	Result	Size	Time	Cycles	GPU	SM Frequency	Process	Attributes
Current	1790 - gemmSharedMem	(128, 128, 1)x(1024, 1, 1)	284.90 ms	256,406,156	0 - NVIDIA GeForce RTX 3060 Laptop GPU	900.00 Mhz	[105151] 03_shared_memory_gemm	

Summary | Details | Source | Context | Comments | Raw | Session

Compare | Tools | View | Export

GPU Speed Of Light Throughput

All

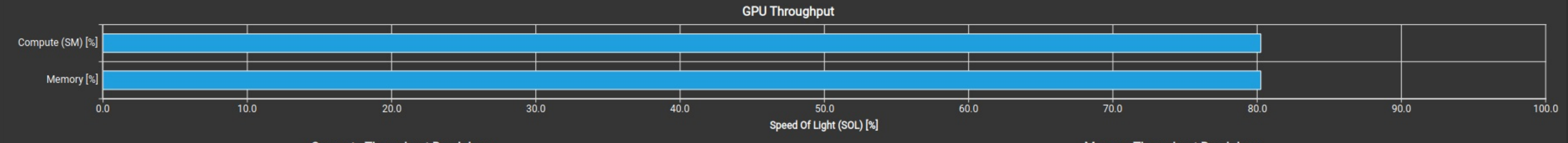
High-level overview of the throughput for compute and memory resources of the GPU. For each unit, the throughput reports the achieved percentage of utilization with respect to the theoretical maximum. Breakdowns show the throughput for each individual sub-metric of Compute and Memory to clearly identify 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 [%]	80.26	Duration [ms]	284.90
Memory Throughput [%]	80.26	Elapsed Cycles [cycle]	256406156
L1/TEX Cache Throughput [%]	80.40	SM Active Cycles [cycle]	256012466.40
L2 Cache Throughput [%]	13.59	SM Frequency [Mhz]	900.00
DRAM Throughput [%]	9.54	DRAM Frequency [Ghz]	6.99

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.

Roofline Analysis The ratio of peak float (fp32) to double (fp64) performance on this device is 64:1. The kernel achieved 7% 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



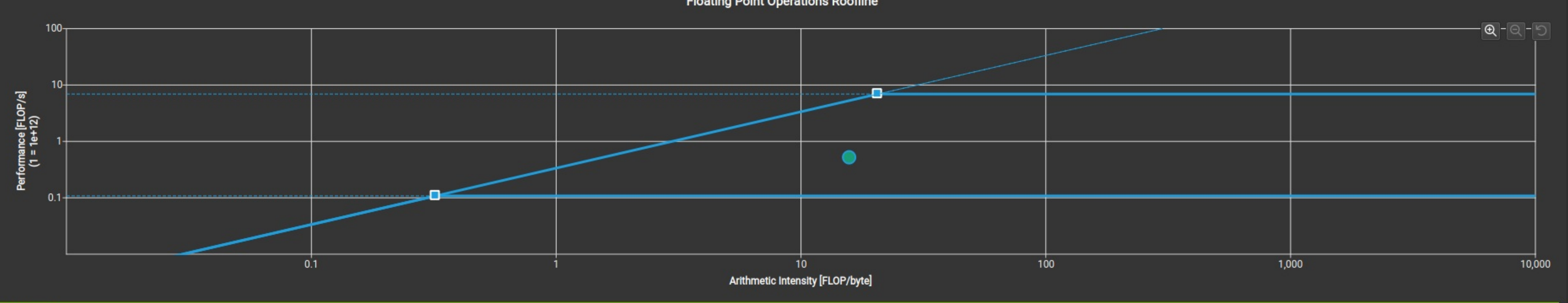
Compute Throughput Breakdown

SM: Inst Executed Pipe Lsu [%]	80.26
SM: Mio Inst Issued [%]	28.51
SM: Mio2rf Writeback Active [%]	22.25
SM: Issue Active [%]	19.47
SM: Inst Executed [%]	19.47
SM: Pipe Fma Cycles Active [%]	7.86
SM: Inst Executed Pipe Adu [%]	5.26
SM: Pipe Fmaheavy Cycles Active [%]	2.82
SM: Mio Pq Read Cycles Active [%]	2.62
SM: Mio Pq Write Cycles Active [%]	2.62
SM: Pipe Alu Cycles Active [%]	1.83
SM: Inst Executed Pipe Cbu Pred On Any [%]	0.01
SM: Inst Executed Pipe Uniform [%]	0.00
SM: Inst Executed Pipe Ipa [%]	0
SM: Inst Executed Pipe Tex [%]	0
SM: Inst Executed Pipe Xu [%]	0
SM: Instruction Throughput Internal Activity [%]	0
SM: Memory Throughput Internal Activity [%]	0
IDC: Request Cycles Active [%]	0
SM: Pipe Fp64 Cycles Active [%]	0
SM: Pipe Tensor Cycles Active [%]	0

Memory Throughput Breakdown

L1: Lsuin Requests [%]	80.26
L1: Data Pipe Lsu Wavefronts [%]	47.80
L1: Lsu Writeback Active [%]	44.49
L1: Data Bank Reads [%]	16.57
L2: T Sectors [%]	13.59
DRAM: Dram Sectors [%]	9.54
DRAM: Cycles Active [%]	9.49
L2: Lts2xbar Cycles Active [%]	9.08
L1: M Xbar2l1tex Read Sectors [%]	7.01
L2: Xbar2lts Cycles Active [%]	6.76
L1: M L1tex2xbar Req Cycles Active [%]	5.24
L2: D Sectors Fill Device [%]	4.73
L2: D Sectors [%]	4.59
L2: T Tag Requests [%]	3.41
GPU: Compute Memory Access Throughput Internal Activity [%]	3.41
L1: Data Bank Writes [%]	2.18
L1: Texin Sm2tex Req Cycles Active [%]	0.08
L2: D Sectors Fill Sysmem [%]	0.00
L1: F Wavefronts [%]	0
L1: Data Pipe Tex Wavefronts [%]	0
L1: Tex Writeback Active [%]	0
L2: D Atomic Input Cycles Active [%]	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 [ms]	2.05	# Pass Groups	2
Maximum Buffer Size [Mbytes]	16.31	Dropped Samples [sample]	7

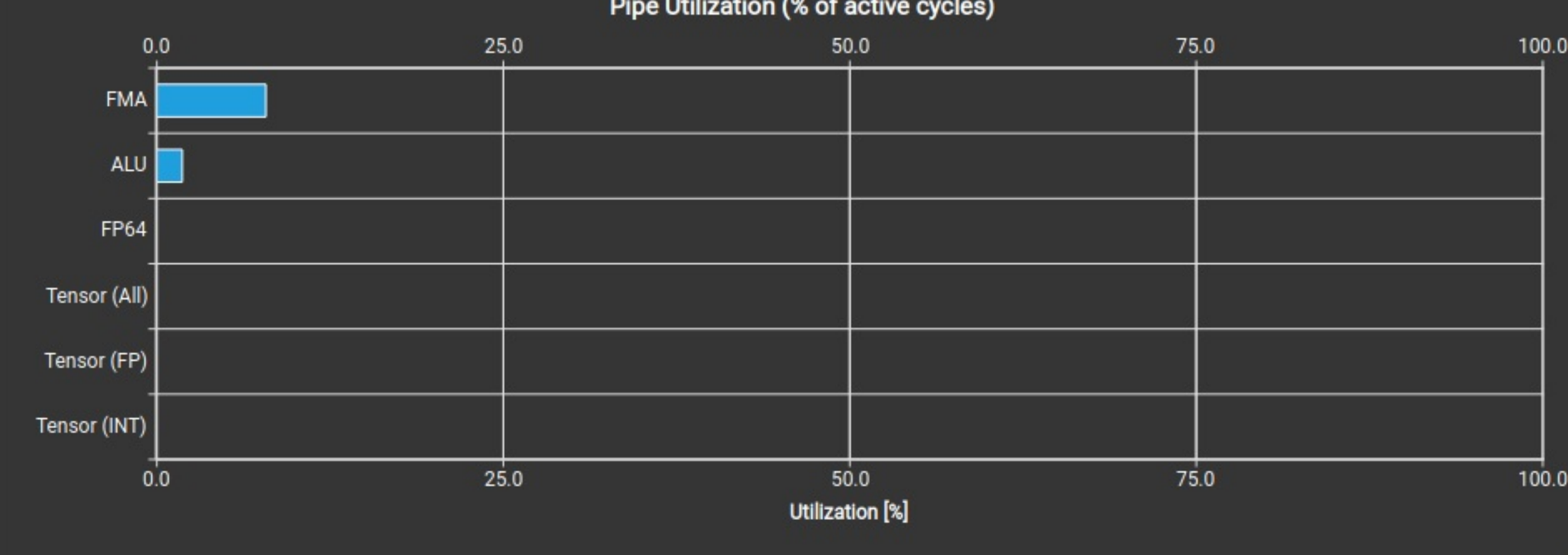
Compute Workload Analysis

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.

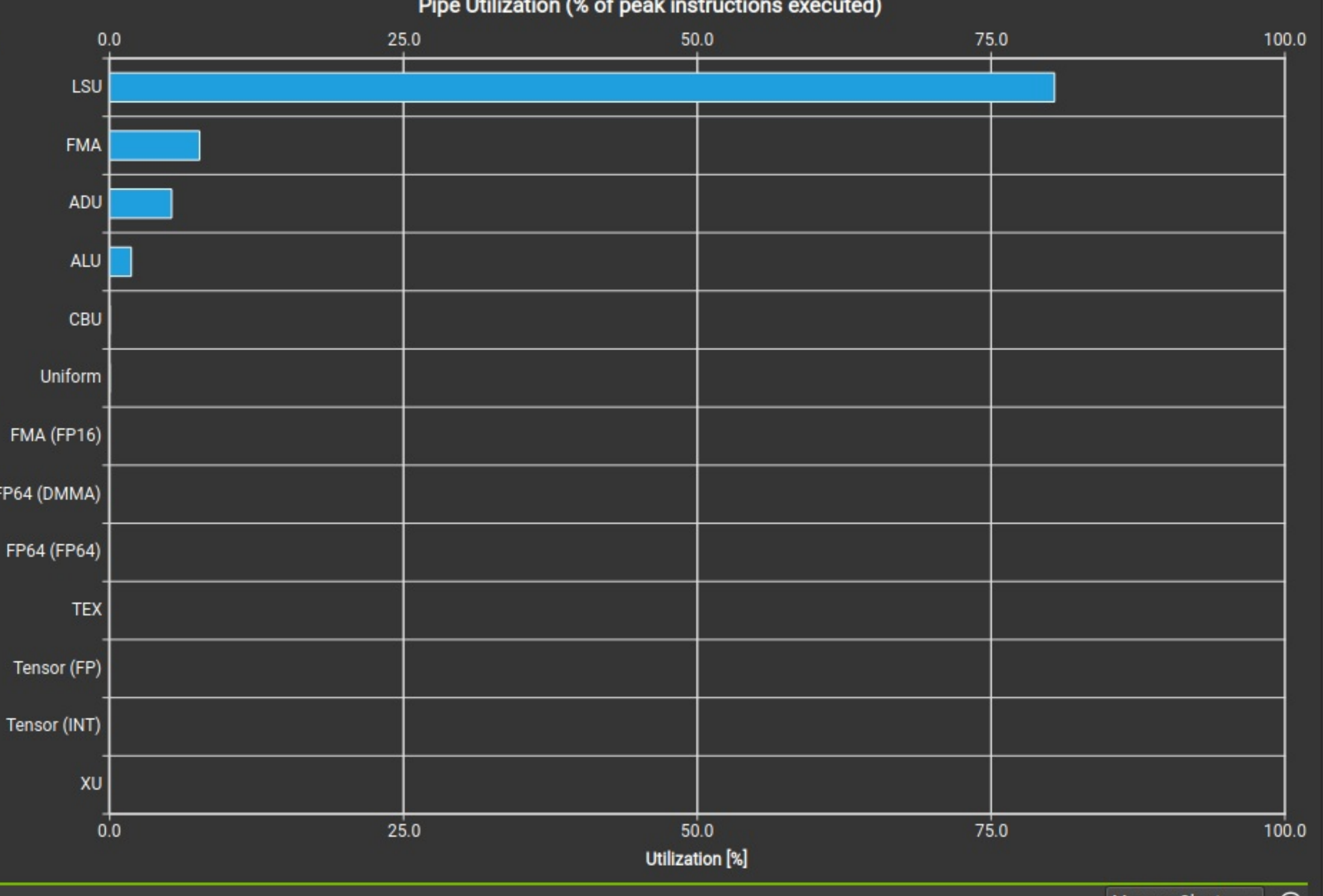
Executed Ipc Elapsed [inst/cycle]	0.78	SM Busy [%]	28.56
Executed Ipc Active [inst/cycle]	0.78	Issue Slots Busy [%]	19.50
Issued Ipc Active [inst/cycle]	0.78		

Low Utilization
Est. Local Speedup: 92.13% All compute pipelines are under-utilized. Either this workload 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

Memory Chart

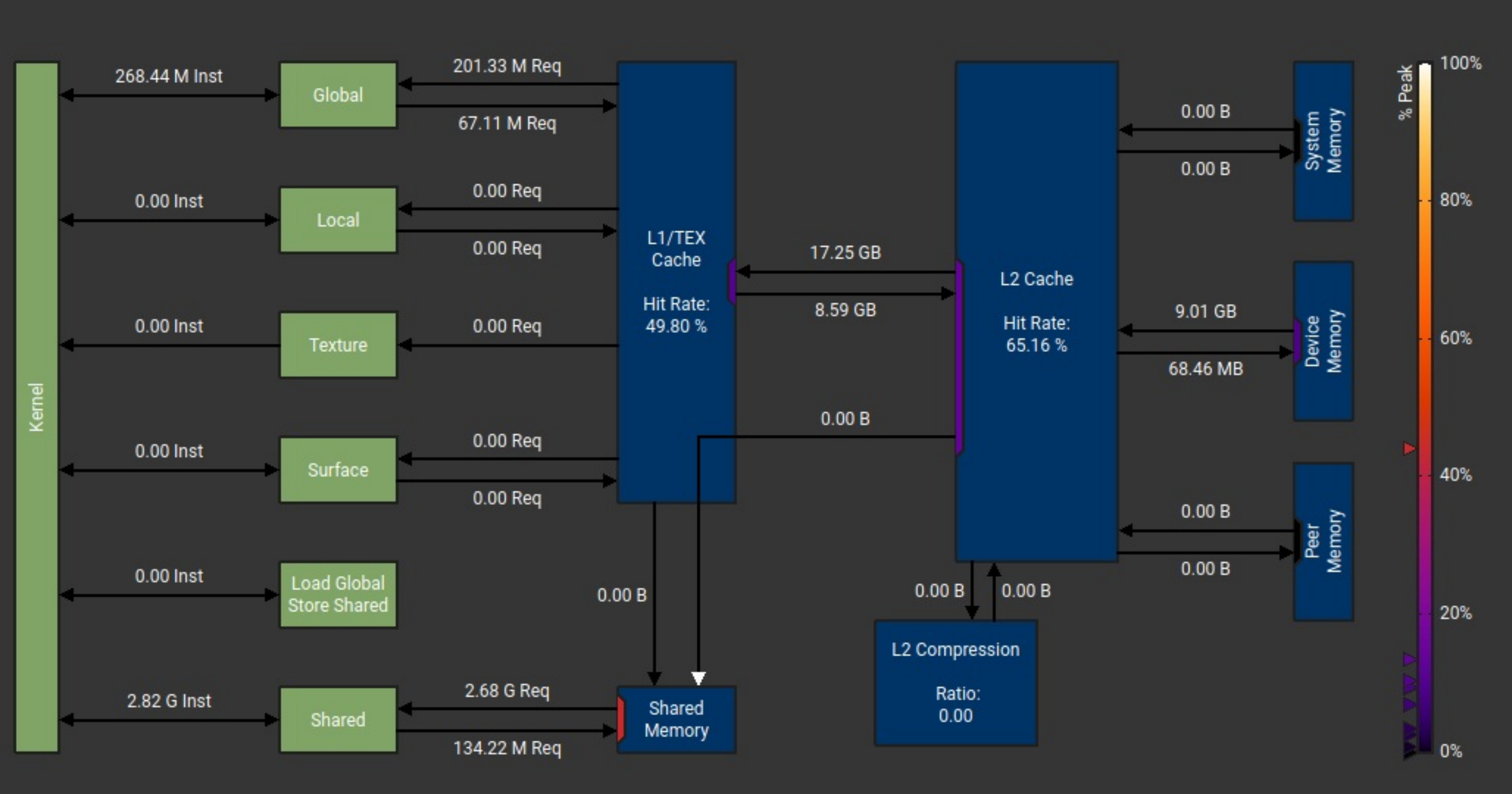
Detailed analysis of the memory resources of the GPU. Memory can become a limiting factor for the overall kernel performance when fully utilizing the involved hardware units (Mem Busy), exhausting the available communication bandwidth between those units (Max Bandwidth), or by reaching the maximum throughput 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]	31.87	Mem Busy [%]	47.80
L1/TEX Hit Rate [%]	49.80	Max Bandwidth [%]	80.26
L2 Hit Rate [%]	65.16	Mem Pipes Busy [%]	80.26
L2 Compression Success Rate [%]	0	L2 Compression Ratio	0

Shared Store Bank Conflicts
Est. Speedup: 16.25% The memory access pattern for shared stores might not be optimal and causes on average a 1.3 - way bank conflict across all 134217728 shared store requests.This results in 34025294 bank conflicts, which represent 20.21% of the overall 168334726 wavefronts for shared stores. Check the Source Counters section for uncoalesced shared stores.

Memory Chart

Show As: Transfer Size



Kernel
268.44 M Inst
Global
201.33 M Req
67.11 M Req
0.00 Inst
Local
0.00 Req
0.00 Req
0.00 Inst
Texture
0.00 Req
0.00 Inst
Surface
0.00 Req
0.00 Req
0.00 Inst
Load Global Store Shared
2.82 G Inst
Shared
2.68 G Req
134.22 M Req
L1/TEX Cache
Hit Rate: 49.80 %
Shared Memory
0.00 B
17.25 GB
8.59 GB
0.00 B
L2 Cache
Hit Rate: 65.16 %
L2 Compression
Ratio: 0.00
0.00 B
0.00 B
System Memory
0.00 B
0.00 B
Device Memory
9.01 GB
68.46 MB
Peer Memory
0.00 B
0.00 B
% Peak
100%
80%
60%
40%
20%
0%

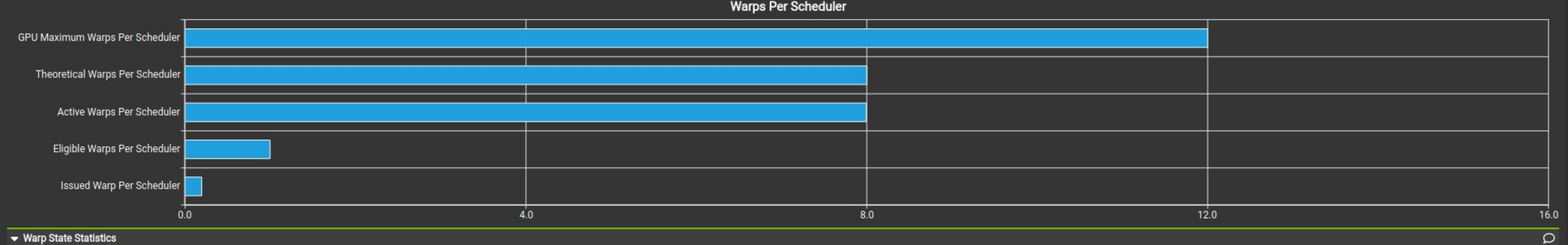
Scheduler Statistics

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]	7.99	No Eligible [%]	80.50
Eligible Warps Per Scheduler [warp]	1.00	One or More Eligible [%]	19.50
Issued Warp Per Scheduler	0.19		

Issue Slot Utilization
Est. Local Speedup: 19.74% Every scheduler is capable of issuing one instruction per cycle, but for this kernel each scheduler only issues an instruction every 5.1 cycles. This might leave hardware resources underutilized and may lead to less optimal performance. Out of the maximum of 12 warps per scheduler, this kernel allocates an average of 7.99 active warps per scheduler, but only an average of 1.00 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, avoid possible load imbalances due to highly different execution durations per warp. Reducing stalls indicated on the Warp State Statistics and Source Counters sections can help, too.

Warps Per Scheduler



Warp State Statistics

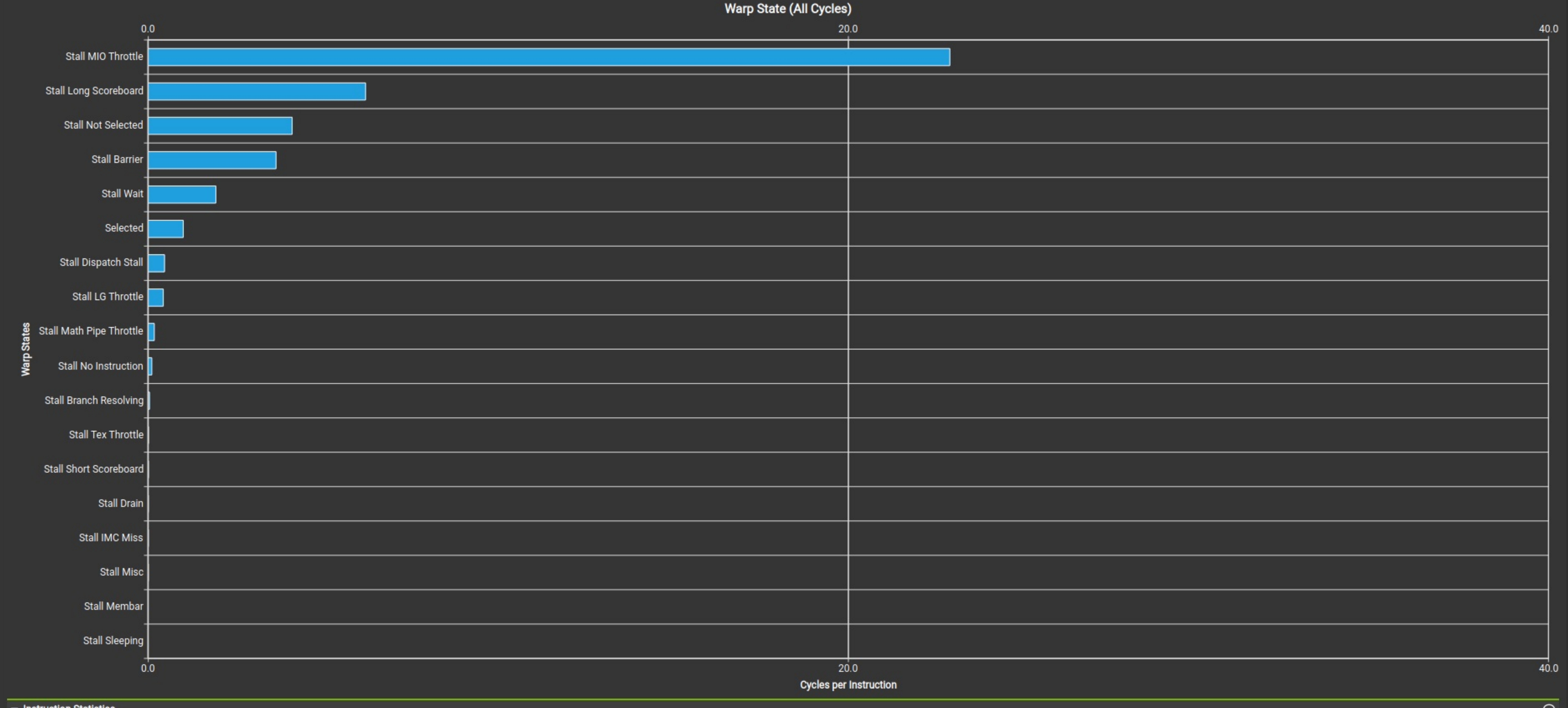
Analysis of the states in which all warps spent cycles during the kernel execution. The warp states describe a warp's readiness or inability to issue its next instruction. The warp cycles per instruction define the latency between two consecutive instructions. The higher the value, the more warp parallelism 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]	40.99	Avg. Active Threads Per Warp	32
Warp Cycles Per Executed Instruction [cycle]	40.99	Avg. Not Predicated Off Threads Per Warp	31.99

Mio Throttle Stalls
Est. Speedup: 19.74% On average, each warp of this kernel spends 22.9 cycles being stalled waiting for the MIO (memory input/output) instruction queue to be not full. This stall reason is high in cases of extreme utilization of the MIO pipelines, which include special math instructions, dynamic branches, as well as shared memory instructions. When caused by shared memory accesses, trying to use fewer but wider loads can reduce pipeline pressure. This stall type represents about 55.9% of the total average of 41.0 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)



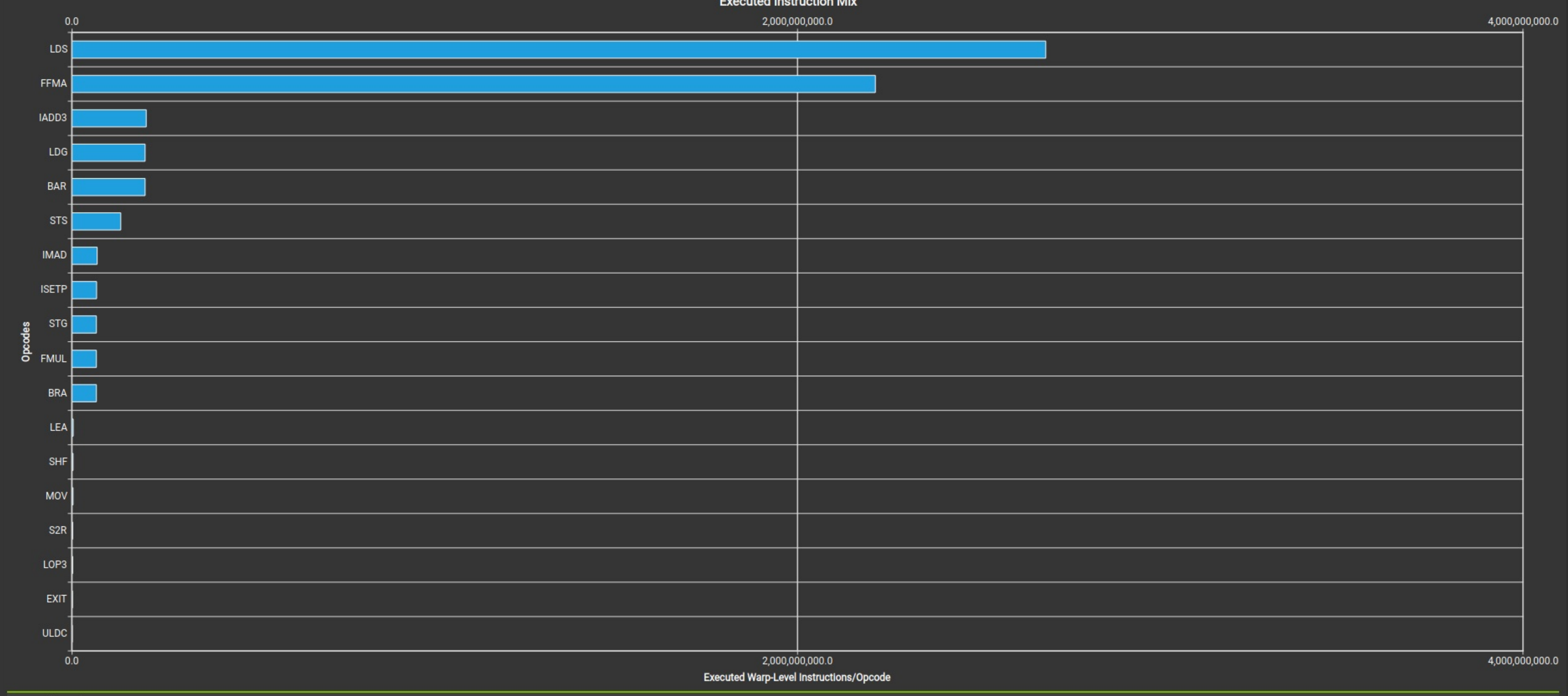
Warp States
Stall MIO Throttle
Stall Long Scoreboard
Stall Not Selected
Stall Barrier
Stall Wait
Selected
Stall Dispatch Stall
Stall LG Throttle
Stall Math Pipe Throttle
Stall No Instruction
Stall Branch Resolving
Stall Tex Throttle
Stall Short Scoreboard
Stall Drain
Stall IMC Miss
Stall Misc
Stall Membar
Stall Sleeping
0.0
20.0
40.0
Cycles per Instruction

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]	5991563264	Avg. Executed Instructions Per Scheduler [inst]	49929693.87
Issued Instructions [inst]	5991569157	Avg. Issued Instructions Per Scheduler [inst]	49929742.98

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	16384	Function Cache Configuration	CachePreferNone
Registers Per Thread [register/thread]	37	Static Shared Memory Per Block [Kbyte/block]	8.19
Block Size	1024	Dynamic Shared Memory Per Block [byte/block]	0
Threads [thread]	16777216	Driver Shared Memory Per Block [Kbyte/block]	1.02
Waves Per SM	546.13	Shared Memory Configuration Size [Kbyte]	16.38
Uses Green Context	0	# SMs [SM]	30

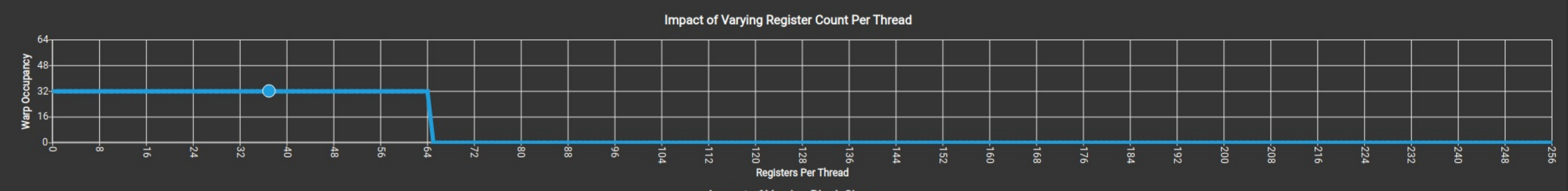
Occupancy

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.

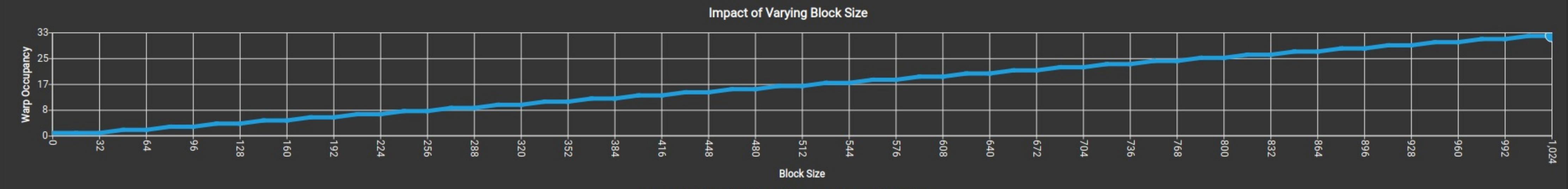
Theoretical Occupancy [%]	66.67	Block Limit Registers [block]	1
Theoretical Active Warps per SM [warp]	32	Block Limit Shared Mem [block]	1
Achieved Occupancy [%]	66.69	Block Limit Warps [block]	1
Achieved Active Warps Per SM [warp]	32.01	Block Limit SM [block]	16

Theoretical Occupancy
Est. Speedup: 19.74% The 8.00 theoretical warps per scheduler this kernel can issue according to its occupancy are below the hardware maximum of 12. 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 required amount of shared memory. This kernel's theoretical occupancy (66.7%) is limited by the number of warps within each block.

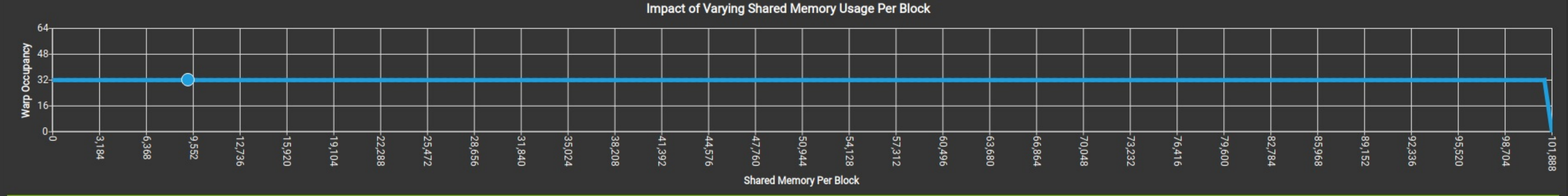
Impact of Varying Register Count Per Thread



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]	256012466.40	Average L1 Active Cycles [cycle]	256012466.40
Average L2 Active Cycles [cycle]	97243012.04	Average SMSP Active Cycles [cycle]	256051604.23
Average DRAM Active Cycles [cycle]	189136037.33	Total SM Elapsed Cycles [cycle]	7693974030
Total L1 Elapsed Cycles [cycle]	7693974030	Total L2 Elapsed Cycles [cycle]	5948622840
Total SMSP Elapsed Cycles [cycle]	30775896120	Total DRAM Elapsed Cycles [cycle]	11955366912

L2 Slices Workload Imbalance
Est. Speedup: 13.37% One or more L2 Slices have a much higher number of active cycles than the average number of active cycles. Maximum instance value is 34.08% above the average, while the minimum instance value is 3.85% below the average.

Workload Distribution

	Average	Min	Max	Sum
SM Active Cycles	256012466.40	255939506	256415429	7680373992
SMSP Active Cycles	256051604.23	255976309	256466442	30726192508
L1 Active Cycles	256012466.40	255939506	256415429	7680373992
L2 Active Cycles	97243012.04	93496276	147517569	2333832289
DRAM Active Cycles	189136037.33	189122256	189153424	1134816224

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]	68157440	Branch Efficiency [%]	100
Branch Instructions Ratio [%]	0.01	Avg. Divergent Branches	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.