

Current	Result	Size	Time	Cycles	GPU	SM Frequency	Prosees	Attributes
578-mvt_kernel1_wvs	256,1,1(704,1,1)	6.51 ms	10,968,330	0	NVIDIA GeForce RTX 5050 Laptop GPU	1.68 GHz	[8936] mvt_k1_wz_loading_A_and_y_ee	
Summary	Details	Source	Context	Comments	Raw	Session	Compare	Tools
							View	Export

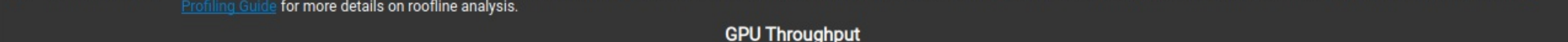
GPU Speed of Light Throughput

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-set 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 routine chart.

Compute (SM) Throughput [%]	22.10	Duration [ms]	6.51
Memory Throughput [%]	43.04	Elapsed Cycles [cycle]	10908330
L1/TEX Cache Throughput [%]	44.36	SM Active Cycles [cycle]	10067312.90
L2 Cache Throughput [%]	19.84	SM Frequency [GHz]	1.68
DRAM Throughput [%]	42.64	DRAM Frequency [GHz]	11.99

Latency Issue This workload exhibits low compute throughput and memory bandwidth utilization relative to the peak performance of this device. Achieved compute throughput and/or memory bandwidth below 60.0% of peak typically indicate latency issues. Look at [Launch Statistics](#) and [Warp State Statistics](#) for potential reasons.

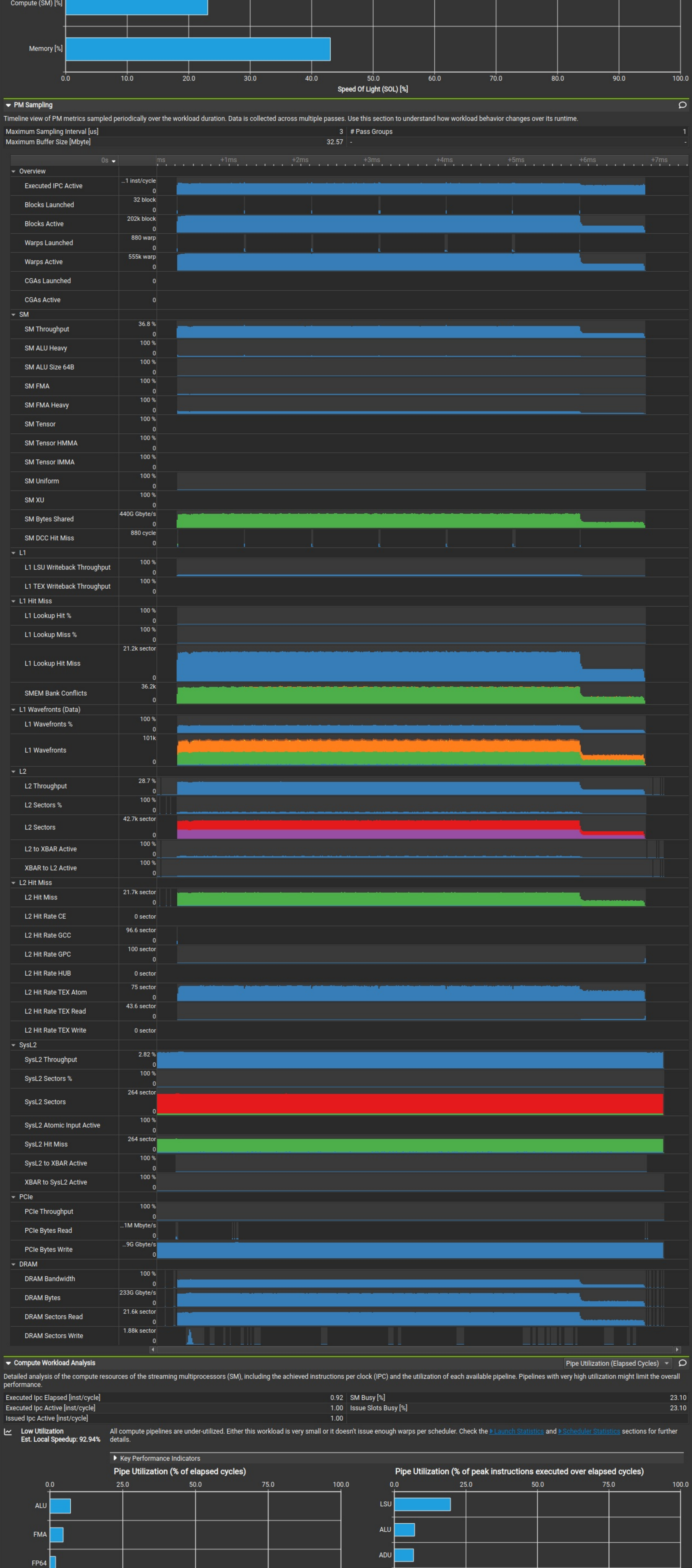
Key Performance Indicators The ratio of peak float (FP32) to double (FP64) performance on this device is 64.1. The workload achieved close to 1% of this device's FP32 peak performance and 0% of its FP64 peak performance. See the [Bottling Guide](#) for more details on routine analysis.



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 [us]	32.57	# Passes	1
Maximum Buffer Size [byte]			



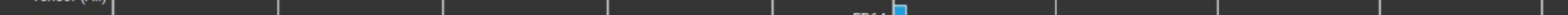
Compute Workload Analysis

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]	0.92	SM Busy [%]	23.10
Executed IPC Active [inst/cycle]	1.00	Issue Slots Busy [%]	23.10
Issued IPC Active [inst/cycle]			

Low Utilization 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 Runtime](#) sections for further details.

Est. Local Speedup: 92.94%	
----------------------------	--

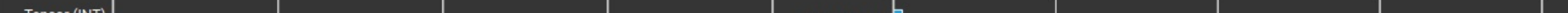


Memory Workload Analysis

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]	165.14	Mem Busy [%]	40.64
L1/TEX Hit Rate [%]	0.95	Max Bandwidth [%]	43.04
L2 Hit Rate [%]	2.25	Mem Pipes Busy [%]	19.59
L2 Compression Input Sectors [sector]	0	Local Memory Spilling Requests	0
L2 Compression Rate	0	Local Memory Spilling Request Overhead [%]	0
L2 Compression Success Rate [%]	0	L2 Persisting Size [Mbyte]	6.29

Shared Load Bank Conflicts The memory access pattern for shared loads might not be optimal and causes on average a 17.0-way bank conflict across all 4194304 shared load requests. This results in 5873477 bank conflicts, which represent 92.36% of the overall 7.7316100 wavefronts for shared loads. Check the [Launch Statistics](#) section for an uncached shared loads.

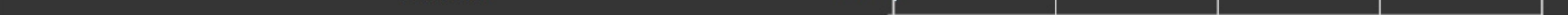


Scheduler Statistics

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 slots are skipped and no instruction is issued. Having many skipped issue slots indicates poor latency hiding.

Active Warps Per Scheduler [warp]	10.92	No Stalled [%]	74.96
Eligible Warps Per Scheduler [warp]	0.54	One or More Eligible [%]	25.04
Issued Warp Per Scheduler	0.25		

Issue Slot Utilization Every scheduler is capable of issuing one instruction per cycle, but for this workload each scheduler only issues an instruction every 4.0 cycles. This might leave hardware resources underutilized and may lead to less optimal performance. Out of the maximum of 12 warps per scheduler, this workload allocates an average of 10.92 active warps per scheduler, but only an average of 0.54 warps are eligible per cycle. Eligible warps are the subset of active warps that are ready to issue their next instruction. Every cycle with no eligible warps 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 [Launch Statistics](#) sections can help, too.



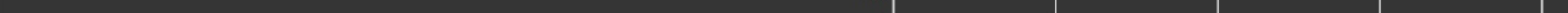
Warp State Statistics

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]	43.63	Avg. Active Threads Per Warp	32
Warp Cycles Per Executed Instruction [cycle]	43.63	Avg. Not Predicted Off Threads Per Warp	27.99

Long Scoreboard Stalls On average, each warp of this workload spends 27.2 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 for a scoreboard dependency on a 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 92.3% of the total average of 43.6 cycles between issuing two instructions.

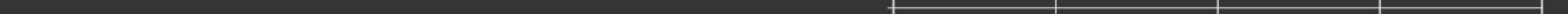
Est. Local Speedup: 56.96%	
----------------------------	--



Instruction Statistics

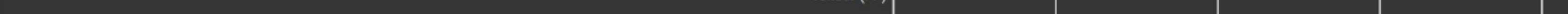
Statistics of the executed low-level assembly instructions (SASS). The instruction mix provides insights into the types and frequency of the executed instructions. A narrow mix of instruction types implies a dependency on few instruction pipelines, whereas others remain unused. Using multiple pipelines 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]	201582592	Avg. Executed Instructions Per Scheduler [inst]	251782.40
Issued Instructions [inst]	201582592	Avg. Issued Instructions Per Scheduler [inst]	251782.40



NVLink Topology

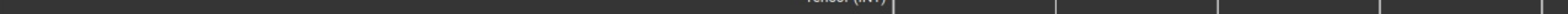
NVLink Topology diagram shows logical NVLink connections with transmit/receive throughput.



The system does not have any NVLink connections.

NVLink Tables

Detailed tables with properties for each NVLink.



NUMA Affinity

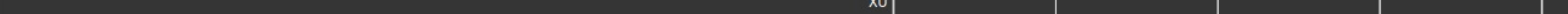
Non-uniform memory access (NUMA) affinities based on compute and memory distances for all GPUs.



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.

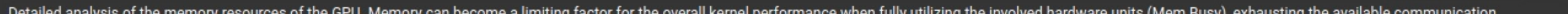
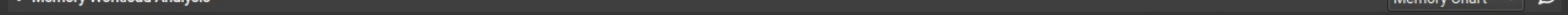
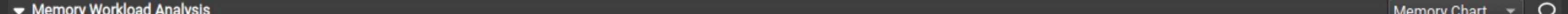
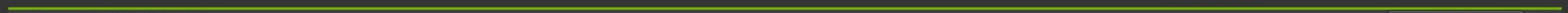
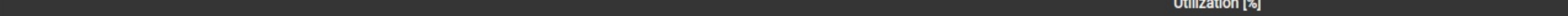
Grid Size	256	Function Cache Configuration	Cache/PreferNone
Cluster Size	0	Preferred Cluster Size	0
Registers Per Thread [register/thread]	40	Cluster Scheduling Policy	PolicySpread
Static Shared Memory Per Block [Kbyte/block]	33.28	Block Size	704
Dynamic Shared Memory Per Block [byte/block]	0	Threads [thread]	180224
Driver Shared Memory Per Block [Kbyte/block]	1.02	Waves Per SM	6.40
Shared Memory Configuration Size [Kbyte]	102.40	Use Green Context	0
Block Size	1024	# SAs SM	24
# TPCs	10	Idempotent TPC IDs	all



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 [%]	91.67	Block Limit Registers [block]	2
Theoretical Active Warps Per SM [warp]	44	Block Limit Shared Mem [block]	2
Achieved Occupancy [%]	90.87	Block Limit Waves Per SM [block]	2
Achieved Active Warps Per SM [warp]	43.62	Block Limit SM [block]	24
Cluster Occupancy [%]	0	Block Limit Barriers [block]	24
Max Active Clusters [cluster]	0	Max Cluster Size [block]	8
Overall GPU Occupancy [%]	0		



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]	10067312.90	Average L1 Active Cycles [cycle]	10067312.90
Average L2 Active Cycles [cycle]	10544454.88	Average SMSP Active Cycles [cycle]	10062704.74
Average DRAM Active Cycles [cycle]	33604328	Total SM Elapsed Cycles [cycle]	218158700
Achieved Active Warps Per SM [warp]	43.62	Total L2 Elapsed Cycles [cycle]	16940368
Total SMSP Elapsed Cycles [cycle]	872634800	Total DRAM Elapsed Cycles [cycle]	312303616

SMs Workload Imbalance One or more SMs have a much higher number of active cycles than the average number of active cycles. Additionally, other SMs have a much lower number of active cycles than the average number of active cycles. Maximum instance value is 7.63% above the average, while the minimum instance value is 6.84% below the average.

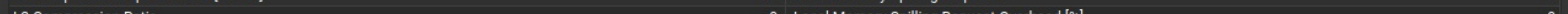
SMs Workload Imbalance	Est. Speedup: 7.04%
------------------------	---------------------

SMSPs Workload Imbalance One or more SMSPs have a much higher number of active cycles than the average number of active cycles. Additionally, other SMSPs have a much lower number of active cycles than the average number of active cycles. Maximum instance value is 7.66% above the average, while the minimum instance value is 6.62% below the average.

SMSPs Workload Imbalance	Est. Speedup: 7.07%
--------------------------	---------------------

L1 Slices Workload Imbalance One or more L1 Slices have a much higher number of active cycles than the average number of active cycles. Additionally, other L1 Slices have a much lower number of active cycles than the average number of active cycles. Maximum instance value is 7.63% above the average, while the minimum instance value is 6.84% below the average.

L1 Slices Workload Imbalance	Est. Speedup: 7.04%
------------------------------	---------------------



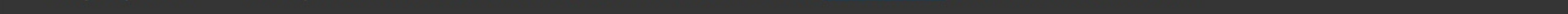
Source Metrics

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]	52198400	Branch Efficiency [%]	100
Branch Instructions Rate [%]	0.26	Avg. Divergent Branches [branch]	0

Uncached Shared Accesses This kernel has uncached shared accesses resulting in a total of 57202656 excessive wavefronts (94% of the total 79822248 wavefronts). Check the L1 Wavefronts Shared Excessive table for the primary source locations. The [Bottling Guide](#) is an example on optimizing shared memory accesses.

Uncached Shared Accesses	Est. Speedup: 67.89%
--------------------------	----------------------



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.