Page: Details | Result: 0 - 115 - MatrixMulKernel | Add Baseline | Apply Rules | Occupancy Calculator | Save as PDF

	Result	Time	Cycles	Regs	GPU	SM Frequency	CC	Process
Current	115 - MatrixMulKernel (14, 48, 1)x(32, 32, 1)	43.59 msecond	٦٠٬٧٣٥٬٢٩٣	٤٠	0 - NVIDIA GeForce RTX 3090	1.39 cycle/nsecond	8.6	[251156] task3

GPU Speed Of Light Throughput

High-level overview of the throughput for compute and memory resources of the GPU. For each unit, the throughput reports the achieved percentage of utilization with respect to the theoretical maximum. Breakdowns show the throughput for each individual sub-metric of Compute and Memory to clearly identify the highest contributor.

Compute (SM) Throughput [%]	89.32	Duration [msecond]	43.59
Memory Throughput [%]	13.50	Elapsed Cycles [cycle]	60,735,293
L1/TEX Cache Throughput [%]	14.82	SM Active Cycles [cycle]	55,317,676.05
L2 Cache Throughput [%]	3.07	SM Frequency [cycle/nsecond]	1.39
DRAM Throughput [%]	1.28	DRAM Frequency [cycle/nsecond]	9.49

High Throughput The kernel is utilizing greater than 80.0% of the available compute or memory performance of the device. To further improve performance, work will likely need to be shifted from the most utilized to another unit. Start by analyzing workloads in the Compute Workload Analysis section.

Launch Statistics

Summary of the configuration used to launch the kernel. The launch configuration defines the size of the kernel grid, the division of the grid into blocks, and the GPU resources needed to execute the kernel. Choosing an efficient launch configuration maximizes device utilization.

Grid Size	672	Function Cache Configuration	cudaFuncCachePreferNone
Registers Per Thread [register/thread]	40	Static Shared Memory Per Block [Kbyte/block]	40.96
Block Size	1,024	Dynamic Shared Memory Per Block [byte/block]	0
Threads [thread]	688,128	Driver Shared Memory Per Block [Kbyte/block]	1.02
Waves Per SM	8.20	Shared Memory Configuration Size [Kbyte]	65.54

Occupancy

Occupancy is the ratio of the number of active warps per multiprocessor to the maximum number of possible active warps. Another way to view occupancy is the percentage of the hardware's ability to process warps that is actively in use. Higher occupancy does not always result in higher performance, however, low occupancy always reduces the ability to hide latencies, resulting in overall performance degradation. Large discrepancies between the theoretical and the achieved occupancy during execution typically indicates highly imbalanced workloads.

Theoretical Occupancy [%]	66.67	Block Limit Registers [block]	1
Theoretical Active Warps per SM [warp]	32	Block Limit Shared Mem [block]	1
Achieved Occupancy [%]	66.62	Block Limit Warps [block]	1
Achieved Active Warps Per SM [warp]	31.98	Block Limit SM [block]	16

Occupancy Limiters This kernel's theoretical occupancy (66.7%) is limited by the number of required registers This kernel's theoretical occupancy (66.7%) is limited by the number of warps within each block This kernel's theoretical occupancy (66.7%) is limited by the required amount of shared memory See the CUDA Best Practices Guide for more details on optimizing occupancy.