Result	Size	Time	Cycles	GPU	SM Frequency	Process	Attributes
Current: 563 - reduce_argmin	(4, 1, 1)x(1024, 1, 1)	13.09 us	7,622	0 - Tesla T4	582.32 Mhz	[12774] exe	

Summary | Details | Source | Context | Comments | Raw | Session — Compare | Tools | View | Export

GPU Speed Of Light Throughput

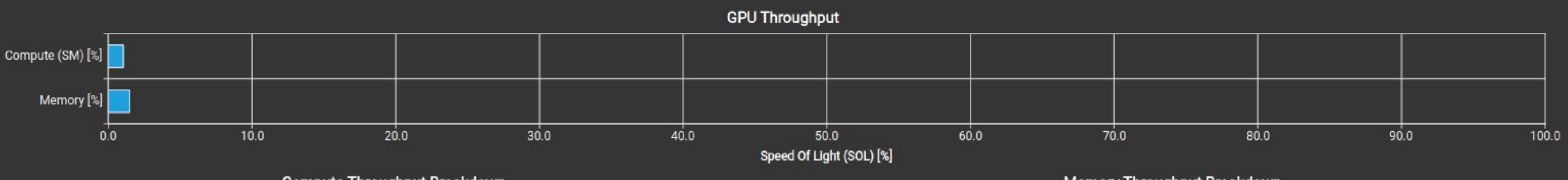
High-level overview of the throughput for compute and memory resources of the GPU. For each unit, the throughput reports the achieved percentage of utilization with respect to the theoretical maximum. Breakdowns show the throughput for each individual sub-metric of Compute and Memory to clearly identify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

Compute (SM) Throughput [%]	1.03	Duration [us]	13.09
Memory Throughput [%]	1.47	Elapsed Cycles [cycle]	7,622
L1/TEX Cache Throughput [%]	14.98	SM Active Cycles [cycle]	516.67
L2 Cache Throughput [%]	0.57	SM Frequency [Mhz]	582.32
DRAM Throughput [%]	1.47	DRAM Frequency [Ghz]	4.89

Small Grid This kernel grid is too small to fill the available resources on this device, resulting in only 0.1 full waves across all SMs. Look at Launch Statistics for more details.

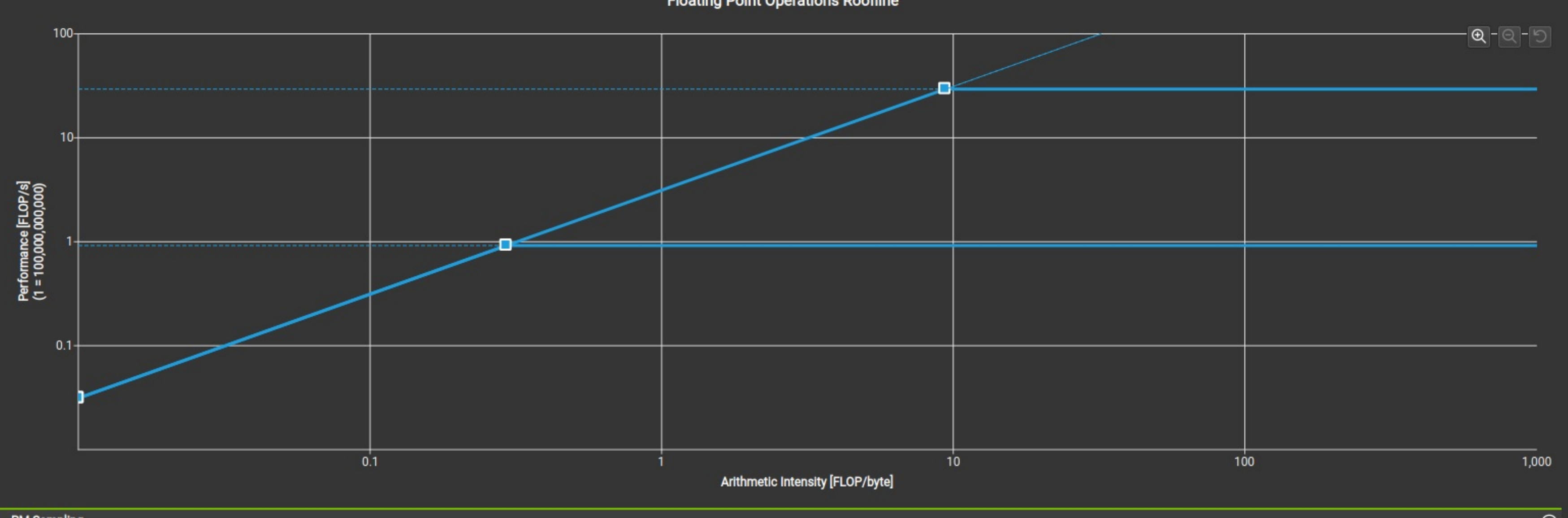
Roofline Analysis The ratio of peak float (fp32) to double (fp64) performance on this device is 32:1. The kernel achieved 0% of this device's fp32 peak performance and 0% of its fp64 peak performance. See the Kernel Profiling Guide for more details on roofline analysis.

GPU Throughput



Compute Throughput Breakdown		Memory Throughput Breakdown	
SM: Inst Executed Pipe Lsu [%]	1.03	DRAM: Cycles Active [%]	1.47
SM: Issue Active [%]	0.54	DRAM: Dram Sectors [%]	1.06
SM: Inst Executed [%]	0.47	L1: Lsuin Requests [%]	1.03
SM: Mio Inst Issued [%]	0.41	L1: Data Pipe Lsu Wavefronts [%]	0.67
SM: Pipe Alu Cycles Active [%]	0.26	L2: T Sectors [%]	0.57
SM: Mio Pq Write Cycles Active [%]	0.21	L2: Lts2xbar Cycles Active [%]	0.54
SM: Inst Executed Pipe Adu [%]	0.19	L1: M Xbar2l1tex Read Sectors [%]	0.52
SM: Mio Pq Read Cycles Active [%]	0.19	L2: D Sectors Fill Device [%]	0.44
SM: Pipe Fma Cycles Active [%]	0.15	L1: Lsu Writeback Active [%]	0.30
SM: Mio2rf Writeback Active [%]	0.15	L2: D Sectors [%]	0.25
SM: Inst Executed Pipe Uniform [%]	0.11	GPU: Compute Memory Access Throughput Internal Activity [%]	0.24
SM: Inst Executed Pipe Cbu Pred On Any [%]	0.10	L2: T Tag Requests [%]	0.18
SM: Inst Executed Pipe Tex [%]	0.01	L2: Xbar2lts Cycles Active [%]	0.16
SM: Memory Throughput Internal Activity [%]	0	L1: M L1tex2xbar Req Cycles Active [%]	0.14
SM: Pipe Tensor Cycles Active [%]	0	L1: Data Bank Writes [%]	0.13
SM: Pipe Shared Cycles Active [%]	0	L1: Data Bank Reads [%]	0.11
SM: Pipe Fp64 Cycles Active [%]	0	L1: Texin Sm2tex Req Cycles Active [%]	0.06
IDC: Request Cycles Active [%]	0	L1: F Wavefronts [%]	0.04
SM: Instruction Throughput Internal Activity [%]	0	L2: D Atomic Input Cycles Active [%]	0.01
SM: Inst Executed Pipe Xu [%]	0	L1: Tex Writeback Active [%]	0
SM: Inst Executed Pipe Ipa [%]	0	L2: D Sectors Fill Sysmem [%]	0
SM: Inst Executed Pipe Fp16 [%]	0	L1: Data Pipe Tex Wavefronts [%]	0
		GPU: Compute Memory Request Throughput Internal Activity [%]	0

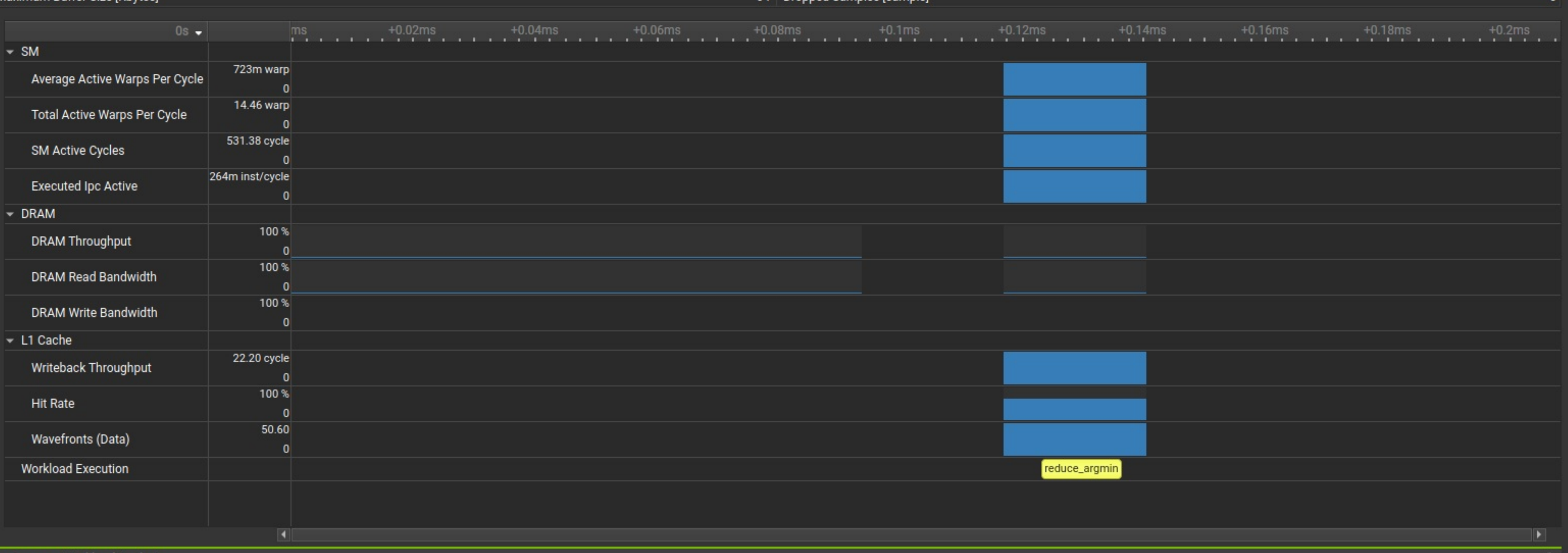
Floating Point Operations Roofline



PM Sampling

Timeline view of PM metrics sampled periodically over the workload duration. Data is collected across multiple passes. Use this section to understand how workload behavior changes over its runtime.

Maximum Sampling Interval [cycle]	20,000	# Pass Groups	1
Maximum Buffer Size [Kbytes]	64	Dropped Samples [sample]	0



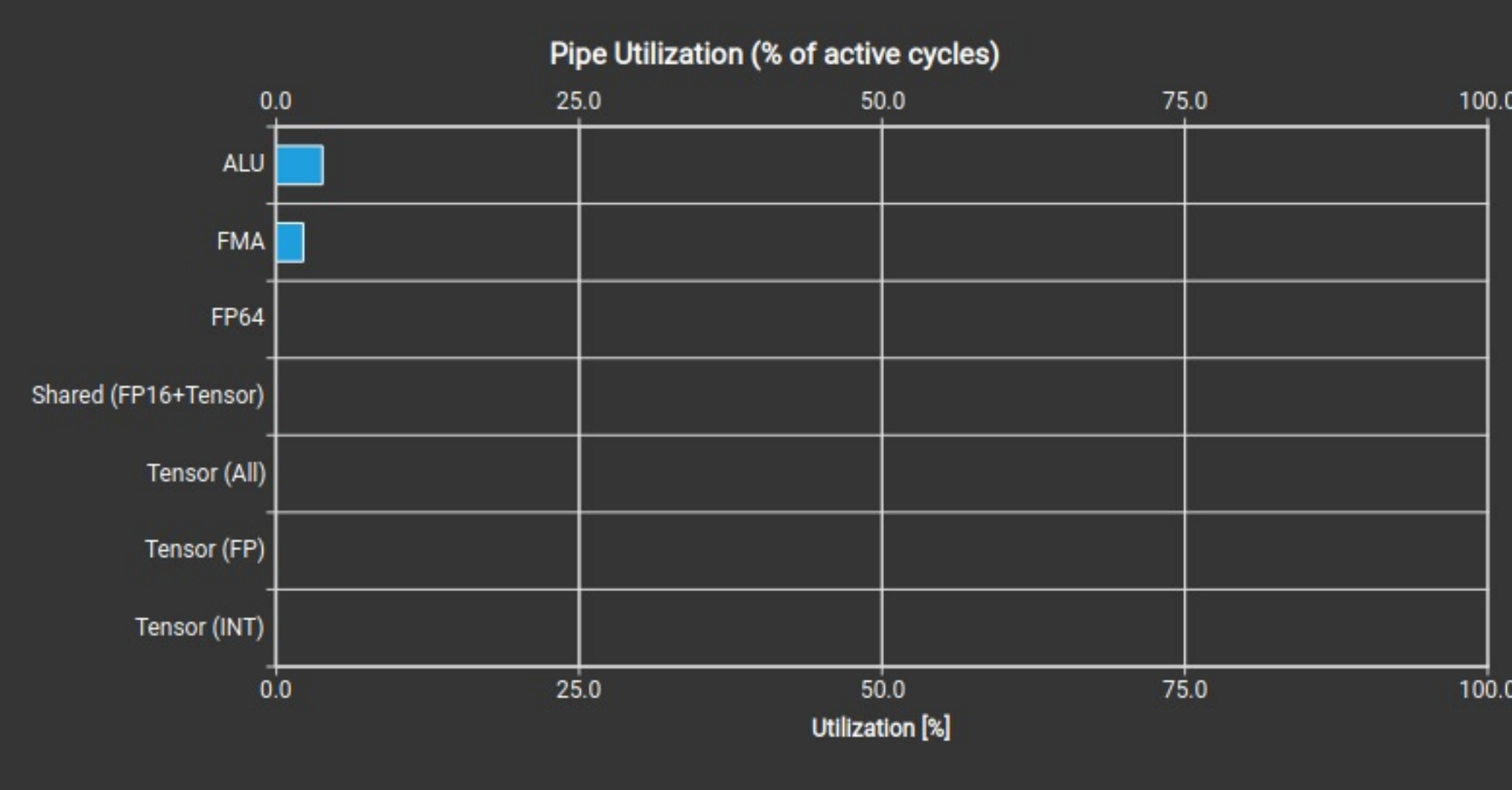
Compute Workload Analysis

Detailed analysis of the compute resources of the streaming multiprocessors (SM), including the achieved instructions per clock (IPC) and the utilization of each available pipeline. Pipelines with very high utilization might limit the overall performance.

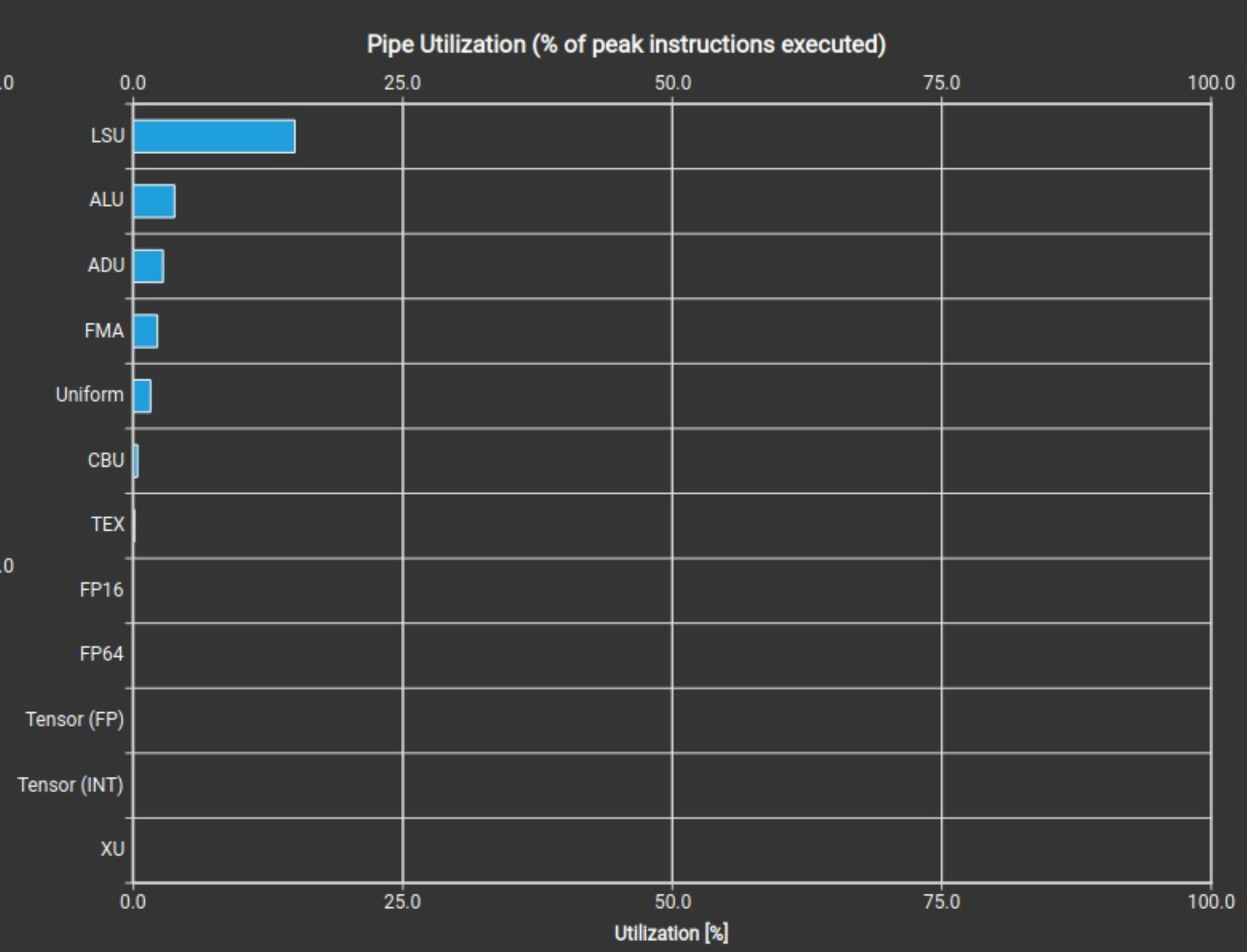
Executed Ipc Elapsed [inst/cycle]	0.02	SM Busy [%]	7.80
Executed Ipc Active [inst/cycle]	0.27	Issue Slots Busy [%]	7.80
Issued Ipc Active [inst/cycle]	0.31		

Low Utilization Est. Local Speedup: 96.19% All compute pipelines are under-utilized. Either this kernel is very small or it doesn't issue enough warps per scheduler. Check the Launch Statistics and Scheduler Statistics sections for further details.

Pipe Utilization (% of active cycles)



Pipe Utilization (% of peak instructions executed)



Memory Workload Analysis

Detailed analysis of the memory resources of the GPU. Memory can become a limiting factor for the overall kernel performance when fully utilizing the involved hardware units (Mem Busy), exhausting the available communication bandwidth between those units (Max Bandwidth), or by reaching the maximum throughput of issuing memory instructions (Mem Pipes Busy). Detailed chart of the memory units. Detailed tables with data for each memory unit.

Memory Throughput [Gbyte/s]	4.61	Mem Busy [%]	0.67
L1/TEX Hit Rate [%]	0.19	Max Bandwidth [%]	1.47
L2 Hit Rate [%]	23.46	Mem Pipes Busy [%]	1.03

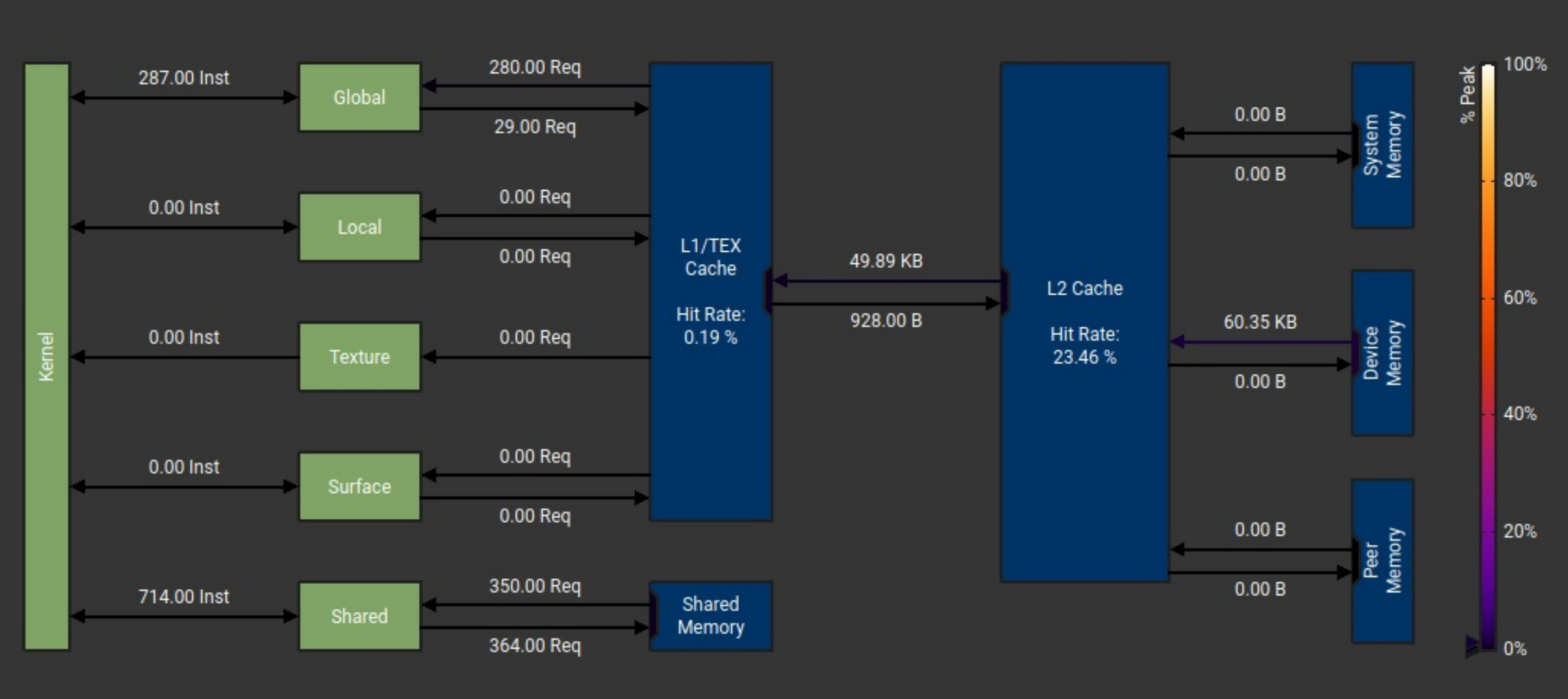
Memory L2 Compression The optional metric lts__average_gcomp_input_sector_success_rate.pct could not be found. Collecting it as an additional metric could enable the rule to provide more guidance.

DRAM Global Load Access Pattern The memory access pattern for global loads from DRAM might not be optimal. On average, only 31.9 of the 32 bytes transmitted per sector are utilized by each thread. This applies to the 95.7% of sectors missed in L2. This could possibly be caused by a stride between threads. Check the Source Counters section for uncoalesced global loads.

L1TEX Global Store Access Pattern Est. Speedup: 1.15% The memory access pattern for global stores to L1TEX might not be optimal. On average, only 4.6 of the 32 bytes transmitted per sector are utilized by each thread. This could possibly be caused by a stride between threads. Check the Source Counters section for uncoalesced global stores.

Memory Chart

Show As: Transfer Size



Shared Memory

	Instructions	Requests	Wavefronts	% Peak	Bank Conflicts
Shared Load	350	350	360	0.12	8
Shared Load Matrix	0	0			
Shared Store	364	364	502	0.17	0
Shared Atomic	0	0	0	0	0
Other	-	-	597	0.20	0
Total	714	714	1,459	0.49	8

L1/TEX Cache

	Instructions	Requests	Wavefronts	% Peak	Sectors	Sectors/Req	Hit Rate	Bytes	Sector Misses to L2	% Peak to L2	Returns to SM
Local Load	0	0	0	0	0	0	0	0	1,539	0.51	408
Global Load	260	260	260	0.09	1,540	5.92	0.06	49,280			
Surface Load	0	0	0	0	0	0	0	0	0	0	0
Texture Load	0	0	0	0	0	0	0	0	0	0	
Global Store	7	9	9	0.00	9	1	22.22	288	9	0.00	-
Local Store	0	0	0	0	0	0	0	0			-
Surface Store	0	0	0	0	0	0	0	0	0	0	-
Global Reduction	0	0	0	0	0	0	0	0	0	0	-
DSMEM Reduction						-	-		0	0	-
Surface Reduction	0	0	0	0	0	0	0	0	0	0	-
Global Atomic ALU	0	20	20	0.01	20	1	0	640	20	0.00	see above
Global Atomic CAS	20										
Surface Atomic ALU	0	0	0	0	0	0	0	0	0	0	see above
Surface Atomic CAS	0										
Loads	260	260	260	0.09	1,540	5.92	0.06	49,280	1,539	0.51	408
Stores	7	9	9	0.00	9	1	22.22	288	9	0.00	-
Atomics & Reductions	20	20	20	0.01	20	1	0	640	20	0.00	-

L2 Cache

	Requests	Sectors	Sectors/Req	% Peak	Hit Rate	Bytes	Throughput	Sector Misses to Device	Sector Misses to System	Sector Misses to Peer
L1/TEX Load	387	1,539	3.98	0.43	0.19	49,248	3,762,836,185.82	1,536	0	0
L1/TEX Store	9	9	1	0.00	77.78	288	22,004,889.98	2	0	0
L1/TEX Atomic ALU	0	0	0	0	0	0	0	0	0	0
L1/TEX Atomic CAS	20	20	1	0.01	90.48	640	48,899,755.50	2	0	0
L1/TEX Reduction	0	0	0	0	0	0	0	0	0	0
L1/TEX Total	416	1,568	3.77	0.44	1.85	50,176	3,833,740,831.30	1,540	0	0
ECC Total	-	10	-	0.00	-	320	24,449,877.75	10	-	-
GPU Total	626	2,029	3.24	0.57	23.45	64,928	4,960,880,195.60	1,554	0	0

Device Memory

	Sectors	% Peak	Bytes	Throughput
Load	1,886	1.47	60,352	4,611,246,943.77
Store	0	0	0	0
Total	1,886	1.47	60,352	4,611,246,943.77

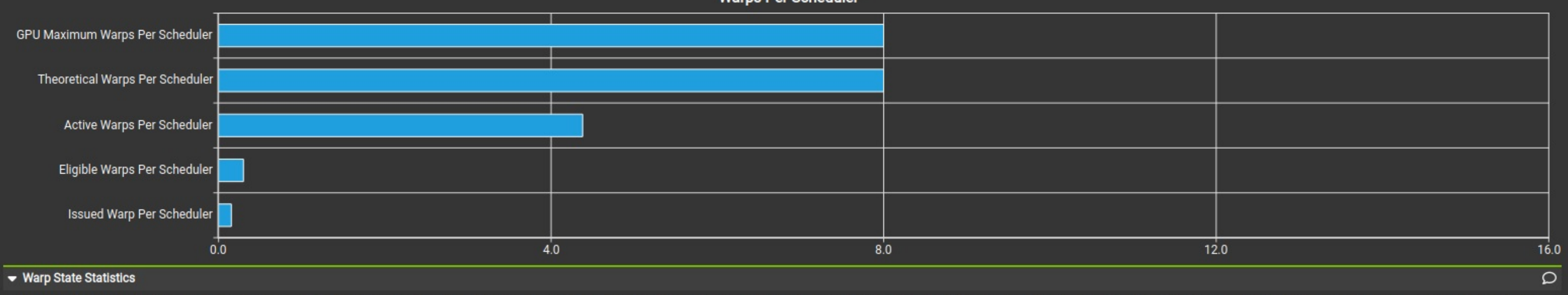
Scheduler Statistics

Summary of the activity of the schedulers issuing instructions. Each scheduler maintains a pool of warps that it can issue instructions for. The upper bound of warps in the pool (Theoretical Warps) is limited by the launch configuration. On every cycle each scheduler checks the state of the allocated warps in the pool (Active Warps). Active warps that are not stalled (Eligible Warps) are ready to issue their next instruction. From the set of eligible warps the scheduler selects a single warp from which to issue one or more instructions (Issued Warp). On cycles with no eligible warps, the issue slot is skipped and no instruction is issued. Having many skipped issue slots indicates poor latency hiding.

Active Warps Per Scheduler [warp]	4.38	No Eligible [%]	84.62
Eligible Warps Per Scheduler [warp]	0.30	One or More Eligible [%]	15.38
Issued Warp Per Scheduler	0.15		

Issue Slot Utilization Est. Local Speedup: 84.62% Every scheduler is capable of issuing one instruction per cycle, but for this kernel each scheduler only issues an instruction every 6.5 cycles. This might leave hardware resources underutilized and may lead to less optimal performance. Out of the maximum of 8 warps per scheduler, this kernel allocates an average of 4.38 active warps per scheduler, but only an average of 0.30 warps were eligible per cycle. Eligible warps are the subset of active warps that are ready to issue their next instruction. Every cycle with no eligible warp results in no instruction being issued and the issue slot remains unused. To increase the number of eligible warps, reduce the time the active warps are stalled by inspecting the top stall reasons on the Warp State Statistics and Source Counters sections.

Warps Per Scheduler



Warp State Statistics

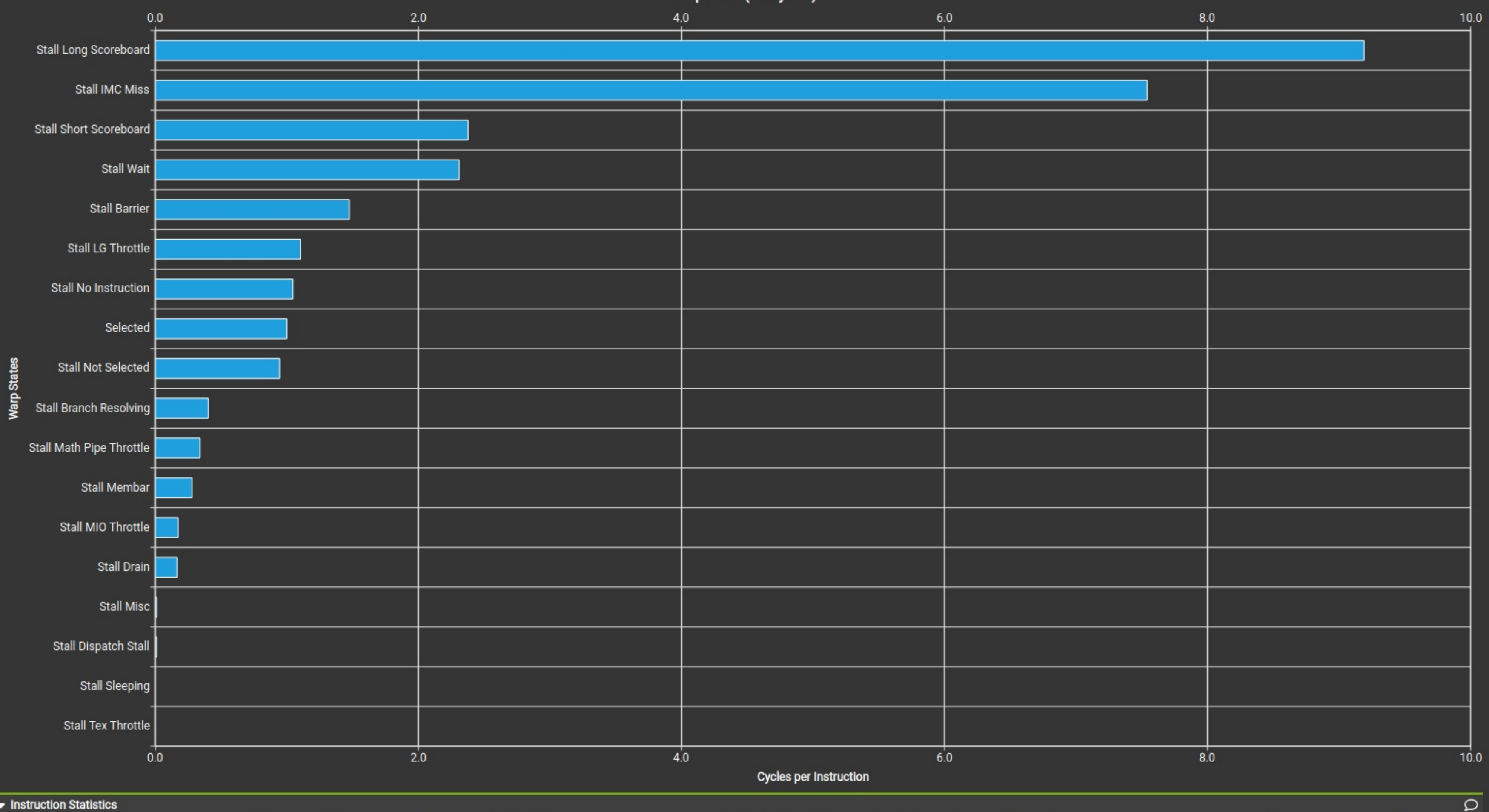
Analysis of the states in which all warps spent cycles during the kernel execution. The warp states describe a warp's readiness or inability to issue its next instruction. The warp cycles per instruction define the latency between two consecutive instructions. The higher the value, the more warp parallelism is required to hide this latency. For each warp state, the chart shows the average number of cycles spent in that state per issued instruction. Stalls are not always impacting the overall performance nor are they completely avoidable. Only focus on stall reasons if the schedulers fail to issue every cycle. When executing a kernel with mixed library and user code, these metrics show the combined values.

Warp Cycles Per Issued Instruction [cycle]	28.48	Avg. Active Threads Per Warp	29.21
Warp Cycles Per Executed Instruction [cycle]	32.68	Avg. Not Predicated Off Threads Per Warp	24.33

Long Scoreboard Stalls Est. Speedup: 32.26% On average, each warp of this kernel spends 9.2 cycles being stalled waiting for a scoreboard dependency on a L1TEX (local, global, surface, texture) operation. Find the instruction producing the data being waited upon to identify the culprit. To reduce the number of cycles waiting on L1TEX data accesses verify the memory access patterns are optimal for the target architecture, attempt to increase cache hit rates by increasing data locality (coalescing), or by changing the cache configuration. Consider moving frequently used data to shared memory. This stall type represents about 32.3% of the total average of 28.5 cycles between issuing two instructions.

Warp Stall Check the Warp Stall Sampling (All Samples) table for the top stall locations in your source based on sampling data. The Kernel Profiling Guide provides more details on each stall reason.

Warp State (All Cycles)

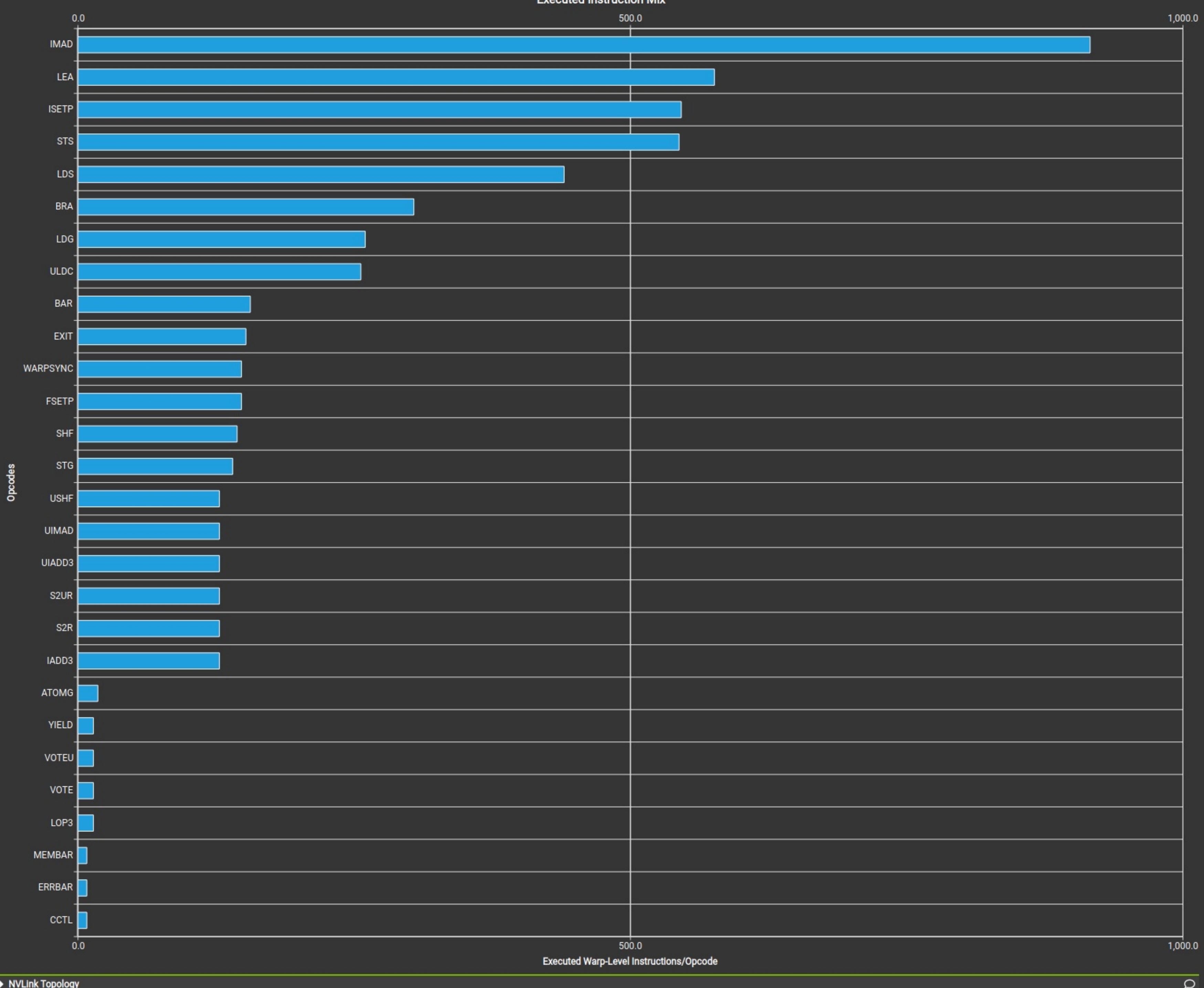


Instruction Statistics

Statistics of the executed low-level assembly instructions (SASS). The instruction mix provides insight into the types and frequency of the executed instructions. A narrow mix of instruction types implies a dependency on few instruction pipelines, while others remain unused. Using multiple pipelines allows hiding latencies and enables parallel execution. Note that 'Instructions/Opcode' and 'Executed Instructions' are measured differently and can diverge if cycles are spent in system calls.

Executed Instructions [inst]	5,616	Avg. Executed Instructions Per Scheduler [inst]	35.10
Issued Instructions [inst]	6,444	Avg. Issued Instructions Per Scheduler [inst]	40.27

Executed Instruction Mix



NVLink Topology

NVLink Topology diagram shows logical NVLink connections with transmit/receive throughput.

NVLink Tables

Detailed tables with properties for each NVLink.

NUMA Affinity

Non-uniform memory access (NUMA) affinities based on compute and memory distances for all GPUs.

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	4	Function Cache Configuration	CachePreferNone
Registers Per Thread [register/thread]	16	Static Shared Memory Per Block [Kbyte/block]	12.29
Block Size	1,024	Dynamic Shared Memory Per Block [byte/block]	0
Threads [thread]	4,096	Driver Shared Memory Per Block [byte/block]	0
Waves Per SM	0.10	Shared Memory Configuration Size [Kbyte]	32.77
Uses Green Context	0	# SMs [SM]	40

Small Grid Est. Speedup: 90.00% The grid for this launch is configured to execute only 4 blocks, which is less than the GPU's 40 multiprocessors. This can underutilize some multiprocessors. If you do not intend to execute this kernel concurrently with other workloads, consider reducing the block size to have at least one block per multiprocessor or increase the size of the grid to fully utilize the available hardware resources. See the Hardware Model description for more details on launch configurations.

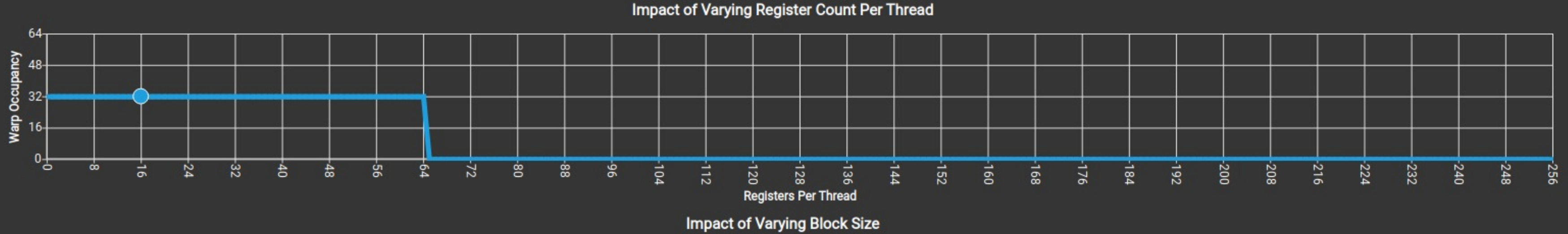
Occupancy

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.

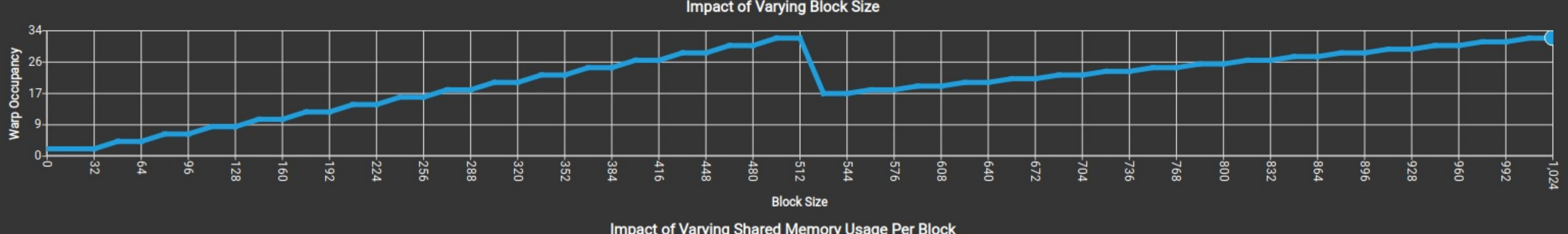
Theoretical Occupancy [%]	100	Block Limit Registers [block]	4
Theoretical Active Warps per SM [warp]	32	Block Limit Shared Mem [block]	2
Achieved Occupancy [%]	28.09	Block Limit Warps [block]	1
Achieved Active Warps Per SM [warp]	8.99	Block Limit SM [block]	16

Achieved Occupancy Est. Speedup: 71.91% The difference between calculated theoretical (100.0%) and measured achieved occupancy (28.1%) can be the result of warp scheduling overheads or workload imbalances during the kernel execution. Load imbalances can occur between warps within a block as well as across blocks of the same kernel. See the CUDA Best Practices Guide for more details on optimizing occupancy.

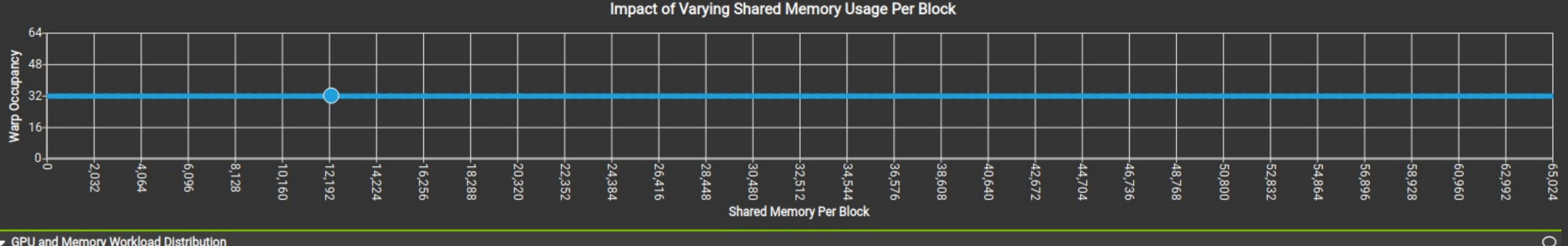
Impact of Varying Register Count Per Thread



Impact of Varying Block Size



Impact of Varying Shared Memory Usage Per Block



GPU and Memory Workload Distribution

Analysis of workload distribution in active cycles of SM, SMP, SMSP, L1 & L2 caches, and DRAM

Average SM Active Cycles [cycle]	516.67	Average L1 Active Cycles [cycle]	516.67
Average L2 Active Cycles [cycle]	537.38	Average SMSP Active Cycles [cycle]	261.95
Average DRAM Active Cycles [cycle]	943	Total SM Elapsed Cycles [cycle]	300,648
Total L1 Elapsed Cycles [cycle]	300,648	Total L2 Elapsed Cycles [cycle]	356,448
Total SMSP Elapsed Cycles [cycle]	1,202,592	Total DRAM Elapsed Cycles [cycle]	512,000

SMs Workload Imbalance Est. Speedup: 6.32% One or more SMs have a much lower number of active cycles than the average number of active cycles. Maximum instance value is 91.92% above the average, while the minimum instance value is 100.00% below the average.

L1 Slices Workload Imbalance Est. Speedup: 6.32% One or more L1 Slices have a much lower number of active cycles than the average number of active cycles. Maximum instance value is 91.92% above the average, while the minimum instance value is 100.00% below the average.

Workload Distribution

	Average	Min	Max	Sum
SM Active Cycles	516.67	0	6,395	20,667
SMSP Active Cycles	261.95	0	6,369	41,912
L1 Active Cycles	516.67	0	6,395	20,667
L2 Active Cycles	537.38	404	1,310	17,196
DRAM Active Cycles	943	856	1,064	7,544

Source Counters

Source metrics, including branch efficiency and sampled warp stall reasons. Warp Stall Sampling metrics are periodically sampled over the kernel runtime. They indicate when warps were stalled and couldn't be scheduled. See the documentation for a description of all stall reasons. Only focus on stalls if the schedulers fail to issue every cycle.

Branch Instructions [inst]	624	Branch Efficiency [%]	94.05
Branch Instructions Ratio [%]	0.11	Avg. Divergent Branches	0.12

Uncoalesced Shared Accesses Est. Speedup: 0.16% This kernel has uncoalesced shared accesses resulting in a total of 20 excessive wavefronts (2% of the total 862 wavefronts). Check the L1 Wavefronts Shared Excessive table for the primary source locations. The CUDA Best Practices Guide has an example on optimizing shared memory accesses.

L1 Wavefronts Shared Excessive

Location	Value	Value (%)
kernels.cu:220 (0x7d3c3f03a250 in reduce_argmin)	10	50
kernels.cu:220 (0x7d3c3f03a240 in reduce_argmin)	10	50
kernels.cu:228 (0x7d3c3f03a2c0 in reduce_argmin)	0	0
kernels.cu:228 (0x7d3c3f03a290 in reduce_argmin)	0	0
kernels.cu:219 (0x7d3c3f03a220 in reduce_argmin)	0	0

Follow the *rules outputs* to get guidance on how to navigate through the report and quickly discover performance bottlenecks in this kernel.
You could also disable individual sections to focus on selected performance aspects and make profiling faster.