NVIDIA DOCA DMA

Sample Guide

MLNX-15-060523 _v1.5.1 | November 2022

Table of Contents

Chapter 1. Introduction

DOCA Direct Memory Access (DMA) library provides an API for copying data between buffers using hardware acceleration, supporting both local and remote memory regions.

The library provides an API for executing DMA operations on DOCA buffers. These buffers can reside in local memory (i.e., within the DPU) or remote memory (i.e., on the host).

Using DOCA DMA you can easily execute complex memory copy operations in an optimized, hardware-accelerated way:

- The `dma_local_copy` sample copies content between two local buffers on the DPU.
- The `dma_copy_dpu`/`dma_copy_host` copies user-defined text from the host to the DPU.

Note: DMA Copy Host must be run before DMA Copy DPU.

For more information about the DOCA DMA library, refer to NVIDIA DOCA DMA Programing Guide.

Chapter 2. Dependencies

The library requires a minimum DPU firmware version of 24.33.0100.

Chapter 3. Prerequisites

N/A

Chapter 4. Running the Sample

1. Refer to the following documents:
   - NVIDIA DOCA Installation Guide for Linux for details on how to install BlueField-related software.
   - NVIDIA DOCA Troubleshooting Guide for any issue you may encounter with the installation, compilation, or execution of DOCA samples.
2. To build a given sample:

```
cd /opt/mellanox/doca/samples/doca_dma/<sample_name>
meson build
ninja -C build
```

   Note: The binary `doca_<sample_name>` will be created under `./build/`.

3. Sample (e.g., `dma_copy_host`) usage:

Sample	Argument	Description
DMA Local Copy	`-p, --pci`	DOCA DMA device PCIe address
	`-t, --text`	Text to DMA copy from one local buffer to another
DMA Copy Host	`-p, --pci`	DOCA DMA device PCIe address
	`-t, --text`	Text to DMA copy from the host to the DPU
	`-d, --descriptor-path`	Path on which the exported descriptor file is saved
	`-b, --buffer-path`	Path on which the buffer information file is saved
DMA Copy DPU	`-p, --pci`	DOCA DMA device PCIe address
	`-d, --descriptor-path`	Path from which the exported descriptor file is read
	`-b, --buffer-path`	Path from which the buffer information file is read

4. For additional information per sample, use the `-h` option:

```
./build/doca_<sample_name> -h
```

Chapter 5. Samples

5.1. DMA Local Copy

This sample illustrates how to locally copy memory with DMA from one buffer to another on the DPU. This sample should be run on the DPU.

The sample logic includes:

1. Locating DOCA device.
2. Initializing needed DOCA core structures.
3. Populating DOCA memory map with two relevant buffers.
4. Allocating element in DOCA buffer inventory for each buffer.
5. Initializing DOCA DMA job object.
6. Submitting DMA job into work queue.
7. Retrieving DMA job from the queue once it is done.
8. Checking job result.
9. Destroying all DMA and DOCA core structures.

Reference:

- `/opt/mellanox/doca/samples/doca_dma/dma_local_copy/dma_local_copy_sample.c`
- `/opt/mellanox/doca/samples/doca_dma/dma_local_copy/dma_local_copy_main.c`
- `/opt/mellanox/doca/samples/doca_dma/dma_local_copy/dma_local_copy_main.c`

5.2. DMA Copy DPU

Note: This sample should run only after DMA Copy Host is run and the required configuration files (descriptor and buffer) have been copied to the DPU.

This sample illustrates how to copy memory (which contains user-defined text) with DMA from the x86 host into the DPU. This sample should be run on the DPU.

The sample logic includes:

1. Locating DOCA device.
2. Initializing needed DOCA core structures.
3. Reading configuration files and saving their content into local buffers.
4. Allocating the local destination buffer in which the host text will be saved.
5. Populating DOCA memory map with destination buffer.
6. Creating the remote memory map with the export descriptor file.
7. Creating memory map to the remote buffer.
8. Allocating element in DOCA buffer inventory for each buffer.
9. Initializing DOCA DMA job object.
10. Submitting DMA job into work queue.
11. Retrieving DMA job from the queue once it is done.
12. Checking DMA job result.
13. If the DMA job ends successfully, printing the text that has been copied to log.
14. Printing to log that the host-side sample can be closed.
15. Destroying all DMA and DOCA core structures.

Reference:

- `/opt/mellanox/doca/samples/doca_dma/dma_copy_dpu/dma_copy_dpu_sample.c`
- `/opt/mellanox/doca/samples/doca_dma/dma_copy_dpu/dma_copy_dpu_main.c`
- `/opt/mellanox/doca/samples/doca_dma/dma_copy_dpu/meson.build`

5.3. DMA Copy Host

> **Note:** This sample should be run first. It is user responsibility to transfer the two configuration files (descriptor and buffer) to the DPU and provide their path to the DMA Copy DPU sample.

This sample illustrates how to allow memory copy with DMA from the x86 host into the DPU. This sample should be run on the host.

The sample logic includes:

1. Locating DOCA device.
2. Initializing needed DOCA core structures.
3. Populating DOCA memory map with source buffer.
4. Exporting memory map.
5. Saving export descriptor and local DMA buffer information into files. These files should be transferred to the DPU before running the DPU sample.
6. Waiting until DPU DMA sample has finished.
7. Destroying all DMA and DOCA core structures.

Reference:

- `/opt/mellanox/doca/samples/doca_dma/dma_copy_host/dma_copy_host_sample.c`

- `/opt/mellanox/doca/samples/doca_dma/dma_copy_host/dma_copy_host_main.c`
- `/opt/mellanox/doca/samples/doca_dma/dma_copy_host/meson.build`

Notice

This document is provided for information purposes only and shall not be regarded as a warranty of a certain functionality, condition, or quality of a product. NVIDIA Corporation nor any of its direct or indirect subsidiaries and affiliates (collectively: "NVIDIA") make no representations or warranties, expressed or implied, as to the accuracy or completeness of the information contained in this document and assume no responsibility for any errors contained herein. NVIDIA shall have no liability for the consequences or use of such information or for any infringement of patents or other rights of third parties that may result from its use. This document is not a commitment to develop, release, or deliver any Material (defined below), code, or functionality.

NVIDIA reserves the right to make corrections, modifications, enhancements, improvements, and any other changes to this document, at any time without notice.

Customer should obtain the latest relevant information before placing orders and should verify that such information is current and complete.

NVIDIA products are sold subject to the NVIDIA standard terms and conditions of sale supplied at the time of order acknowledgement, unless otherwise agreed in an individual sales agreement signed by authorized representatives of NVIDIA and customer ("Terms of Sale"). NVIDIA hereby expressly objects to applying any customer general terms and conditions with regards to the purchase of the NVIDIA product referenced in this document. No contractual obligations are formed either directly or indirectly by this document.

NVIDIA products are not designed, authorized, or warranted to be suitable for use in medical, military, aircraft, space, or life support equipment, nor in applications where failure or malfunction of the NVIDIA product can reasonably be expected to result in personal injury, death, or property or environmental damage. NVIDIA accepts no liability for inclusion and/or use of NVIDIA products in such equipment or applications and therefore such inclusion and/or use is at customer's own risk.

NVIDIA makes no representation or warranty that products based on this document will be suitable for any specified use. Testing of all parameters of each product is not necessarily performed by NVIDIA. It is customer's sole responsibility to evaluate and determine the applicability of any information contained in this document, ensure the product is suitable and fit for the application planned by customer, and perform the necessary testing for the application in order to avoid a default of the application or the product. Weaknesses in customer's product designs may affect the quality and reliability of the NVIDIA product and may result in additional or different conditions and/or requirements beyond those contained in this document. NVIDIA accepts no liability related to any default, damage, costs, or problem which may be based on or attributable to: (i) the use of the NVIDIA product in any manner that is contrary to this document or (ii) customer product designs.

No license, either expressed or implied, is granted under any NVIDIA patent right, copyright, or other NVIDIA intellectual property right under this document. Information published by NVIDIA regarding third-party products or services does not constitute a license from NVIDIA to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property rights of the third party, or a license from NVIDIA under the patents or other intellectual property rights of NVIDIA.

Reproduction of information in this document is permissible only if approved in advance by NVIDIA in writing, reproduced without alteration and in full compliance with all applicable export laws and regulations, and accompanied by all associated conditions, limitations, and notices.

THIS DOCUMENT AND ALL NVIDIA DESIGN SPECIFICATIONS, REFERENCE BOARDS, FILES, DRAWINGS, DIAGNOSTICS, LISTS, AND OTHER DOCUMENTS (TOGETHER AND SEPARATELY, "MATERIALS") ARE BEING PROVIDED "AS IS." NVIDIA MAKES NO WARRANTIES, EXPRESSED, IMPLIED, STATUTORY, OR OTHERWISE WITH RESPECT TO THE MATERIALS, AND EXPRESSLY DISCLAIMS ALL IMPLIED WARRANTIES OF NONINFRINGEMENT, MERCHANTABILITY, AND FITNESS FOR A PARTICULAR PURPOSE. TO THE EXTENT NOT PROHIBITED BY LAW, IN NO EVENT WILL NVIDIA BE LIABLE FOR ANY DAMAGES, INCLUDING WITHOUT LIMITATION ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL, PUNITIVE, OR CONSEQUENTIAL DAMAGES, HOWEVER CAUSED AND REGARDLESS OF THE THEORY OF LIABILITY, ARISING OUT OF ANY USE OF THIS DOCUMENT, EVEN IF NVIDIA HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. Notwithstanding any damages that customer might incur for any reason whatsoever, NVIDIA's aggregate and cumulative liability towards customer for the products described herein shall be limited in accordance with the Terms of Sale for the product.

Trademarks

NVIDIA, the NVIDIA logo, and Mellanox are trademarks and/or registered trademarks of Mellanox Technologies Ltd. and/or NVIDIA Corporation in the U.S. and in other countries. The registered trademark Linux® is used pursuant to a sublicense from the Linux Foundation, the exclusive licensee of Linus Torvalds, owner of the mark on a world¬wide basis. Other company and product names may be trademarks of the respective companies with which they are associated.

Copyright