	Report	Result	Size	Time	Cycles	GPU	SM Frequency	Process	Attributes
Current	tcu_analysis_gm_write	705 - conv2d_kernel	(1, 392, 2)x(128, 1, 1)	1.35 ms	1,077,710	0 - Quadro T1000	796.41 Mhz	[1842142] resnet	
Baseline 1	direct_shared_analysis	699 - conv2d_kernel	(14, 14, 64)x(8, 8, 1)	3.19 ms	2,537,574	0 - Quadro T1000	794.72 Mhz	[1768587] resnet	

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.

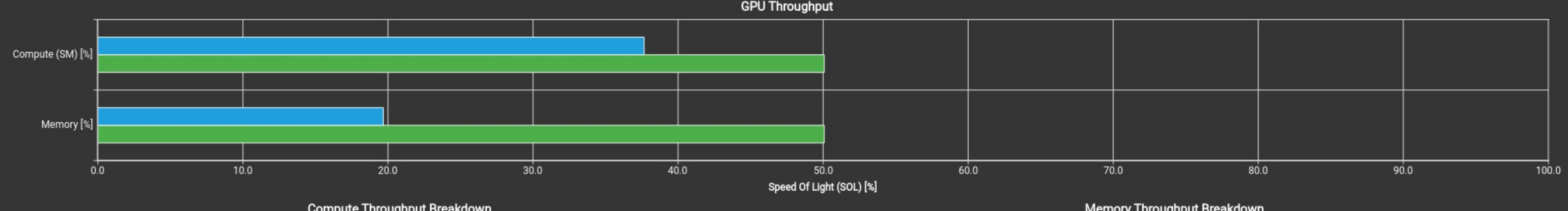
Compute (SM) Throughput [%]	37.66 (-24.79%)	Duration [ms]	1.35 (-57.62%)
Memory Throughput [%]	19.68 (-60.70%)	Elapsed Cycles [cycle]	1,077,710 (-57.53%)
L1/TEX Cache Throughput [%]	39.36 (-60.47%)	SM Active Cycles [cycle]	1,057,359.86 (-58.27%)
L2 Cache Throughput [%]	4.83 (-66.48%)	SM Frequency [Mhz]	796.41 (+0.21%)
DRAM Throughput [%]	2.66 (+170.69%)	DRAM Frequency [Ghz]	3.50 (+0.21%)

Latency Issue This kernel 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 Scheduler Statistics and Warp State Statistics for potential reasons.

Key Performance Indicators

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

GPU Throughput



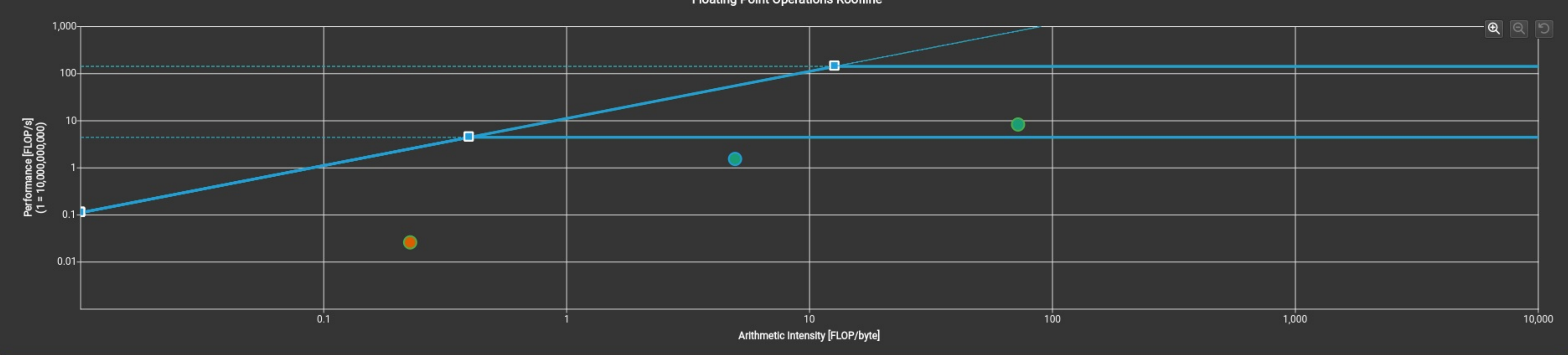
Compute Throughput Breakdown

SM: Inst Executed [%]	37.66 (-12.77%)
SM: Pipe Alu Cycles Active [%]	31.67 (-10.36%)
SM: Pipe Tensor Cycles Active [%]	26.60 (+inf%)
SM: Pipe Fma Cycles Active [%]	25.92 (+16.96%)
SM: Inst Executed Pipe Xu [%]	21.20 (+614.63%)
SM: Inst Executed Pipe Lsu [%]	15.79 (-68.48%)
SM: Mio Pq Write Cycles Active [%]	7.78 (+231.72%)
SM: Inst Executed Pipe Cbu Pred On Any [%]	7.42 (-3.95%)
SM: Mio Inst Issued [%]	6.14 (-64.45%)
SM: Mio2rf Writeback Active [%]	2.50 (-78.95%)
SM: Mio Pq Read Cycles Active [%]	2.13 (+3.03%)
SM: Inst Executed Pipe Adu [%]	0.96 (-24.78%)
SM: Inst Executed Pipe Uniform [%]	0.55 (-75.63%)
IDC: Request Cycles Active [%]	0 (+0.00%)
SM: Inst Executed Pipe Tex [%]	0 (+0.00%)
SM: Inst Executed Pipe Ipa [%]	0 (+0.00%)
SM: Inst Executed Pipe Fp16 [%]	0 (+0.00%)
SM: Pipe Fp64 Cycles Active [%]	0 (-100.00%)
SM: Pipe Shared Cycles Active [%]	0 (+0.00%)

Memory Throughput Breakdown

L1: Data Pipe Lsu Wavefronts [%]	19.68 (-60.47%)
L1: Lsuin Requests [%]	15.79 (-68.48%)
L1: Lsu Writeback Active [%]	8.38 (-66.58%)
L2: T Sectors [%]	4.83 (-66.48%)
L2: Lts2xbar Cycles Active [%]	3.78 (-72.47%)
L2: T Tag Requests [%]	3.42 (-60.07%)
L2: Xbar2lts Cycles Active [%]	3.41 (-56.69%)
DRAM: Cycles Active [%]	2.66 (+170.69%)
L1: Data Bank Reads [%]	2.54 (-59.27%)
L1: M Xbar2l1tex Read Sectors [%]	2.32 (-72.06%)
L1: M L1tex2xbar Req Cycles Active [%]	1.75 (-50.16%)
L2: D Sectors [%]	1.54 (-59.02%)
DRAM: Dram Sectors [%]	1.36 (+170.69%)
L1: Data Bank Writes [%]	1.28 (-35.04%)
L2: D Sectors Fill Device [%]	0.43 (+366.63%)
L1: Texin Sm2tex Req Cycles Active [%]	0.11 (-48.39%)
L1: F Wavefronts [%]	0.00 (+135.46%)
L1: Tex Writeback Active [%]	0 (+0.00%)
L2: D Atomic Input Cycles Active [%]	0 (+0.00%)
L2: D Sectors Fill Sysmem [%]	0 (+0.00%)
L1: Data Pipe Tex Wavefronts [%]	0 (+0.00%)

Floating Point Operations Roofline



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]	1.51 (-12.77%)	SM Busy [%]	38.39 (-11.23%)
Executed Ipc Active [inst/cycle]	1.54 (-11.23%)	Issue Slots Busy [%]	38.39 (-11.23%)
Issued Ipc Active [inst/cycle]	1.54 (-11.23%)		

Balanced No pipeline is over-utilized.

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]	2.98 (+171.26%)	Mem Busy [%]	19.68 (-60.47%)
L1/TEX Hit Rate [%]	90.76 (+259.82%)	Max Bandwidth [%]	15.79 (-68.48%)
L2 Hit Rate [%]	90.92 (-6.82%)	Mem Pipes Busy [%]	15.79 (-68.48%)

L1TEX Global Load Access Pattern The memory access pattern for global loads in L1TEX might not be optimal. On average, this kernel accesses 4.3 bytes per thread per memory request; but the address pattern, possibly caused by the stride between threads, results in 10.6 sectors per request, or 10.6*32 = 338.4 bytes of cache data transfers per request. The optimal thread address pattern for 4.3 byte accesses would result in 4.3*32 = 136.9 bytes of cache data transfers per request, to maximize L1TEX cache performance.Check the Source Counters section for uncoalesced global loads.

Key Performance Indicators

L2 Load Access Pattern The memory access pattern for loads from L1TEX to L2 is not optimal. The granularity of an L1TEX request to L2 is a 128 byte cache line. That is 4 consecutive 32-byte sectors per L2 request. However, this kernel only accesses an average of 1.5 sectors out of the possible 4 sectors per cache line. Check the Source Counters section for uncoalesced loads and try to minimize how many cache lines need to be accessed per memory request.

Key Performance Indicators

L2 Store Access Pattern The memory access pattern for stores from L1TEX to L2 is not optimal. The granularity of an L1TEX request to L2 is a 128 byte cache line. That is 4 consecutive 32-byte sectors per L2 request. However, this kernel only accesses an average of 1.1 sectors out of the possible 4 sectors per cache line. Check the Source Counters section for uncoalesced stores and try to minimize how many cache lines need to be accessed per memory request.

Key Performance Indicators

Local Memory Usage The optional metric smsp__sass_inst_executed_op_local_ld.sum could not be found. Collecting it as an additional metric could enable the rule to provide more guidance.

Shared Memory Conflicts The optional metric smsp__sass_inst_executed_op_shared_ld.sum could not be found. Collecting it as an additional metric could enable the rule to provide more guidance.

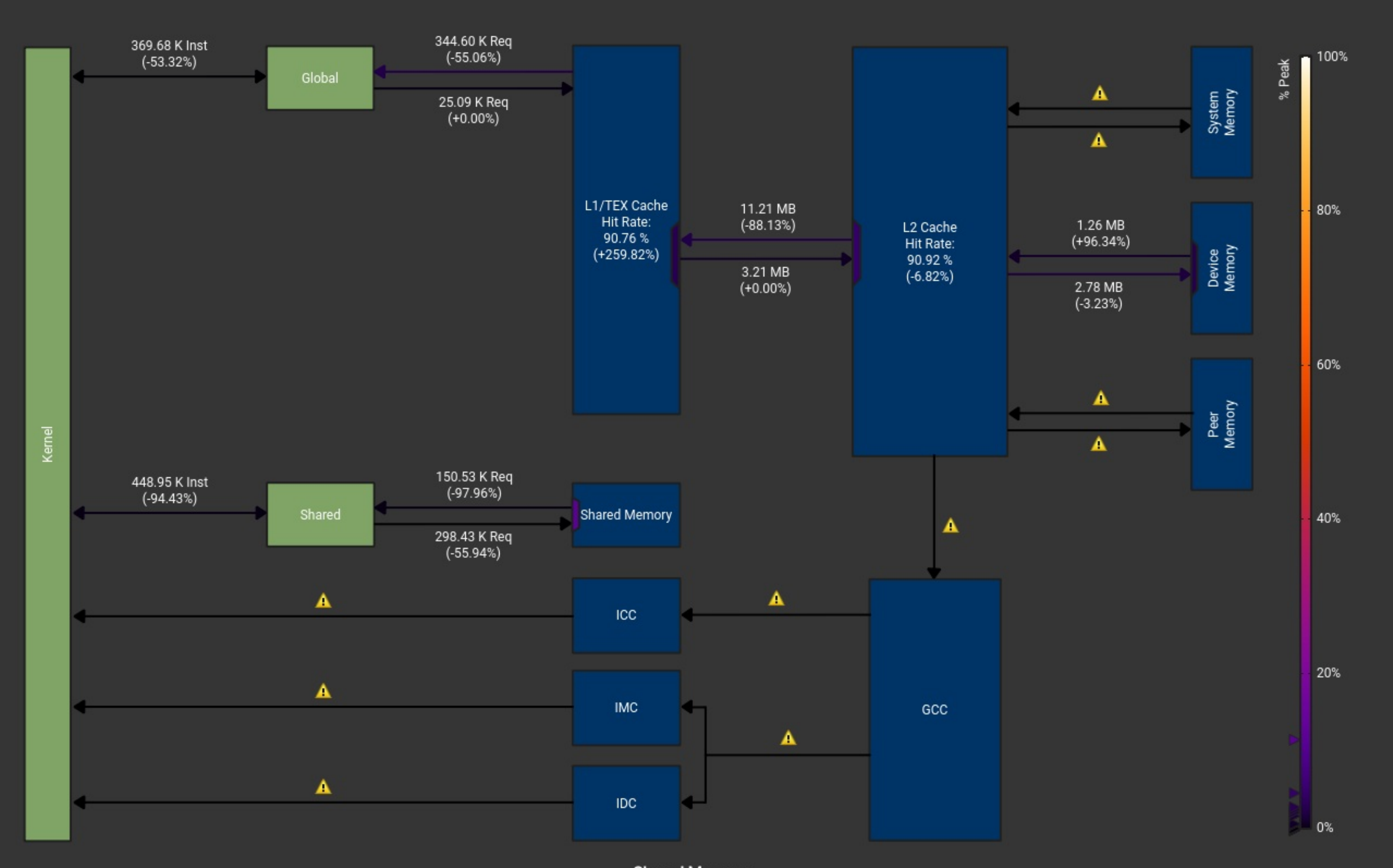
Shared Load Bank Conflicts Est. Local Speedup: 87.69% The memory access pattern for shared loads might not be optimal and causes on average a 1.6 - way bank conflict across all 125440 shared load requests.This results in 178768 bank conflicts, which represent 87.69% of the overall 203856 wavefronts for shared loads. Check the Source Counters section for uncoalesced shared loads.

Key Performance Indicators

Metric Name	Value	Guidance
l1tex__data_bank_conflicts_pipe_lsu_mem_shared_op_ld.sum	178768	Decrease bank conflicts for shared loads

Memory Chart

Values: Transfer Size Inactivity: Hidden



Kernel
369.68 K Inst
(-53.32%)
Global
344.60 K Req
(-55.06%)
25.09 K Req
(+0.00%)
L1/TEX Cache
Hit Rate:
90.76 %
(+259.82%)
11.21 MB
(-88.13%)
3.21 MB
(+0.00%)
L2 Cache
Hit Rate:
90.92 %
(-6.82%)
1.26 MB
(+96.34%)
2.78 MB
(-3.23%)
System Memory
Device Memory
Peer Memory
448.95 K Inst
(-94.43%)
Shared
150.53 K Req
(-97.96%)
298.43 K Req
(-55.94%)
Shared Memory
ICC
IMC
IDC
GCC
% Peak
100%
80%
60%
40%
20%
0%

Shared Memory

	Instructions	Requests	Wavefronts	% Peak	Bank Conflicts
Shared Load	25,088 (-99.66%)	25,088 (-99.66%)	203,856 (-98.62%)	1.35 (-96.75%)	178,768 (-97.58%)
Shared Load Matrix	125,440 (+inf%)	125,440 (+inf%)			
Shared Store	298,426 (-55.94%)	298,426 (-55.94%)	670,629 (-4.59%)	4.44 (+124.65%)	354,062 (+1,333.22%)
Shared Atomic	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)
Other	-	-	891,015 (+608.28%)	5.91 (+1,567.73%)	252,668 (+24,991.16%)
Total	448,954 (-94.43%)	448,954 (-94.43%)	1,765,500 (-88.67%)	11.70 (-73.33%)	785,498 (-89.40%)

L1/TEX Cache

	Instructions	Requests	Wavefronts	% Peak	Sectors	Sectors/Req	Hit Rate	Bytes	Sector Misses to L2	% Peak to L2	Returns to SM	% Peak to SM
Local Load	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	348,906 (-88.21%)	2.31 (-72.24%)	941,982 (-36.10%)	6.24 (+50.46%)
Global Load	344,596 (-55.06%)	344,596 (-55.06%)	813,580 (-31.37%)	5.39 (+61.60%)	3,644,383 (-5.96%)	10.58 (+109.25%)	91.19 (+248.61%)	116,620,256 (-5.96%)				
Surface Load	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)
Texture Load	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)		
Global Store	25,088 (+0.00%)	25,088 (+0.00%)	50,944 (-1.29%)	0.34 (+132.42%)	100,352 (+0.00%)	4 (+0.00%)	74.65 (+544.51%)	3,211,264 (+0.00%)	100,352 (+0.00%)	0.67 (+135.46%)	-	-
Local Store	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)			-	-
Surface Store	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	-	-
Global Reduction	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	-	-
Surface Reduction	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	-	-
Global Atomic ALU	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	see above	see above
Global Atomic CAS	0 (+0.00%)											
Surface Atomic ALU	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	see above	see above
Surface Atomic CAS	0 (+0.00%)											
Loads	344,596 (-55.06%)	344,596 (-55.06%)	813,580 (-31.37%)	5.39 (+61.60%)	3,644,383 (-5.96%)	10.58 (+109.25%)	91.19 (+248.61%)	116,620,256 (-5.96%)	348,906 (-88.21%)	2.31 (-72.24%)	941,982 (-36.10%)	6.24 (+50.46%)
Stores	25,088 (+0.00%)	25,088 (+0.00%)	50,944 (-1.29%)	0.34 (+132.42%)	100,352 (+0.00%)	4 (+0.00%)	74.65 (+544.51%)	3,211,264 (+0.00%)	100,352 (+0.00%)	0.67 (+135.46%)	-	-
Atomics & Reductions	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	-	-
Total	369,684 (-53.32%)	369,684 (-53.32%)	864,524 (-30.11%)	5.73 (+64.56%)	3,744,735 (-5.81%)	10.13 (+101.75%)	90.74 (+251.86%)	119,831,520 (-5.81%)	449,258 (-85.32%)	2.98 (-65.43%)	941,982 (-36.10%)	6.24 (+50.46%)

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	225,743 (-87.15%)	343,815 (-88.62%)	1.52 (-11.47%)	3.71 (-73.21%)	88.74 (-10.67%)	11,002,080 (-88.62%)	8,130,320,658.34 (-73.16%)	38,840 (+93.74%)	0 (+0.00%)	0 (+0.00%)
L1/TEX Store	89,130 (-10.52%)	100,352 (+0.00%)	1.13 (+11.75%)	1.08 (+135.46%)	100 (+0.00%)	3,211,264 (+0.00%)	2,373,060,915.63 (+135.96%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)
L1/TEX Atomic ALU	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)
L1/TEX Atomic CAS	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)
L1/TEX Reduction	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)	0 (+0.00%)
L1/TEX Total	313,916 (-83.00%)	446,679 (-85.80%)	1.42 (-16.47%)	4.82 (-66.57%)	91.35 (-8.04%)	14,293,728 (-85.80%)	10,562,783,768.44 (-66.49%)	38,840 (+93.81%)	0 (+0.00%)	0 (+0.00%)
GCC Total	0	0	0	0	0	0	0	0	0	0
ECC Total	-	0	-	0	-	0	0	0	-	-
GPU Total	315,956 (-83.00%)	447,815 (-85.77%)	1.42 (-16.26%)	4.83 (-66.48%)	91.29 (-8.10%)	14,330,080 (-85.77%)	10,589,647,181.23 (-66.41%)	38,845 (+93.91%)	0 (+0.00%)	0 (+0.00%)

Device Memory

	Sectors	% Peak	Bytes	Throughput
Load	39,355 (+96.34%)	0.83 (+362.32%)	1,259,360 (+96.34%)	930,642,262.58 (+363.29%)
Store	86,779 (-3.23%)	1.83 (+127.86%)	2,776,928 (-3.23%)	2,052,095,157.02 (+128.34%)
Total	126,134 (+14.96%)	2.66 (+170.69%)	4,036,288 (+14.96%)	2,982,737,419.60 (+171.26%)

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]	4.87 (-38.56%)	No Eligible [%]	61.68 (+8.71%)
Eligible Warps Per Scheduler [warp]	0.56 (-38.45%)	One or More Eligible [%]	38.32 (-11.42%)
Issued Warp Per Scheduler	0.38 (-11.42%)		

Issue Slot Utilization Every scheduler is capable of issuing one instruction per cycle, but for this kernel each scheduler only issues an instruction every 2.6 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 4.87 active warps per scheduler, but only an average of 0.56 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.

Issue Slot Utilization The 5.00 theoretical warps per scheduler this kernel can issue according to its occupancy are below the hardware maximum of 8. Use the Occupancy section to identify what limits this kernel's theoretical occupancy.

Key Performance Indicators

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]	12.71 (-30.64%)	Avg. Active Threads Per Warp	31.04 (+0.63%)
Warp Cycles Per Executed Instruction [cycle]	12.71 (-30.64%)	Avg. Not Predicated Off Threads Per Warp	26.77 (-1.16%)

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]	22,727,910 (-62.95%)	Avg. Executed Instructions Per Scheduler [inst]	405,855.54 (-62.95%)
Issued Instructions [inst]	22,729,821 (-62.95%)	Avg. Issued Instructions Per Scheduler [inst]	405,889.66 (-62.95%)

FP32/64 Instructions Floating-point instruction analysis. Apply

NVLink Topology

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

NVLink Tables

Detailed tables with properties for each NVLink.

Logical NVLink Properties

The system does not have any NVLink connections.

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	784 (-93.75%)	Registers Per Thread [register/thread]	40 (-14.89%)
Block Size	128 (+100.00%)	Static Shared Memory Per Block [Kbyte/block]	6.14 (+inf%)
Threads [thread]	100,352 (-87.50%)	Dynamic Shared Memory Per Block [Kbyte/block]	6.14 (+188.18%)
Waves Per SM	11.20 (-80.00%)	Driver Shared Memory Per Block [byte/block]	0 (+0.00%)
Function Cache Configuration	cudaFuncCachePreferNone (cudaFuncCachePreferNone)	Shared Memory Configuration Size [Kbyte]	65.54 (+0.00%)

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 [%]	62.50 (-37.50%)	Block Limit Registers [block]	12 (-40.00%)
Theoretical Active Warps per SM [warp]	20 (-37.50%)	Block Limit Shared Mem [block]	5 (-82.14%)
Achieved Occupancy [%]	60.96 (-38.47%)	Block Limit Warps [block]	8 (-50.00%)
Achieved Active Warps Per SM [warp]	19.51 (-38.47%)	Block Limit SM [block]	16 (+0.00%)

Occupancy Limiters This kernel's theoretical occupancy (62.5%) is limited by the required amount of shared memory See the CUDA Best Practices Guide for more details on optimizing occupancy.

Achieved Occupancy Analysis of the Achieved Occupancy Apply

Theoretical Occupancy Analysis of Theoretical Occupancy and its Limiters Apply

Source Counters

Sampling Data (All)

Source metrics, including branch efficiency and sampled warp stall reasons. Sampling Data 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]	1,461,882 (-80.41%)	Branch Efficiency [%]	96.69 (-0.81%)
Branch Instructions Ratio [%]	0.06 (-47.12%)	Avg. Divergent Branches [branches]	625.04 (-78.35%)

Uncoalesced Global Accesses Uncoalesced global access, expected 116032 sectors, got 589568 (5.08x) at PC @source:7f22e3e4a470:0x7f22e3e4a470@

Key Performance Indicators

Uncoalesced Global Accesses Uncoalesced global access, expected 116032 sectors, got 589568 (5.08x) at PC @source:7f22e3e4a600:0x7f22e3e4a600@

Key Performance Indicators

Uncoalesced Global Accesses Uncoalesced global access, expected 116032 sectors, got 589568 (5.08x) at PC @source:7f22e3e4a740:0x7f22e3e4a740@

Key Performance Indicators

Uncoalesced Global Accesses Uncoalesced global access, expected 112896 sectors, got 564480 (5.00x) at PC @source:7f22e3e4a880:0x7f22e3e4a880@

Key Performance Indicators

Uncoalesced Global Accesses Uncoalesced global access, expected 230272 sectors, got 470716 (2.04x) at PC @source:7f22e3e492f0:0x7f22e3e492f0@

Key Performance Indicators

Uncoalesced Global Accesses Uncoalesced global access, expected 227192 sectors, got 461640 (2.03x) at PC @source:7f22e3e49aa0:0x7f22e3e49aa0@

Key Performance Indicators

Uncoalesced Global Accesses See the CUDA Programming Guide for reducing uncoalesced device memory accesses.

Uncoalesced Shared Accesses Uncoalesced shared access, expected 12544 wavefronts, got 100352 (8.00x) at PC @source:7f22e3e4b630:0x7f22e3e4b630@

Key Performance Indicators

Uncoalesced Shared Accesses Uncoalesced shared access, expected 12544 wavefronts, got 100352 (8.00x) at PC @source:7f22e3e4bd60:0x7f22e3e4bd60@

Key Performance Indicators

Uncoalesced Shared Accesses Uncoalesced shared access, expected 62720 wavefronts, got 125440 (2.00x) at PC @source:7f22e3e4ae60:0x7f22e3e4ae60@

Key Performance Indicators

Uncoalesced Shared Accesses Uncoalesced shared access, expected 62720 wavefronts, got 125440 (2.00x) at PC @source:7f22e3e4ae70:0x7f22e3e4ae70@

Key Performance Indicators

Uncoalesced Shared Accesses Uncoalesced shared access, expected 62720 wavefronts, got 125440 (2.00x) at PC @source:7f22e3e4ae80:0x7f22e3e4ae80@

Key Performance Indicators

Uncoalesced Shared Accesses Uncoalesced shared access, expected 62720 wavefronts, got 125440 (2.00x) at PC @source:7f22e3e4ae90:0x7f22e3e4ae90@

Key Performance Indicators

Uncoalesced Shared Accesses Uncoalesced shared access, expected 62552 wavefronts, got 125104 (2.00x) at PC @source:7f22e3e49310:0x7f22e3e49310@

Key Performance Indicators

Uncoalesced Shared Accesses Uncoalesced shared access, expected 59388 wavefronts, got 118776 (2.00x) at PC @source:7f22e3e49ac0:0x7f22e3e49ac0@

Key Performance Indicators

Uncoalesced Shared Accesses Uncoalesced shared access, expected 31360 wavefronts, got 62720 (2.00x) at PC @source:7f22e3e4a490:0x7f22e3e4a490@

Key Performance Indicators

Uncoalesced Shared Accesses Uncoalesced shared access, expected 31360 wavefronts, got 62720 (2.00x) at PC @source:7f22e3e4a620:0x7f22e3e4a620@

Key Performance Indicators

Uncoalesced Shared Accesses See the CUDA Best Practices Guide for an example on optimizing shared memory accesses.

Sampling Data (All)

Location	Value	Value (%)
0x7f22e3e4aef0 in conv2d_kernel	1,188	11
0x7f22e3e4aec0 in conv2d_kernel	925	9
0x7f22e3e4ada0 in conv2d_kernel	832	8
0x7f22e3e49300 in conv2d_kernel	481	5
0x7f22e3e49ab0 in conv2d_kernel	317	3

To customize your report even further, you might want to learn about custom sections and writing your own rules.
You might also want to consider adding individual metrics.