PCIe Host Driver Software for CertusPro-NX AXI-MM and AXI-S DMA User Guide

Technical Note

FPGA-TN-02386-1.2

April 2025

Disclaimers

Lattice makes no warranty, representation, or guarantee regarding the accuracy of information contained in this document or the suitability of its products for any particular purpose. All information herein is provided AS IS, with all faults, and all associated risk is the responsibility entirely of the Buyer. The information provided herein is for informational purposes only and may contain technical inaccuracies or omissions, and may be otherwise rendered inaccurate for many reasons, and Lattice assumes no obligation to update or otherwise correct or revise this information. Products sold by Lattice have been subject to limited testing and it is the Buyer's responsibility to independently determine the suitability of any products and to test and verify the same. LATTICE PRODUCTS AND SERVICES ARE NOT DESIGNED, MANUFACTURED, OR TESTED FOR USE IN LIFE OR SAFETY CRITICAL SYSTEMS, HAZARDOUS ENVIRONMENTS, OR ANY OTHER ENVIRONMENTS REQUIRING FAIL-SAFE PERFORMANCE, INCLUDING ANY APPLICATION IN WHICH THE FAILURE OF THE PRODUCT OR SERVICE COULD LEAD TO DEATH, PERSONAL INJURY, SEVERE PROPERTY DAMAGE OR ENVIRONMENTAL HARM (COLLECTIVELY, "HIGH-RISK USES"). FURTHER, BUYER MUST TAKE PRUDENT STEPS TO PROTECT AGAINST PRODUCT AND SERVICE FAILURES, INCLUDING PROVIDING APPROPRIATE REDUNDANCIES, FAIL-SAFE FEATURES, AND/OR SHUT-DOWN MECHANISMS. LATTICE EXPRESSLY DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY OF FITNESS OF THE PRODUCTS OR SERVICES FOR HIGH-RISK USES. The information provided in this document is proprietary to Lattice Semiconductor, and Lattice reserves the right to make any changes to the information in this document or to any products at any time without notice.

Inclusive Language

This document was created consistent with Lattice Semiconductor’s inclusive language policy. In some cases, the language in underlying tools and other items may not yet have been updated. Please refer to Lattice’s inclusive language FAQ 6878 for a cross reference of terms. Note in some cases such as register names and state names it has been necessary to continue to utilize older terminology for compatibility.

© 2025 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

Contents

© 2025 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

Figures

Tables

© 2025 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

Abbreviations in This Document

A list of abbreviations used in this document.

Abbreviation	Definition
API	Application Programming Interface
AXI-MM	Advanced eXtensible Interface Memory-Mapped
AXI-S	Advanced eXtensible Interface Stream
BAR	Base Address Register
CPNX	CertusPro-NX
CPU	Central Processing Unit
DMA	Direct Memory Access
FPGA	Field Programmable Gate Array
OS	Operating System
PCIe	Peripheral Component Interconnect Express
RAM	Random Access Memory
RO	Read-only
RW	Read-write
SGDMA	Scatter-Gather Direct Memory Access
SPI	Serial Peripheral Interface
SRAM	Static Random Access Memory

© 2025 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

1. Introduction

PCI Express® is a high performance, fully scalable, and well-defined standard for a wide variety of computing and communications platforms. Refer to the PCIe X4 IP Core User Guide (FPGA-IPUG-02126) for more details about the IP core.

This guide describes how to use the PCIe host driver software with the CertusPro™-NX AXI-MM DMA module and AXI-S DMA module.

The software driver is developed using Jungo WinDriver software toolkit. You can develop your own driver or use the Jungo WinDriver for driver development. To use Jungo WinDriver, contact Jungo to obtain a valid paid annual subscription. For more information, contact Jungo.

1.1. Purpose

This document is intended to act as a reference guide for developers by providing details of the C language driver APIs and function call flows.

1.2. Audience

The intended audience for this document includes embedded system designers and embedded software developers using Lattice CertusPro-NX devices. The technical guide assumes readers have expertise in embedded systems and FPGA technologies.

1.3. Driver Version

PCIe Host AXI-MM DMA driver version 24.02.01.

PCIe Host AXI-S DMA driver version 24.02.00.

1.4. Driver and IP Compatibility

Table 1.1. Driver and IP Compatibility

Driver	Driver Version	IP Version
AXI-MM DMA driver	24.02.01	3.3.0
AXI-S DMA driver	24.02.00	3.4.0

Refer to the PCIe x4 IP Release Notes (FPGA-RN-02059) for more information on the driver and IP versions.

Table 1.2. Quick Facts of Driver Tested Environment

Driver Tested on Hardware Devices	PC Environment
CertusPro-NX PCIe Bridge Board	OS: Windows 10, Windows 11, Linux
	Supported architecture: x86_64
	CPU: Intel, AMD

© 2025 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

2. Software Setup

This section describes the steps to install the software driver on the host machine. The installation procedure is similar for both the AXI-MM DMA driver and the AXI-S DMA driver.

The following installation procedures use the AXI-MM DMA driver as an example.

2.1. Driver Installation on a Windows Machine

1. If you are installing the driver for the first time, skip step 2 and go to step 3.
2. If the driver is already installed previously, run *Uninstall.exe* in the installation folder: *C:\Program Files\cpnxdma_24.02.01* before you install the driver.
3. Double click on *cpnxdma-24.02.01-win64.exe* to install the driver. Select **Next** to continue with the installation.

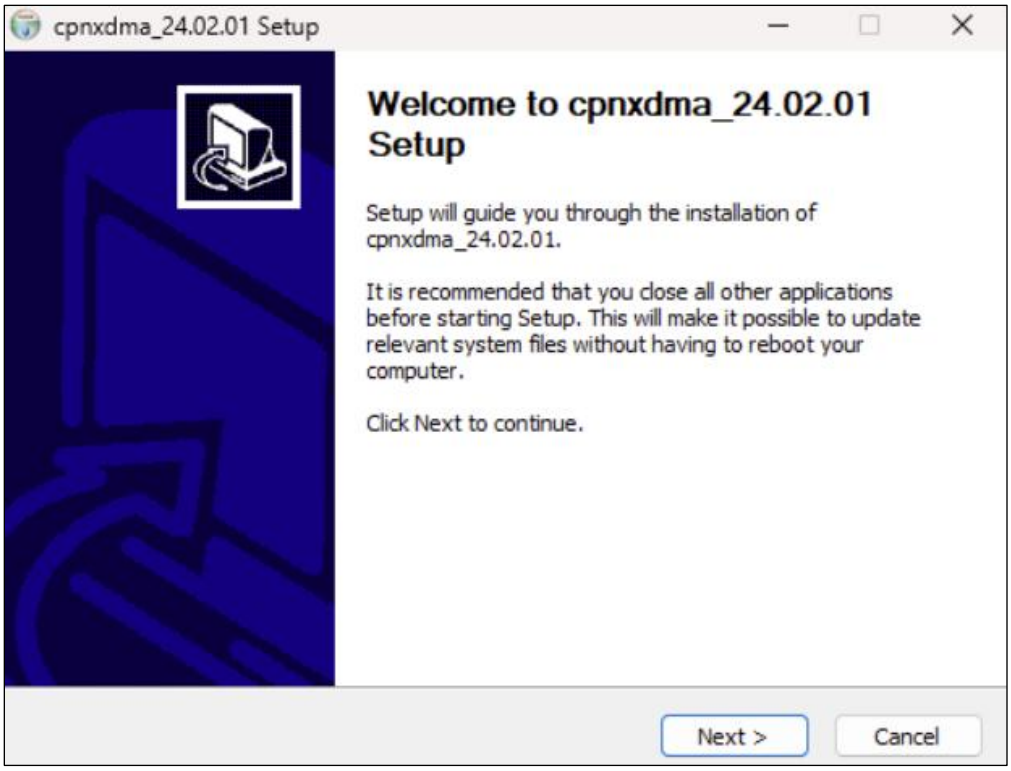

Figure 2.1. Setup

© 2025 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

4. Click **I Agree** to continue with the installation.

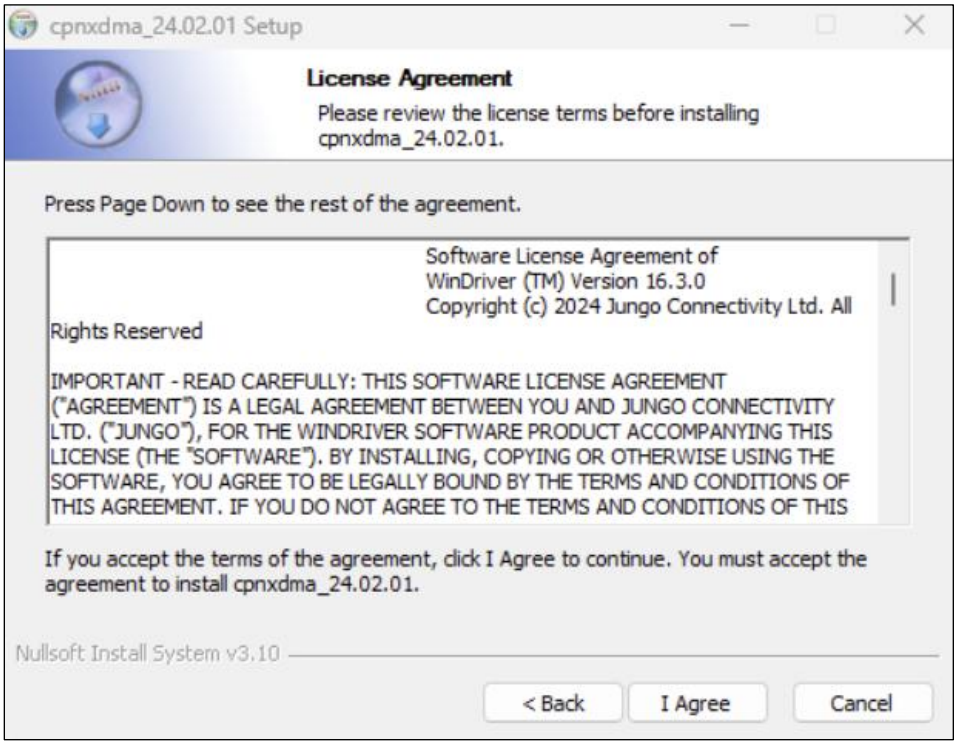

Figure 2.2. Setup License Agreement

5. Use the default location in the **Destination Folder** as the installation location and click **Next** to continue.

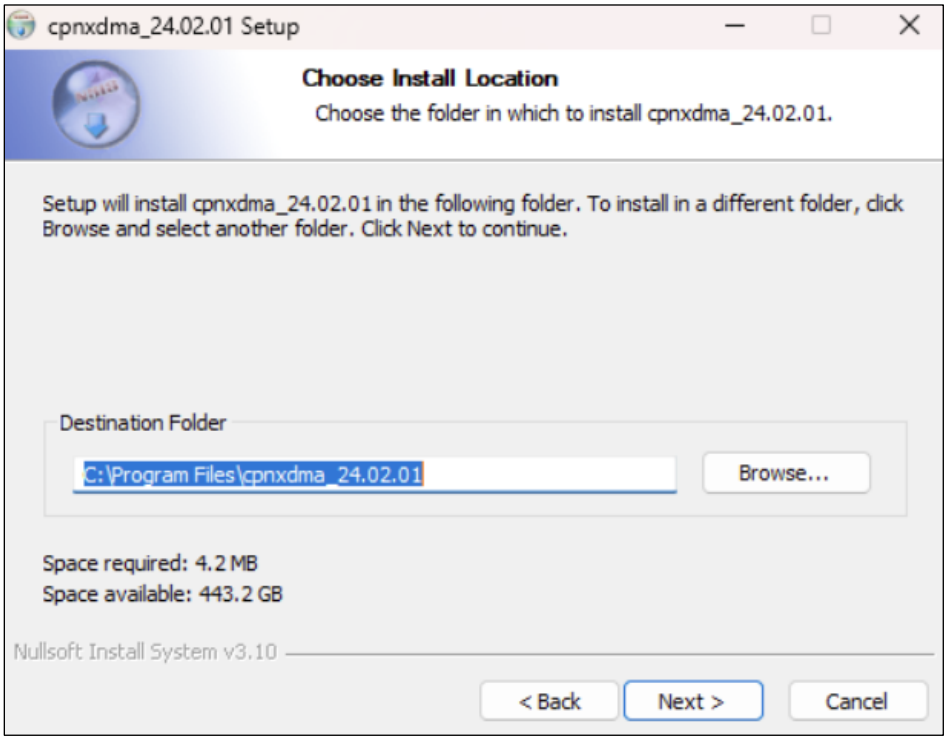

Figure 2.3. Setup Install Location

© 2025 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

6. Click **Next** until you see the window as shown in Figure 2.4. Click **Install**.

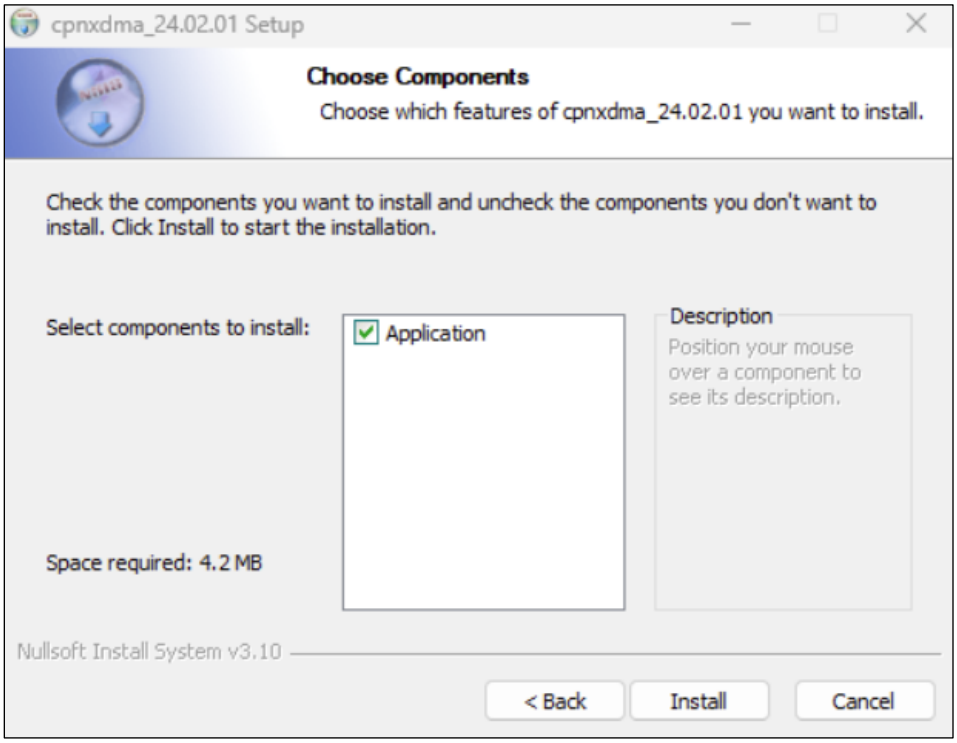

Figure 2.4. Setup Install

© 2025 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

7. When installation is complete, open **Device Manager**. Two new devices are listed under **Jungo Connectivity**:
   - cpnxdma Driver
   - PCIe (Gen3 – Lattice-Device: 0x9c25 (Requiring driver: cpnxdma)

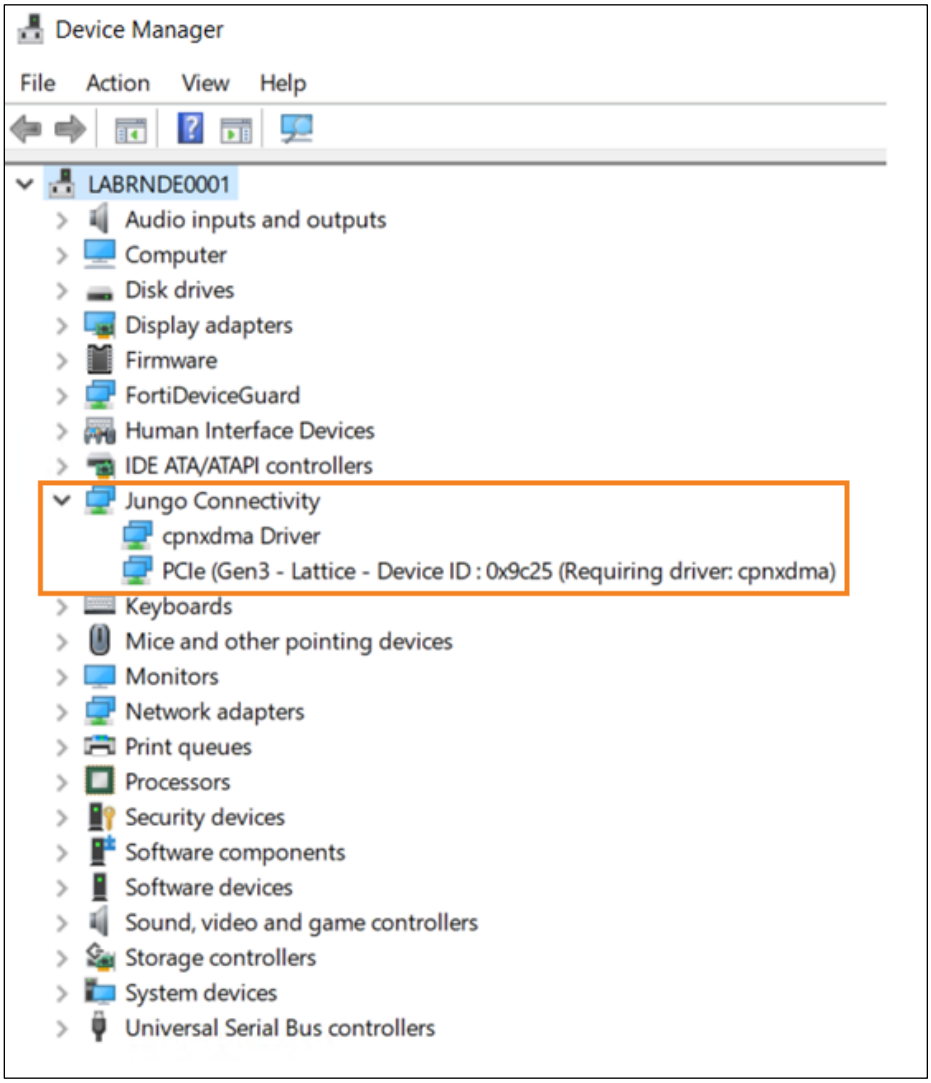

Figure 2.5. Device Manager

8. A new folder is created in *C:\Program Files\cpnxdma_24.02.01*.
9. A shortcut: *cpnxdma_24.02.01* is added to your desktop.
10. You can run the user app by running the desktop shortcut or by running: *C:\Program Files\cpnxdma_24.02.01\bin\cpnxdma.exe*.

Note: This driver is built with a demo license and is limited to 30 days. After 30 days, you need to replace the license with a valid license and rebuild the driver. For more information, refer to the Jungo WinDriver Official Documentation. To continue using WinDriver drivers, you can get a valid paid annual subscription from Jungo. For more information, contact Jungo.

© 2025 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

2.2. Driver Installation on a Linux Machine

1. Before installing the driver, check if you have gcc installed by typing the following command in your terminal:
   ```
   gcc –version
   ```
2. If you see a message indicating that the command is not found, install *gcc*:
   ```
   sudo apt update
   sudo apt install gcc
   gcc –version
   ```
3. Go to the directory where file *cpnxdma-24.02.01-Linux.sh* is located. Give executable permission to the *.sh* file:
   ```
   sudo chmod +x cpnxdma-24.02.01-Linux.sh
   ```
4. Then, run the command below to build the driver and user application and install the driver.
   ```
   sudo ./cpnxdma-24.02.01-Linux.sh
   ```
5. Enter *Y* when prompted.
6. Two new modules are installed. Run the command below to verify that *cpnxdma* is installed:
   ```
   sudo lsmod | grep cpnxdma
   ```

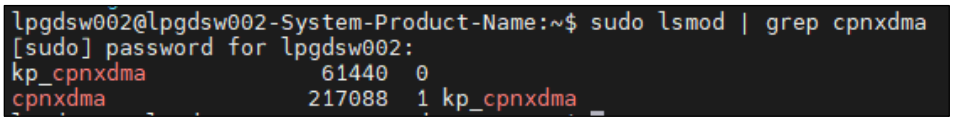

Figure 2.6. lsmod

7. A new folder is created in your current directory *cpnxdma-24.02.01-Linux*.
8. Give permission to all the files in the directory:
   ```
   sudo chmod +x -R *
   ```
9. Run the user app:
   ```
   sudo ./cpnxdma-24.02.01-Linux/bin/cpnxdma
   ```

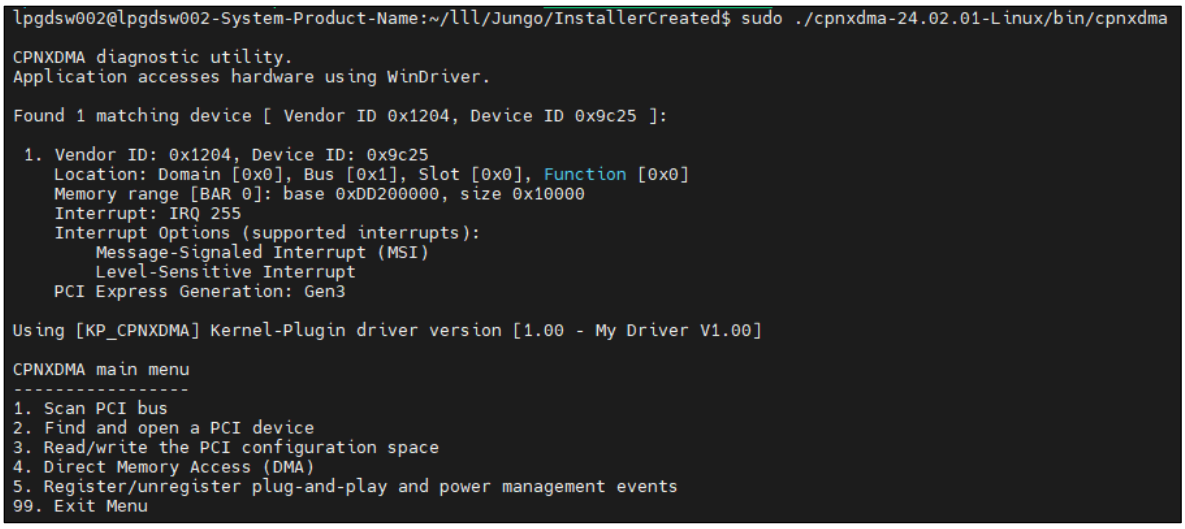

Figure 2.7. Run cpnxdma

Note: This driver is built with a demo license and is limited to one hour. You need to reinstall the driver after one hour. To continue using WinDriver drivers without having to reinstall every hour, contact Jungo to obtain a valid paid annual subscription. Contact Jungo for more information.

© 2025 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

3. Application Overview

The PCIe software driver can run on both Windows and Linux operating systems. To run the software driver, you need to program the FPGA the example design to SPI flash or SRAM on the board.

The PCIe software driver is developed using the Jungo WinDriver software toolkit. It comes with a console-based user application that provides the following functionalities:

- Scan PCI bus
- Find and open a PCI device
- Read/Write the PCI configuration space
- Direct Memory Access (DMA)

3.1. DMA Functionalities

Below are the DMA functionalities provided by the user application.

```
CPNXDMA main menu
-----------------
1. Scan PCI bus
2. Find and open a PCI device
3. Read/write the PCI configuration space
4. Direct Memory Access (DMA)
5. Register/unregister plug-and-play and power management events
99. Exit Menu

Enter option: 4

CPNXDMA DMA menu
----------------
1. Perform DMA transfer
2. Perform DMA transfer to device, perform DMA transfer from device and compare the DMA buffers
3. Measure DMA performance
99. Exit Menu

Enter option:
```

Figure 3.1. CPNXDMA DMA Menu

```
CPNXAXISDMA main menu
---------------------
1. Scan PCI bus
2. Find and open a PCI device
3. Read/write memory and I/O addresses on the device
4. Read/write the PCI configuration space
5. Direct Memory Access (DMA)
6. Register/unregister plug-and-play and power management events
99. Exit Menu

Enter option: 5

CPNXAXISDMA DMA menu
--------------------
1. Perform DMA transfer
3. Perform DMA transfer from device and compare the DMA buffer
4. Measure DMA performance
99. Exit Menu

Enter option:
```

Figure 3.2 CPNXAXISDMA DMA Menu

© 2025 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

3.1.1. DMA Transfer

3.1.1.1. AXI-MM DMA Transfer

When selecting **Perform DMA transfer**, you are prompted to select the DMA direction.

The **From device** DMA direction indicates from the FPGA to the host device, while the **To device** DMA direction indicates from the host device to the FPGA.

```
CPNXDMA DMA menu
----------------
1. Perform DMA transfer
2. Perform DMA transfer to device, perform DMA transfer from device and compare the DMA buffers
3. Measure DMA performance
99. Exit Menu

Enter option: 1

Select DMA direction:
---------------------
1. From device
2. To device
99. Cancel
Enter option: 1

Enter number of packets to transfer (32 bit packets) (max value: 32768) or 'x' to cancel: 10

NOTICE: The device memory address should be 8 DWORD aligned, otherwise it is rounded down to the nearest 8 DWORD aligned address

Enter device memory address for transfer (max value: 0x1FFD8) or 'x' to cancel: 0x0

Running DMA using Kernel-PlugIn driver [KP_CPNXDMA]

###
Level Sensitive Interrupt #1 received
###

Buffer:

0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000


DMA transfer completed successfully
DMA interrupts disabled
Closed DMA handle
```

Figure 3.3. CPNXDMA Transfer Direction

The AXI-MM example design is built with an FPGA internal RAM of 128 kilobytes (131072 bytes). Therefore, when prompted for the device memory address, enter a range from 0 to 0x1FFFF (131071 in decimal). However, consider the size of data to transfer and ensure that the device memory address + data size does not exceed 0x1FFFF.

3.1.1.2. AXI-S DMA Transfer

When selecting **Perform DMA transfer**, only the **From device** DMA direction is supported.

The AXI-S example design does not include FPGA internal RAM. Therefore, you are not prompted to enter a device memory address when performing the DMA transfer.

© 2025 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

```
CPNXAXISDMA DMA menu
--------------------
1. Perform DMA transfer
3. Perform DMA transfer from device and compare the DMA buffer
4. Measure DMA performance
99. Exit Menu

Enter option: 1

Select DMA direction:
---------------------
1. From device
99. Cancel
Enter option: 1

Enter number of packets to transfer (4-byte packets) (max value: 26214400) or 'x' to cancel: 10
Running DMA using Kernel-PlugIn driver [KP_CPNXAXISDMA]

###
Message Signalled Interrupt #1 received
MSI data 0x0
###

Buffer:

00000000 e3e2e1e0 e7e6e5e4 ebeae9e8 efeeedec f3f2f1f0 f7f6f5f4 fbfaf9f8 fffefdfc
00000020 03020100 07060504

DMA transfer completed successfully
DMA interrupts disabled
Closed DMA handle
```

Figure 3.4 CPNXAXISDMA Transfer Direction

© 2025 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

3.1.2. DMA Transfer and Compare Data

3.1.2.1. AXI-MM DMA Transfer and Compare Data

When selecting **Perform DMA transfer to device, perform DMA transfer from device and compare the DMA buffers**, the driver application will perform DMA transfer from host to device, followed by a DMA transfer from device to host, then compare the values of the data read to data written. The status of the DMA buffer compare will be reported on the console.

```
CPNXDMA DMA menu
----------------
1. Perform DMA transfer
2. Perform DMA transfer to device, perform DMA transfer from device and compare the DMA buffers
3. Measure DMA performance
99. Exit Menu

Enter option: 2

Enter DMA data pattern as 32 bit packet (to cancel press 'x'): 0x12345678

Enter number of packets to transfer (32 bit packets) (max value: 32768) or 'x' to cancel: 10

NOTICE: The device memory address should be 8 DWORD aligned, otherwise it is rounded down to the nearest 8 DWORD aligned address

Enter device memory address for transfer (max value: 0x1FFD8) or 'x' to cancel: 0x0

Running DMA using Kernel-PlugIn driver [KP_CPNXDMA]

###
Level Sensitive Interrupt #1 received
###


DMA transfer completed successfully
Running DMA using Kernel-PlugIn driver [KP_CPNXDMA]

###
Level Sensitive Interrupt #2 received
###

Buffer:

0x12345678 0x12345678 0x12345678 0x12345678 0x12345678 0x12345678 0x12345678 0x12345678 0x12345678


DMA transfer completed successfully
Write buffer and read buffer are identical
DMA interrupts disabled
```

Figure 3.5. CPNXDMA Data Compare

3.1.2.2. AXI-S DMA Transfer and Compare Data

The AXI-S DMA example design configures the FPGA to transfer a running incremental data when the host initiates a DMA transfer. The data returned by the FPGA increments byte by byte, for example, 0x00, 0x01, 0x02, 0x03, up to 0xFF, then rollover to 0x00 and continues the sequence.

When selecting **Perform DMA transfer from device and compare the DMA buffer**, the driver application executes a DMA transfer from the device to host. The application then compares and verifies that the data follows an incremental pattern, ranging from 0x00 to 0xFF, byte by byte. The status of the data verification is reported on the console.

© 2025 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

```
CPNXAXISDMA DMA menu
--------------------
1. Perform DMA transfer
3. Perform DMA transfer from device and compare the DMA buffer
4. Measure DMA performance
99. Exit Menu

Enter option: 3

NOTICE: The DMA transfer size should be DWORD aligned, otherwise it is rounded down to the nearest DWORD aligned address

Enter number of packets to transfer (1-byte packets) (max value: 104857600) or 'x' to cancel: 100
Running DMA using Kernel-PlugIn driver [KP_CPNXAXISDMA]

###
Message Signalled Interrupt #1 received
MSI data 0x0
###

Buffer:

00000000 20 21 22 23 24 25 26 27 28 29 2a 2b 2c 2d 2e 2f 30 31 32 33 34 35 36 37 38 39 3a 3b 3c 3d 3e 3f
00000020 40 41 42 43 44 45 46 47 48 49 4a 4b 4c 4d 4e 4f 50 51 52 53 54 55 56 57 58 59 5a 5b 5c 5d 5e 5f
00000040 60 61 62 63 64 65 66 67 68 69 6a 6b 6c 6d 6e 6f 70 71 72 73 74 75 76 77 78 79 7a 7b 7c 7d 7e 7f
00000060 80 81 82 83

DMA transfer completed successfully
DMA data verification successful
DMA interrupts disabled
```

Figure 3.6 CPNXAXISDMA Data Compare

© 2025 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

3.1.3. DMA Performance Measure

When selecting **Measure DMA performance**, ensure the PCIe link status is showing *Link Speed* of 8GT/s and *Link Width* of x4 to get the maximum throughput. You can check this by using this application driver to read the configuration space.

Select **Read/Write PCI configuration space**, followed by **Read all configuration registers defined**. Then, look for register *LINK_STS*.

```
9. LNK_STS              0x1043      Link Status          Current Link Speed: 8.0 GT/s
                                                         Negotiated Link Width: x4
                                                         Link Training: False
                                                         Slot Clock Configuration: True
                                                         Data Link Layer Link Active: False
                                                         Link Bandwidth Management Status: False
                                                         Link Autonomous Bandwidth Status: False
```

Figure 3.7. PCIe Link Status Register

If the link is not 8.0 GT/s, verify that the FPGA card is connected to a suitable slot that supports PCIe Gen3 and that the slot is properly configured to support Gen3. You may need to configure the PCIe speed in BIOS.

3.1.3.1. AXI-MM DMA Performance Measure

DMA performance measurement is available for single-direction DMA transfer (host to device or device to host) and bi-directional simultaneous DMA transfer.

For the single-direction DMA transfers, the maximum buffer size available is 128 kB.

```
CPNXDMA DMA menu
----------------
1. Perform DMA transfer
2. Perform DMA transfer to device, perform DMA transfer from device and compare the DMA buffers
3. Measure DMA performance
99. Exit Menu

Enter option: 3

DMA performance
---------------
1. DMA host-to-device performance
2. DMA device-to-host performance
3. DMA host-to-device and device-to-host performance running simultaneously
99. Exit Menu

Enter option: 2

Enter single transfer buffer size in KBs (max value: 128) or 'x' to cancel: 128

Enter number of times to repeat DMA: (max value: 4294967295) or 'x' to cancel: 1000


Running DMA device-to-host performance test 1000 times
Running DMA performance test from Kernel-PlugIn driver [KP_CPNXDMA]
DMA device-to-host performance test: Transferred 131072000 bytes, elapsed time 39935200[ns], rate 3130.07 [MB/sec]
```

Figure 3.8. CPNXDMA Measure DMA Performance for Single-Direction Transfer

For bi-directional simultaneous DMA transfer, the maximum buffer size available is 128 ÷ 2 = 64 kB. The internal FPGA RAM is divided into two halves: one half for host-to-device DMA transfers and the other half for device-to-host DMA transfers.

© 2025 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

```
DMA performance
---------------
1. DMA host-to-device performance
2. DMA device-to-host performance
3. DMA host-to-device and device-to-host performance running simultaneously
99. Exit Menu

Enter option: 3

Enter single transfer buffer size in KBs (max value: 64) or 'x' to cancel: 64

Enter number of times to repeat DMA: (max value: 4294967295) or 'x' to cancel: 1000


Running DMA bi-directional performance test 1000 times
Running DMA performance test from Kernel-PlugIn driver [KP_CPNXDMA]
Running DMA performance test from Kernel-PlugIn driver [KP_CPNXDMA]
```

Figure 3.9. CPNXDMA Measure DMA Performance for Bi-Directional Simultaneous Transfer

3.1.3.2. AXI-S DMA Performance Measure

DMA performance measurement is available only with single-direction DMA transfer and with device to host DMA transfer. Bi-directional simultaneous DMA transfer is not supported by the AXI-S DMA driver. The maximum transfer size is 100 MB at a time.

```
CPNXAXISDMA DMA menu
--------------------
1. Perform DMA transfer
3. Perform DMA transfer from device and compare the DMA buffer
4. Measure DMA performance
99. Exit Menu

Enter option: 4

DMA performance
---------------
1. DMA device-to-host performance
99. Exit Menu

Enter option: 1

Enter single transfer buffer size in KBs (max value: 102400) or 'x' to cancel: 102400

Enter number of times to repeat DMA: (max value: 100) or 'x' to cancel: 100


Running DMA device-to-host performance test 100 times
Running DMA performance test from Kernel-PlugIn driver [KP_CPNXAXISDMA]
DMA device-to-host performance test: Transferred 10485760000 bytes, elapsed time 2807942900[ns], rate 3561.33 [MB/sec]
```

Figure 3.10 CPNXAXISDMA Measure DMA Performance

© 2025 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

3.2. Read/Write Memory and I/O Addresses

This section is only applicable to the AXI-S DMA driver. The AXI-S DMA example design has the DMA bypass mode enabled, allowing you to perform memory read/write on the device. When selecting **Read/write memory and I/O addresses on the device**, the default active address space is BAR 0. BAR 0 is mapped to the DMA internal registers for DMA configuration and status. Therefore BAR 0 memory is read-only (RO) to prevent any interference with the internal registers.

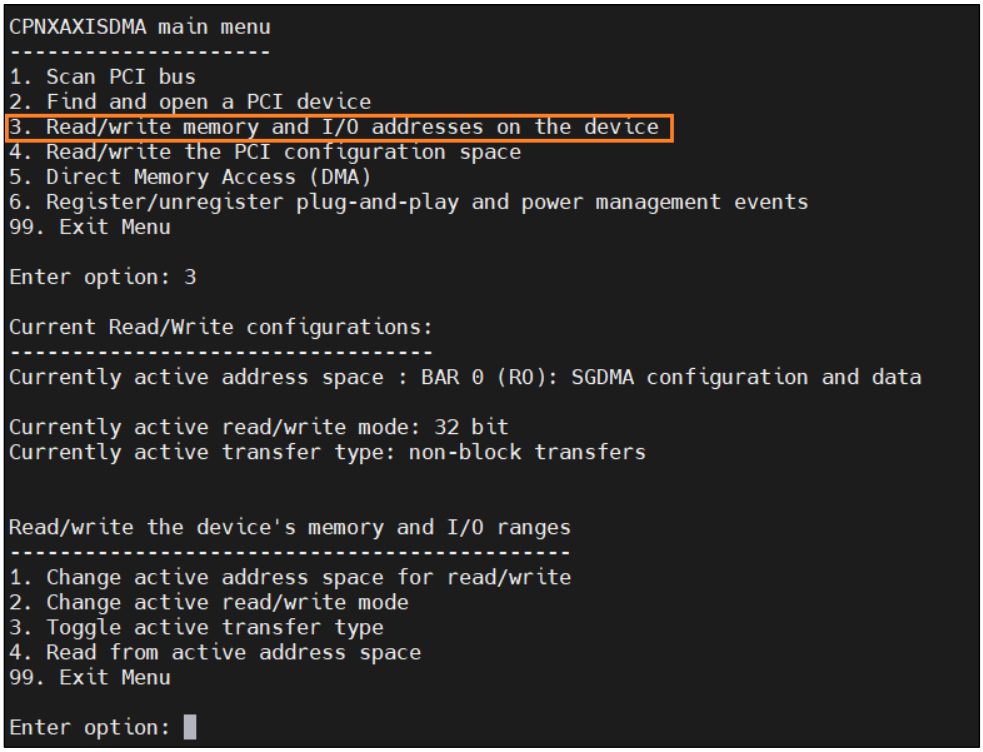

Figure 3.11 CPNXAXISDMA Read/Write Memory

The AXI-S DMA example design has 2 BARs enabled:

- BAR 0 for DMA registers
- BAR 1 for memory or I/O read/write operations.

BAR 0 memory is read-only (RO), while BAR 1 memory is read-write (RW).

To change the active address space to BAR1, select **Change active address space for read/write.** When BAR 1 is selected, both read and write operations can be performed on the memory.

© 2025 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

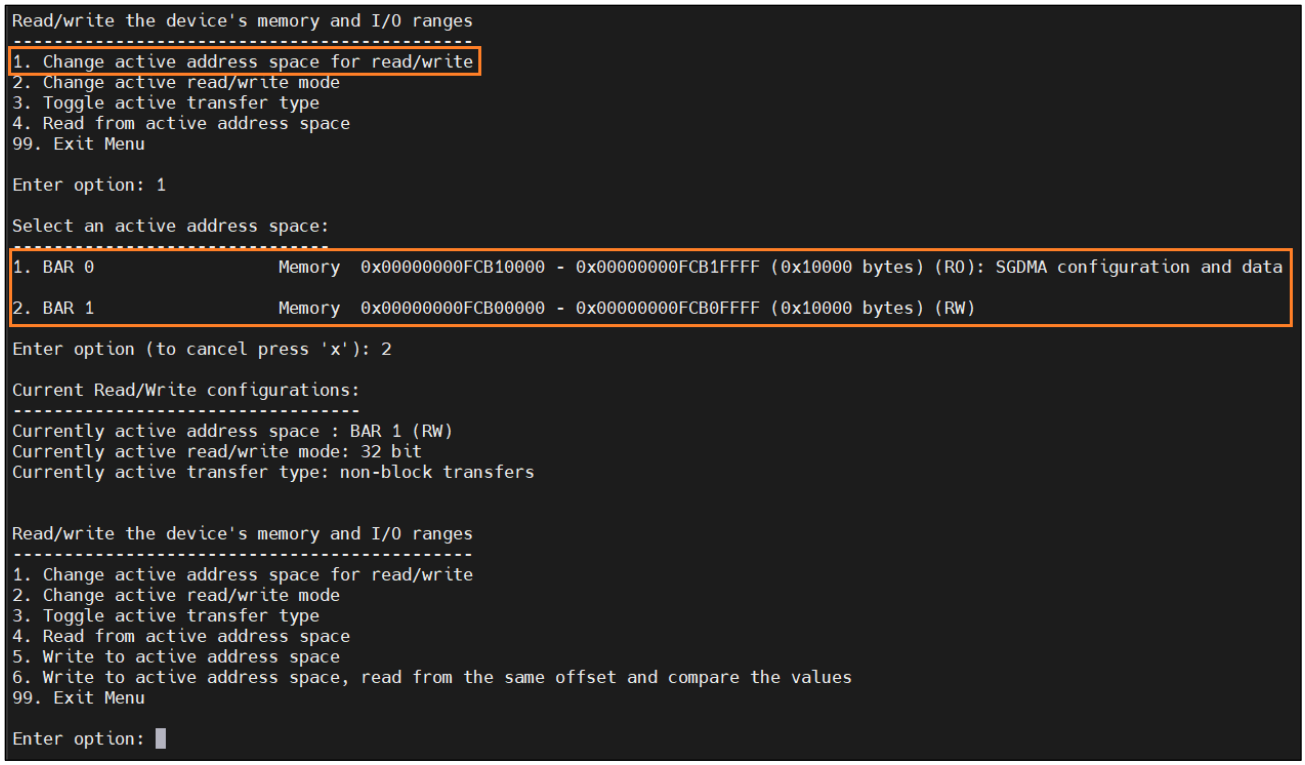

Figure 3.12 CPNXAXISDMA Change Active Address Space for Read/Write

3.2.1. Read/Write Mode

The driver application supports read/write modes of 8 bits, 16 bits, 32 bits, and 64 bits. However, the CertusPro-NX FPGA only supports up to 32 bits. You can change the read/write mode by selecting **Change active read/write mode.**

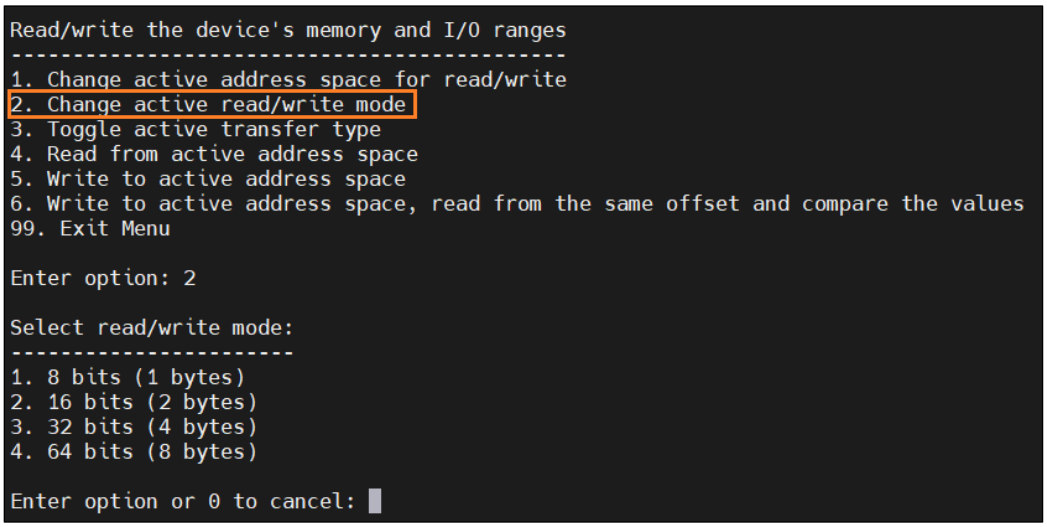

Figure 3.13 CPNXAXISDMA Change Active Read/Write Mode

3.2.2. Transfer Type: Non-block or Block Transfers

You can toggle between non-block transfers and block transfers by selecting **Toggle active transfer type**.

© 2025 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

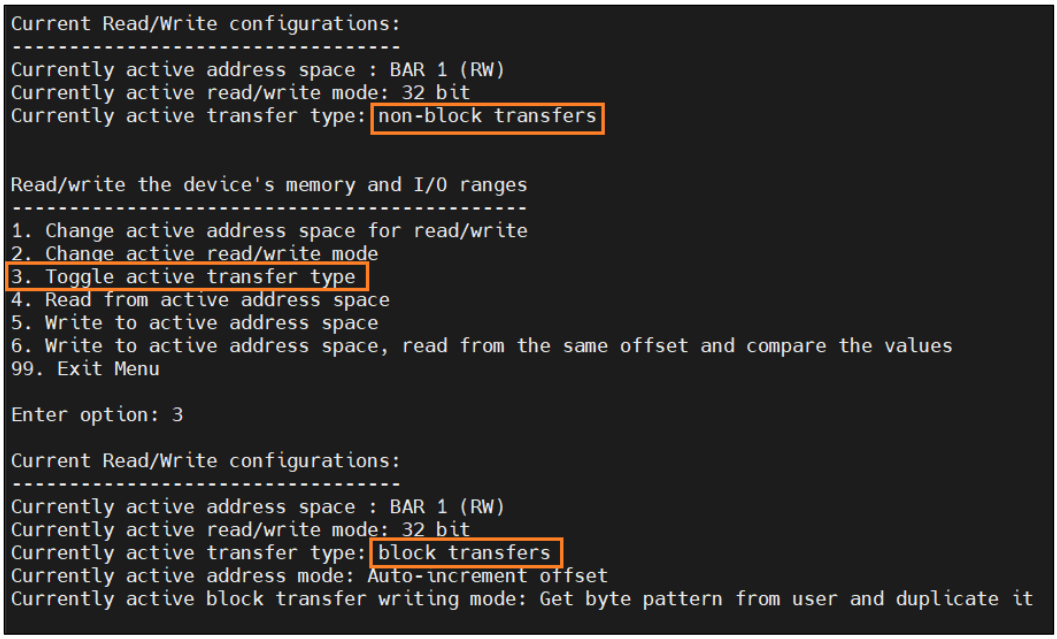

Figure 3.14 CPNXAXISDMA Non-block or Block Transfer

3.2.2.1. Non-block Transfer

Non-block transfer refers to a read/write operation to a single unit of memory depending on the read/write mode:

- If 8-bit mode is selected, non-block transfer reads/writes 1 byte of data.
- If 16-bit mode is selected, non-block transfer reads/writes 2 bytes of data.
- If 32-bit mode is selected, non-block transfer reads/writes 4 bytes of data.
- If 64-bit mode is selected, non-block transfer reads/writes 8 bytes of data.

3.2.2.2. Block Transfer

Block transfer allows you to read/write data to/from a block of memory.

3.2.3. Write, Read, and Compare Values

For both block and non-block transfers, there is a function that writes your entered data, then reads it back, and compares the values. Any data mismatches are reported on the console.

© 2025 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

```
Current Read/Write configurations:
----------------------------------
Currently active address space : BAR 1 (RW)
Currently active read/write mode: 32 bit
Currently active transfer type: non-block transfers


Read/write the device's memory and I/O ranges
---------------------------------------------
1. Change active address space for read/write
2. Change active read/write mode
3. Toggle active transfer type
4. Read from active address space
5. Write to active address space
6. Write to active address space, read from the same offset and compare the values
99. Exit Menu

Enter option: 6
Notice: Your device must support both read and write of size [0x4] from the selected offset for this option to work
offset (to cancel press 'x'): 0x0
Enter data to write (max value: 0xFFFFFFFF) or 'x' to cancel: 0x12345678
Wrote 0x12345678 to offset 0x0 in BAR 1
Read 0x12345678 from offset 0x0 in BAR 1
Write buffer and read buffer are identical
```

Figure 3.15 Non-block Transfer – Write, Read, and Compare Values

```
Current Read/Write configurations:
----------------------------------
Currently active address space : BAR 1 (RW)
Currently active read/write mode: 32 bit
Currently active transfer type: block transfers
Currently active address mode: Auto-increment offset
Currently active block transfer writing mode: Get start value from user and increment it


Read/write the device's memory and I/O ranges
---------------------------------------------
1. Change active address space for read/write
2. Change active read/write mode
3. Toggle active transfer type
4. Read from active address space
5. Write to active address space
6. Write to active address space, read from the same offset and compare the values
7. Toggle active block transfer address increment mode
8. Change active block transfer write mode
99. Exit Menu

Enter option: 6
Notice: Your device must support both read and write of size [0x4] from the selected offset for this option to work
offset (to cancel press 'x'): 0x0
bytes (to cancel press 'x'): 0x100
Enter starting value (max value: 0xFFFFFFFF) or 'x' to cancel: 0x1
Wrote 0x100 bytes to offset 0x0

01 00 00 00 02 00 00 00 03 00 00 00 04 00 00 00
05 00 00 00 06 00 00 00 07 00 00 00 08 00 00 00
09 00 00 00 0A 00 00 00 0B 00 00 00 0C 00 00 00
0D 00 00 00 0E 00 00 00 0F 00 00 00 10 00 00 00
11 00 00 00 12 00 00 00 13 00 00 00 14 00 00 00
15 00 00 00 16 00 00 00 17 00 00 00 18 00 00 00
19 00 00 00 1A 00 00 00 1B 00 00 00 1C 00 00 00
1D 00 00 00 1E 00 00 00 1F 00 00 00 20 00 00 00
21 00 00 00 22 00 00 00 23 00 00 00 24 00 00 00
25 00 00 00 26 00 00 00 27 00 00 00 28 00 00 00
29 00 00 00 2A 00 00 00 2B 00 00 00 2C 00 00 00
2D 00 00 00 2E 00 00 00 2F 00 00 00 30 00 00 00
31 00 00 00 32 00 00 00 33 00 00 00 34 00 00 00
35 00 00 00 36 00 00 00 37 00 00 00 38 00 00 00
39 00 00 00 3A 00 00 00 3B 00 00 00 3C 00 00 00
3D 00 00 00 3E 00 00 00 3F 00 00 00 40 00 00 00

Write buffer and read buffer are identical
Press ENTER to continue
```

Figure 3.16 Block Transfer – Write, Read, and Compare Values

© 2025 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

4. APIs

The APIs are defined by Jungo's WinDriver.

Table 4.1 shows the list of APIs used by the host PCIe driver to enable SGDMA data transfer and its operation. For more details, refer to the links provided in the table below.

Table 4.1. List of APIs

WinDriver API	Description
OsEventCreate	Creates an event object.
OsEventWait	Waits until a specified event object is in the signaled state or the time-out interval elapses.
OsEventSignal	Sets the specified event object to the signaled state.
OsEventClose	Closes a handle to an event object.
WDC_IntEnable	• Enables interrupt handling for the device. • The software passes in a user-mode interrupt handler callback function, CPNXDMA_IntHandler/CPNXAXISDMA_IntHandler, which is called after an interrupt is received and processed in the kernel. • The last argument is set to TRUE as the driver uses a Kernel Plugin driver.
WDC_IntDisable	Disables interrupt handling for the device.
WDC_DMASGBufLock	• Locks a pre-allocated user-mode memory buffer for DMA which is passed in as 2nd argument of the API • Returns the corresponding physical mappings of the locked DMA pages as the 5th argument of the API.
WDC_DMAContigBufLock	• Allocates a contiguous descriptor buffer, locks it in physical memory • Returns mapping of the allocated descriptor buffer to physical address and to user-mode and kernel virtual address spaces.
WDC_DMABufUnlock	Unlocks and frees the memory allocated for a DMA buffer by a previous call to WDC_DMASGBufLock and WDC_DMAContigBufLock.
WDC_CallKerPlug	Sends a message from a user-mode application to a Kernel PlugIn driver, in this case, the software sends a message to start the DMA transfer.
kp_interlocked_init	Initializes a Kernel PlugIn interlocked counter.
kp_interlocked_read	• Reads to the value of a Kernel PlugIn interlocked counter, in this case, the software driver reads the • intReceived flag
kp_interlocked_set	Sets the value of a Kernel PlugIn interlocked counter to the specified value. Here, it is used to set the intReceived flag inside the kernel plugin interrupt handler function.
kp_interlocked_uninit	Uninitialized a Kernel PlugIn interlocked counter.
WDC_DMASyncCpu	Synchronizes all CPU caches with the DMA buffer by flushing the data from the CPU caches.

© 2025 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

5. Software Flow Diagrams

Note that some details of the software are omitted from the flowchart to simplify the diagrams and to describe the operation more clearly.

5.1. CPNXDMA_DmaOpen/CPNXAXISDMA_DmaOpen

This function opens a DMA handle and allocate and initialize CPNX DMA information structure, including allocation of the scatter-gather DMA buffer.

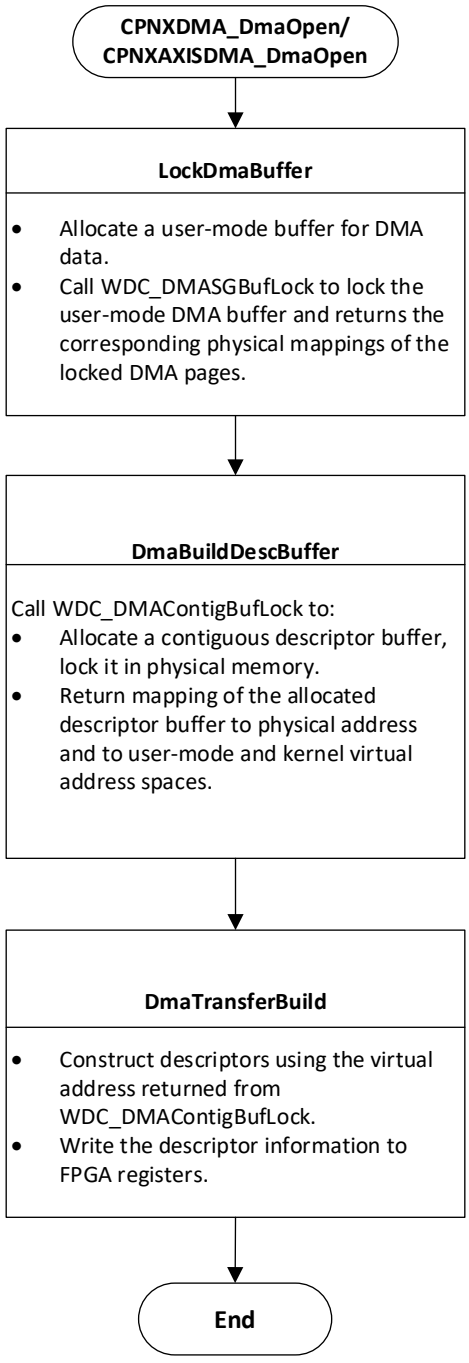

Figure 5.1. CPNXDMA_DmaOpen/CPNXAXISDMA_DmaOpen

© 2025 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

5.2. DmaPerformanceSingleDir

This function initializes the DMA and starts a thread to initiate DMA transfer and measure the DMA throughput.

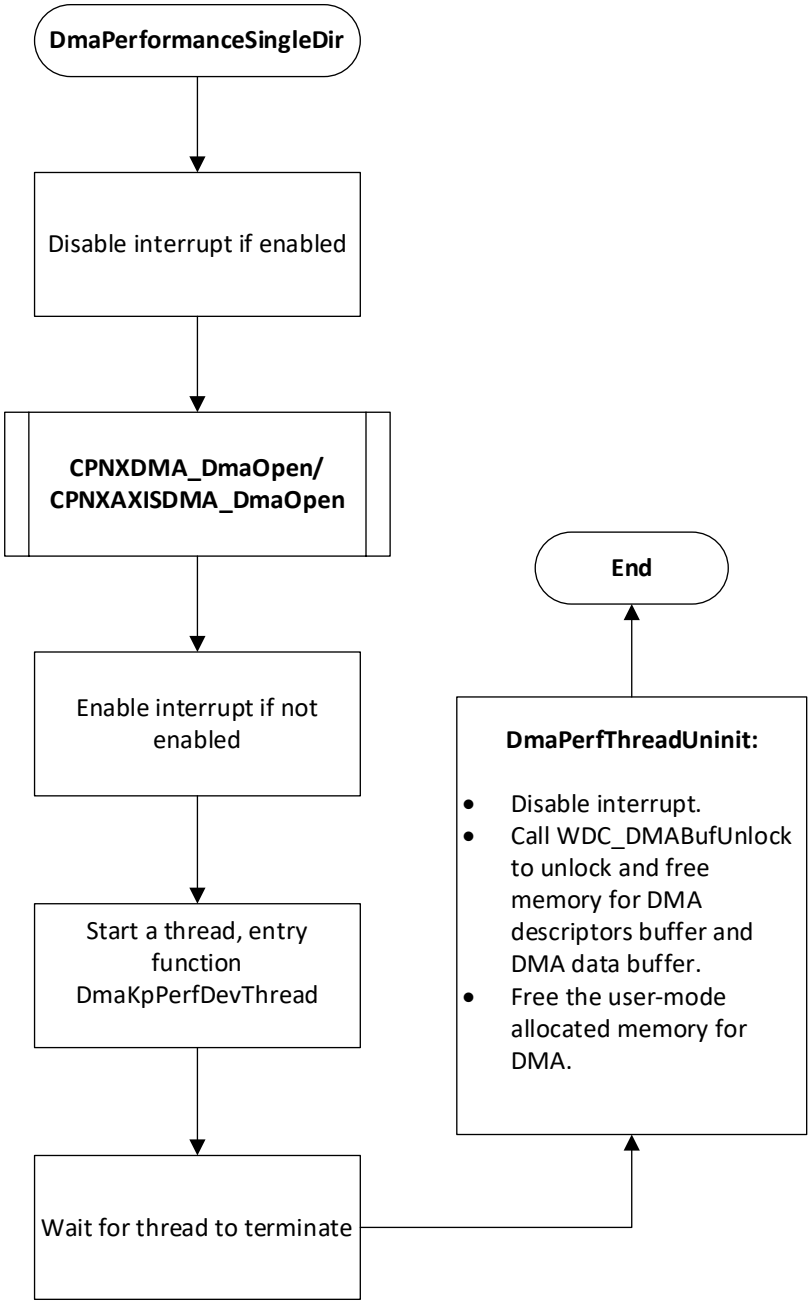

Figure 5.2. DmaPerformanceSingleDir

© 2025 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

5.3. DmaKpPerfDevThread

This is the thread function that sends a message to the kernel mode to start the DMA transfer, get the start time, poll for DMA completion, and get the end time for DMA throughput measurement.

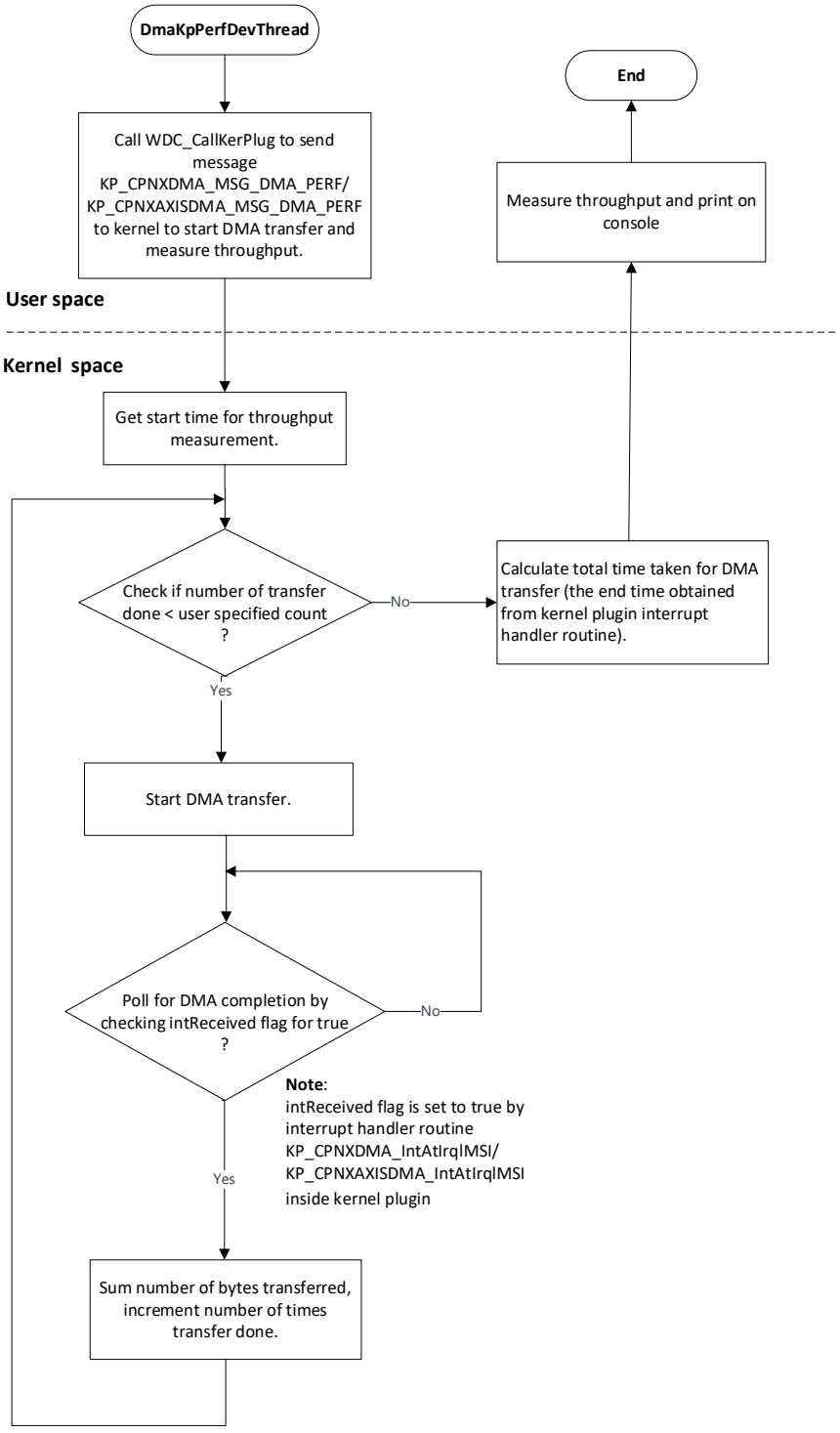

Figure 5.3. DmaKpPerfDevThread

© 2025 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

References

- Jungo WinDriver Official Documentation: Introduction
- PCIe x4 IP Release Notes (FPGA-RN-02059)
- PCIe X4 IP Core User Guide (FPGA-IPUG-02126)
- PCI Express x1/x2/x4 Endpoint IP Core User Guide (FPGA-IPUG-02009)
- CertusPro-NX web page
- Lattice Solutions IP Cores web page
- Lattice Insights web page for Lattice Semiconductor training courses and learning plans

© 2025 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

Technical Support Assistance

Submit a technical support case through www.latticesemi.com/techsupport.

For frequently asked questions, refer to the Lattice Answer Database at www.latticesemi.com/Support/AnswerDatabase.

© 2025 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

Revision History

Revision 1.2, April 2025

Section	Change Summary
All	• Added *and AXI-Stream* and *User Guide* to the document title. • Made editorial fixes.
Abbreviations in This Document	Added *AXI-MM*, *AXI-S*, *BAR*, *RO*, and *RW*.
Introduction	• Updated the description in the Introduction section. • Updated the Driver Version and Driver and IP Compatibility sections.
Software Setup	• Updated the description in the Software Setup section.

Application Overview	• Added the following figures:
	  • Figure 3.2 CPNXAXISDMA DMA Menu
	  • Figure 3.4 CPNXAXISDMA Transfer Direction
	  • Figure 3.6 CPNXAXISDMA Data Compare
	  • Figure 3.10 CPNXAXISDMA Measure DMA Performance
	  • Figure 3.11 CPNXAXISDMA Read/Write Memory
	  • Figure 3.12 CPNXAXISDMA Change Active Address Space for Read/Write
	  • Figure 3.13 CPNXAXISDMA Change Active Read/Write Mode
	  • Figure 3.14 CPNXAXISDMA Non-block or Block Transfer
	  • Figure 3.15 Non-block Transfer – Write, Read, and Compare Values
	  • Figure 3.16 Block Transfer – Write, Read, and Compare Values
	• Added the AXI-MM DMA Transfer section header and updated its content.
	• Updated the following figure captions:
	  • Figure 3.3. CPNXDMA Transfer Direction
	  • Figure 3.5. CPNXDMA Data Compare
	  • Figure 3.8. CPNXDMA Measure DMA Performance for Single-Direction Transfer
	  • Figure 3.9. CPNXDMA Measure DMA Performance for Bi-Directional Simultaneous Transfer
	• Added the following sections:
	  • AXI-S DMA Transfer
	  • AXI-S DMA Transfer and Compare Data
	  • AXI-S DMA Performance Measure
	  • Read/Write Memory and I/O Addresses
	• Added the following section headers:
	  • AXI-MM DMA Transfer and Compare Data
	  • AXI-MM DMA Performance Measure

APIs	• Updated the *WDC_IntEnable* description. • Added the *WDC_DMASyncCpu* API.

Software Flow Diagrams	• Added *CPNXAXISDMA_DmaOpen* to the CPNXDMA_DmaOpen/CPNXAXISDMA_DmaOpen section header.
	• Updated the following figures:
	  • Figure 5.1. CPNXDMA_DmaOpen/CPNXAXISDMA_DmaOpen and its caption
	  • Figure 5.2. DmaPerformanceSingleDir
	  • Figure 5.3. DmaKpPerfDevThread

References	• Added the *PCIe x4 IP Release Notes (FPGA-RN-02059)*.

Revision 1.1, January 2025

Section	Change Summary
All	Made editorial fixes.
Abbreviations in This Document	Added *Field Programmable Gate Array* (*FPGA*) and *Random Access Memory* (*RAM*).

© 2025 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

Section	Change Summary
Introduction	• Updated the Driver Version section. • Updated *Driver Version* in Table 1.1. Driver and IP Compatibility.

Software Setup	• Updated the driver version in the Driver Installation on a Windows Machine and Driver Installation on a Linux Machine sections.
	• Updated the programming code in step 6 of the Driver Installation on a Linux Machine section.
	• Updated the following figures and their captions:
	  • Figure 2.1. Setup
	  • Figure 2.2. Setup License Agreement
	  • Figure 2.3. Setup Install Location
	  • Figure 2.4. Setup Install
	  • Figure 2.6. lsmod (caption not updated)
	  • Figure 2.7. Run cpnxdma (caption not updated)

Application Overview	• Updated Figure 3.2. DMA Transfer Direction and Figure 3.3. DMA Data Compare. • Updated the DMA Performance Measure section. • Added Figure 3.5. Measure DMA performance for Single Direction Transfer. • Updated Figure 3.6. Measure DMA Performance for Bi-Directional Simultaneous Transfer and its caption.

Revision 1.0, December 2024

Section	Change Summary
All	Initial release.

© 2025 Lattice Semiconductor Corp. All Lattice trademarks, registered trademarks, patents, and disclaimers are as listed at www.latticesemi.com/legal.
All other brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

www.latticesemi.com