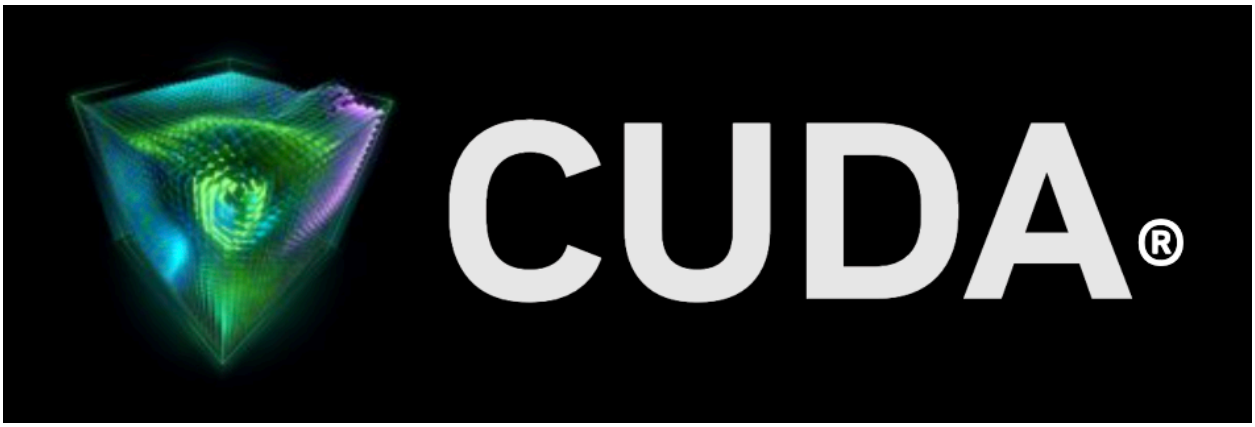

Ada Tuning Guide

Release 12.6

NVIDIA Corporation

Jul 23, 2024

Contents

Tuning CUDA Applications for NVIDIA Ada GPU Architecture

The programming guide for tuning CUDA Applications for GPUs based on the NVIDIA Ada GPU Architecture.

Chapter 1. NVIDIA Ada GPU Architecture

The NVIDIA® Ada GPU architecture is NVIDIA's latest architecture for CUDA® compute applications. The NVIDIA Ada GPU architecture retains and extends the same CUDA programming model provided by previous NVIDIA GPU architectures such as NVIDIA Ampere and Turing, and applications that follow the best practices for those architectures should typically see speedups on the NVIDIA Ada architecture without any code changes. This guide summarizes the ways that an application can be fine-tuned to gain additional speedups by leveraging the NVIDIA Ada GPU architecture's features.[1]

For further details on the programming features discussed in this guide, please refer to the CUDA C++ Programming Guide.

[1] Throughout this guide, *Volta* refers to devices of compute capability 7.0, *Turing* refers to devices of compute capability 7.5, *NVIDIA Ampere GPU Architecture* refers to devices of compute capability 8.0 and 8.6, *NVIDIA Ada* refers to devices of compute capability 8.9.

Chapter 2. CUDA Best Practices

The performance guidelines and best practices described in the CUDA C++ Programming Guide and the CUDA C++ Best Practices Guide apply to all CUDA-capable GPU architectures. Programmers must primarily focus on following those recommendations to achieve the best performance.

The high-priority recommendations from those guides are as follows:

- Find ways to parallelize sequential code.
- Minimize data transfers between the host and the device.
- Adjust kernel launch configuration to maximize device utilization.
- Ensure global memory accesses are coalesced.
- Minimize redundant accesses to global memory whenever possible.
- Avoid long sequences of diverged execution by threads within the same warp.

Chapter 3. Application Compatibility

Before addressing specific performance tuning issues covered in this guide, refer to the NVIDIA Ada GPU Architecture Compatibility Guide for CUDA Applications to ensure that your application is compiled in a way that is compatible with the NVIDIA Ada GPU Architecture.

Chapter 4. NVIDIA Ada GPU Architecture Tuning

4.1. Streaming Multiprocessor

The NVIDIA Ada GPU architecture's Streaming Multiprocessor (SM) provides the following improvements over Turing and NVIDIA Ampere GPU architectures.

4.1.1. Occupancy

The maximum number of concurrent warps per SM is 48, remaining the same compared to compute capability 8.6 GPUs, and other factors influencing warp occupancy are:

- The register file size is 64K 32-bit registers per SM.
- The maximum number of registers per thread is 255.
- The maximum number of thread blocks per SM is 24.
- The shared memory capacity per SM is 100 KB.
- The maximum shared memory per thread block is 99 KB.

Overall, developers can expect similar occupancy as on compute capability 8.6 GPUs without changes to their application.

4.1.2. Improved Tensor Core Operations

The NVIDIA Ada GPU architecture includes new Ada Fourth Generation Tensor Cores featuring the Hopper FP8 Transformer Engine.

4.1.3. Improved FP32 throughput

Devices of compute capability 8.9 have 2x more FP32 operations per cycle per SM than devices of compute capability 8.0. While a binary compiled for 8.0 will run as-is on 8.9, it is recommended to compile explicitly for 8.9 to benefit from the increased FP32 throughput.

4.2. Memory System

4.2.1. Increased L2 capacity

The NVIDIA Ada GPU architecture increases the capacity of the L2 cache to 98304 KB in AD102, 16x larger than GA102. The NVIDIA Ada GPU architecture allows CUDA users to control the persistence of data in the L2 cache. For more information on the persistence of data in the L2 cache, refer to the section on managing the L2 cache in the CUDA C++ Programming Guide.

4.2.2. Unified Shared Memory/L1/Texture Cache

NVIDIA Ada architecture features a unified L1 cache, texture cache, and shared memory similar to that of the NVIDIA Ampere architecture. The combined L1 cache capacity is 128 KB.

In the NVIDIA Ada GPU architecture, the portion of the L1 cache dedicated to shared memory (known as the *carveout*) can be selected at runtime as in previous architectures, such as NVIDIA Ampere, using `cudaFuncSetAttribute()` with the attribute `cudaFuncAttributePreferredSharedMemoryCarveout`. The NVIDIA Ada GPU architecture supports shared memory capacity of 0, 8, 16, 32, 64 or 100 KB per SM.

CUDA reserves 1 KB of shared memory per thread block. Hence, GPUs with compute capability 8.9 can address up to 99 KB of shared memory in a single thread block. To maintain architectural compatibility, static shared memory allocations remain limited to 48 KB, and an explicit opt-in is also required to enable dynamic allocations above this limit. See the CUDA C++ Programming Guide for details.

Like the NVIDIA Ampere and NVIDIA Volta GPU architectures, the NVIDIA Ada GPU architecture combines the functionality of the L1 and texture caches into a unified L1/Texture cache that acts as a coalescing buffer for memory accesses, gathering up the data requested by the threads of a warp prior to delivery of that data to the warp. Another benefit of its union with shared memory, similar to previous architectures, is improvement in terms of both latency and bandwidth.

Chapter 5. Revision History

Version 1.0

- Initial Public Release
- Added support for compute capability 8.9

Chapter 6. Notices

6.1. 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 ("NVIDIA") makes no representations or warranties, expressed or implied, as to the accuracy or completeness of the information contained in this document and assumes 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.

6.2. OpenCL

OpenCL is a trademark of Apple Inc. used under license to the Khronos Group Inc.

6.3. Trademarks

NVIDIA and the NVIDIA logo are trademarks or registered trademarks of NVIDIA Corporation in the U.S. and other countries. Other company and product names may be trademarks of the respective companies with which they are associated.

Copyright