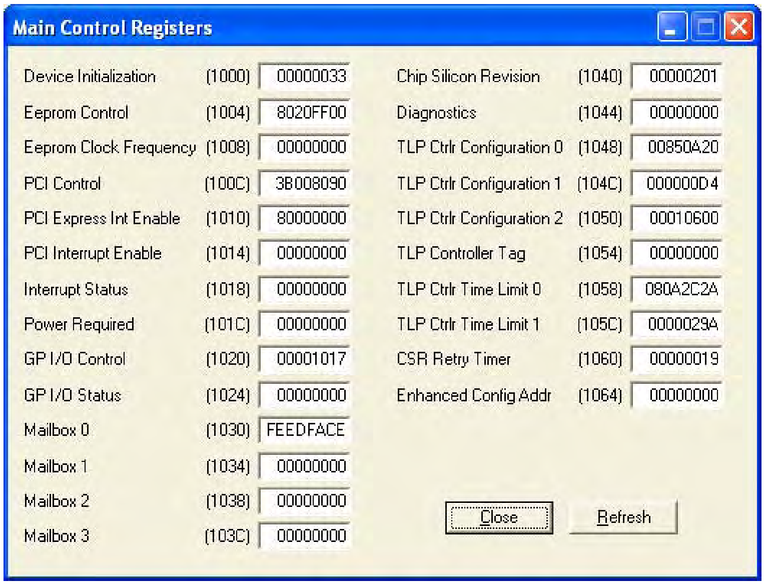
PLXMon - 8111 Device: Bus 06 Slot 00 Fn 00 [Dev/Ven ID = 8111 10B5]

Using direct access through PCI | Active Pane: Lower | Current Mode: PCI Service

Figure 1. PLXMon Utility with PEX 8311 (8111) Express Configuration Space Registers Selected (Program Window)

c. Click the button on the **PLXMon** program window, to open the **Main Control Registers** dialog box (Figure 2). The values shown are in hex.

d. Change **PCI Control (100C)** to read *3Bnnnnnn*, where *n* is don't care, to select a Prefetch Count of 4 KB. Click the **Refresh** button to confirm and return to the program window.



Main Control Registers

Device Initialization	(1000)	00000033	Chip Silicon Revision	(1040)	00000201
Eeprom Control	(1004)	8020FF00	Diagnostics	(1044)	00000000
Eeprom Clock Frequency	(1008)	00000000	TLP Ctrlr Configuration 0	(1048)	00850A20
PCI Control	(100C)	3B008090	TLP Ctrlr Configuration 1	(104C)	000000D4
PCI Express Int Enable	(1010)	80000000	TLP Ctrlr Configuration 2	(1050)	00010600
PCI Interrupt Enable	(1014)	00000000	TLP Controller Tag	(1054)	00000000
Interrupt Status	(1018)	00000000	TLP Ctrlr Time Limit 0	(1058)	080A2C2A
Power Required	(101C)	00000000	TLP Ctrlr Time Limit 1	(105C)	0000029A
GP I/O Control	(1020)	00001017	CSR Retry Timer	(1060)	00000019
GP I/O Status	(1024)	00000000	Enhanced Config Addr	(1064)	00000000
Mailbox 0	(1030)	FEEDFACE			
Mailbox 1	(1034)	00000000			
Mailbox 2	(1038)	00000000			
Mailbox 3	(103C)	00000000			

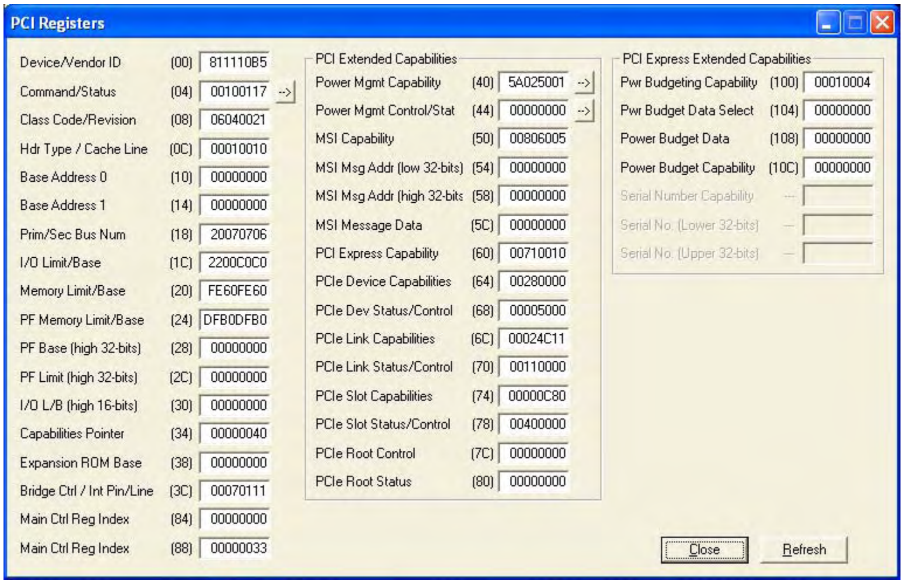
Figure 2. PEX 8311 Main Control Registers Dialog Box

Next, the PCI Express Configuration Space **PCI Express Device Control** register *Maximum Read Request Size* field must be modified from its default 512 to 4 KB, consistent with the Prefetch Size.

e. Click the PCR button on the **PLXMon** program window, to open the **PCI Register** dialog box (Figure 3). The values shown are in hex.

f. Change **PCIe Dev Status/Control (68)** to read *nnnn5nnn*, where *n* is don't care, to select a Maximum Read Request Size of 4 KB. Click the **Refresh** button to confirm and return to the program window.

Note: *The PCI Express Configuration Space* ***PCI Express Device-Specific Control*** *register (offset 48h,* ***DEVSPECCTL****) is not accessible in version 4.4 of the* ***PLXMon*** *utility; however, it can be accessed using the Command Line interface. The* ***DEVSPECCTL*** *register is used to enable Blind Prefetch, among other functions. Typing "pci 48" displays the register contents. Typing "pci 48 <value>" is used to Write.*



PCI Registers

Device/Vendor ID	(00)	811110B5
Command/Status	(04)	00100117
Class Code/Revision	(08)	06040021
Hdr Type / Cache Line	(0C)	00010010
Base Address 0	(10)	00000000
Base Address 1	(14)	00000000
Prim/Sec Bus Num	(18)	20070706
I/O Limit/Base	(1C)	2200C0C0
Memory Limit/Base	(20)	FE60FE60
PF Memory Limit/Base	(24)	DFB0DFB0
PF Base (high 32-bits)	(28)	00000000
PF Limit (high 32-bits)	(2C)	00000000
I/O L/B (high 16-bits)	(30)	00000000
Capabilities Pointer	(34)	00000040
Expansion ROM Base	(38)	00000000
Bridge Ctrl / Int Pin/Line	(3C)	00070111
Main Ctrl Reg Index	(84)	00000000
Main Ctrl Reg Index	(88)	00000033

PCI Extended Capabilities

Power Mgmt Capability	(40)	5A025001
Power Mgmt Control/Stat	(44)	00000000
MSI Capability	(50)	00806005
MSI Msg Addr (low 32-bits)	(54)	00000000
MSI Msg Addr (high 32-bits	(58)	00000000
MSI Message Data	(5C)	00000000
PCI Express Capability	(60)	00710010
PCIe Device Capabilities	(64)	00280000
PCIe Dev Status/Control	(68)	00005000
PCIe Link Capabilities	(6C)	00024C11
PCIe Link Status/Control	(70)	00110000
PCIe Slot Capabilities	(74)	00000C80
PCIe Slot Status/Control	(78)	00400000
PCIe Root Control	(7C)	00000000
PCIe Root Status	(80)	00000000

PCI Express Extended Capabilities

Pwr Budgeting Capability	(100)	00010004
Pwr Budget Data Select	(104)	00000000
Power Budget Data	(108)	00000000
Power Budget Capability	(10C)	00000000
Serial Number Capability	---	
Serial No. (Lower 32-bits)	---	
Serial No. (Upper 32-bits)	---	

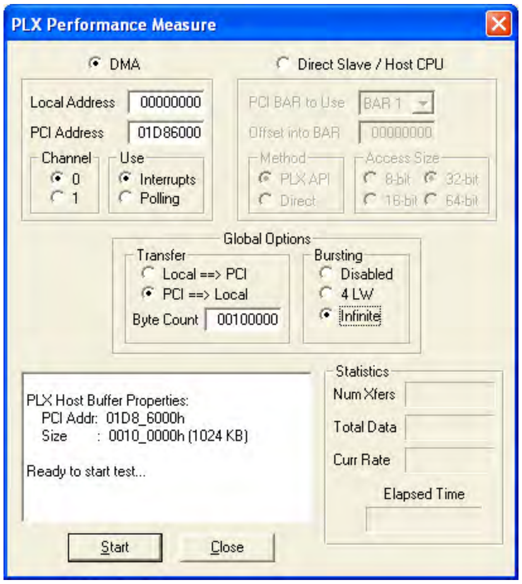
Figure 3. PEX 8311 PCI Express Configuration Space Configuration Registers Dialog Box

Running PEX 8311 Performance Measure GUI

To run the PEX 8311 Performance Measure GUI within PLXMon

a. From the **Command** menu, select **Select Device**, then select **PEX 8311**.

b. Click the (**Performance Measure**) icon in the toolbar, to open the **PLX Performance Measure** GUI dialog box (Figure 4).



PLX Performance Measure

● DMA   ○ Direct Slave / Host CPU

Local Address: 00000000
PCI Address: 01D86000

Channel: ● 0 ○ 1
Use: ● Interrupts ○ Polling

PCI BAR to Use: BAR 1
Offset into BAR: 00000000
Method: ● PLX API ○ Direct
Access Size: ○ 8-bit ● 32-bit ○ 16-bit ○ 64-bit

Global Options
Transfer: ○ Local ==> PCI ● PCI ==> Local
Byte Count: 00100000
Bursting: ○ Disabled ○ 4 LW ● Infinite

```
PLX Host Buffer Properties:
   PCI Addr:  01D8_6000h
   Size     :  0010_0000h (1024 KB)

Ready to start test...
```

Statistics: Num Xfers, Total Data, Curr Rate, Elapsed Time

Figure 4. PLX Performance Measure Dialog Box – DMA Test Settings

c. Ensure the **DMA** radio button is selected.

d. In the **Global Options**, **Transfer** area:

   1. Select the **PCI ==> Local** radio button.

   2. Type *00100000* (hex) in the **Byte Count** text box, to specify a 1-MB Transfer Size.

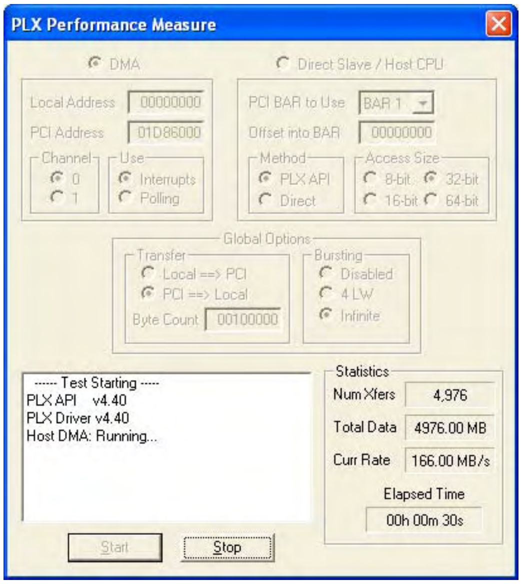
      Caution: Take care not to exceed the DMA Memory Buffer size, shown in the lower-left window. Exceeding this buffer can cause computer failure. The Performance Measure GUI dialog box does not check for memory bound violations.

e. In the **Global Options**, **Bursting** area, select the **Infinite** radio button, to enable continuous bursting on the Local Bus.

f. Click the **Start** button, to run the test.

Results should look similar to Figure 5.

Note: *Results may vary slightly from system to system. Figure 5 displays factors in GUI overhead. Actual performance, therefore, is slightly higher in actual applications.*



Local Address 00000000
PCI Address 01D86000
Channel: ● 0 ○ 1
Use: ● Interrupts ○ Polling
PCI BAR to Use BAR 1
Offset into BAR 00000000
Method: ● PLX API ○ Direct
Access Size: ○ 8-bit ● 32-bit ○ 16-bit ○ 64-bit
Transfer: ○ Local ==> PCI ● PCI ==> Local
Byte Count 00100000
Bursting: ○ Disabled ○ 4 LW ● Infinite

```
------ Test Starting -----
PLX API    v4.40
PLX Driver v4.40
Host DMA: Running...
```

Num Xfers 4,976
Total Data 4976.00 MB
Curr Rate 166.00 MB/s
Elapsed Time 00h 00m 30s

Figure 5. Performance Measure Results Dialog Box

g. Click the **Stop** button when finished, to return to the **PLXMon** program window.

4. Conclusion

The PEX 8311RDK board and PLX PCI SDK is a useful tool for designers to use in estimating system performance for given operating parameters. Using the **PLXMon** utility and its **Performance Measure** GUI, it is possible for designers to select various options in the PEX 8311 registers, then immediately visualize the effects on DMA and Direct Slave transfer performance.

The PEX 8311AA maximum Write speed was measured to be 177 MB/s. The maximum Read speed was measured to be 167 MB/s. The PCI clock used was 66 MHz, with a 128-byte TLP.