Session: 1 - 2313 - ugemm_128x12 | Add Baseline | Apply Rules | Occupancy Calculator

	Kernel	Time	Cycles	Regs	GPU	SM Frequency	CC	Process
Current	2313 - ugemm_128x128x8_kernel_nanoxnl (64, 86, 1)x(256, 1, 1)	121.43 ms(cmd 128,320,707	128	8 - NVIDIA A800 80GB PCIe	1.06 cycle/nsecond	8.8 [1075683]	julia

GPU Speed Of Light Throughput

Breakdown of the throughput for the 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 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 [%]	82.24	Duration [nsecond]	121.43
Memory Throughput [%]	27.83	Elapsed Cycles [cycle]	128320707
L1/TEX Cache Throughput [%]	28.37	SM Active Cycles [cycle]	127722651.14
L2 Cache Throughput [%]	10.01	IM Frequency [cycle/nsecond]	1.04
DRAM Throughput [%]	3.74	DRAM Frequency [cycle/nsecond]	1.97

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 2:1. The kernel achieved 16% 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

Speed Of Light (SOL) [%]

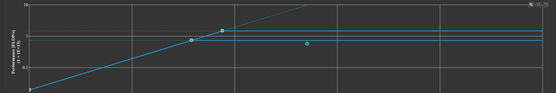
Compute Throughput Breakdown

SM: Instn Active [%]	82.24
SM: Inst Executed [%]	82.24
SM: Pipe Alls Cycles Active [%]	79.45
SM: Pipe Fma Cycles Active [%]	77.80
SM: Inst Executed Pipe Lsu [%]	34.56
SM: Mio2rf Writeback Active [%]	10.82
SM: Mio Inst Issued [%]	6.93
SM: Pipe Fma Cycles Active [%]	4.49
SM: Mio Pq Read Cycles Active [%]	2.44
SM: Inst Executed Pipe Adu [%]	0.61
SM: Inst Executed Pipe Cbu Pred On Any [%]	0.61
SM: Inst Executed Pipe Uniform [%]	0.00
SM: Pipe Shared Cycles Active [%]	0.00
SM: Pipe Shared Cycles Active [%]	0.00
SM: Inst Executed Pipe Ipa [%]	0.00
SM: Inst Executed Pipe Ipa [%]	0.00
SM: Inst Executed Pipe Fp16 [%]	0
SM: Pipe Fp64 Cycles Active [%]	0
SM: Inst Executed Pipe Tu [%]	0
IDC: Request Cycles Active [%]	0

Memory Throughput Breakdown

L1: Data Pipe Lsu Wavefronts [%]	27.83
L1: Data Pipe Lsd Wavefronts [%]	21.68
L1: Lsuin Requests [%]	13.56
L1: Tag Requests [%]	10.01
L2: Slm2Gc Cycles Active [%]	8.42
L1: Data Bank Reads [%]	5.25
L2: L1Gcbar Cycles Active [%]	4.39
L1: Data Bank Writes [%]	3.77
L2: L1miscInst Busy Cycles Active [%]	3.48
L2: D Sectors [%]	3.56
L1: DRAM Cycles Active [%]	3.74
L1: M Sbac2Gcm Read Sectors [%]	4.59
L1: M Sbac Data Sectors [%]	4.27
L2: D Sectors Fill Device [%]	4.11
L2: D Sectors Fill Sysmem [%]	3.47
L1: Lsuout Requests Active [%]	3.47
L1: F Wavefronts [%]	0.06
L2: D Sectors Fill Sysmem [%]	0.06
L1: Data Pipe Tex Wavefronts [%]	0
L2: D Atomic Input Cycles Active [%]	0
L1: Tex Writeback Active [%]	0

Floating Point Operations Roofline



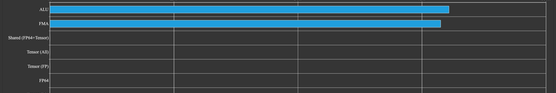
Compute Workload Analysis

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.

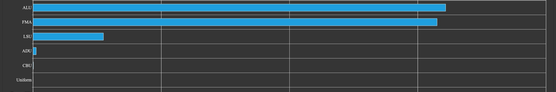
Executed Ipc Elapsed [inst/cycle]	3.29	IM Busy [%]	83.27
Executed Ipc Active [inst/cycle]	3.33	Issue Slots Busy [%]	83.27
Issued Ipc Active [inst/cycle]	3.33		

Very High Utilization ALU is the highest-utilized pipeline (80.4%) based on active cycles, taking into account the rates of its different instructions. It executes integer and logic operations. The pipeline is over-utilized and likely a performance bottleneck. Based on the number of executed instructions, the highest utilized pipeline (80.4%) is ALU. It executes integer and logic operations. Comparing the two, the overall pipeline utilization appears to be caused by frequent, low-latency instructions. See the Kernel Profiling Guide or hover over the pipeline name to understand the workloads handled by each pipeline. The Instruction Statistics section provides a breakdown of instructions issued.

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/second]	11.09	Mem Busy [%]	27.83
L1/TEX Hit Rate [%]	0.14	Max Bandwidth [%]	21.68
L2 Hit Rate [%]	66.74	Mem Pipes Busy [%]	13.56
L2 Compression Success Rate [%]	0	L2 Compression Ratio	0

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 1.8 sectors out of the possible 4 sectors per cache line. Check the L2 Cache Locality section for uncoalesced loads and try to minimize how many cache lines need to be accessed per memory request.

Memory Chart

Shared Memory

	Instructions	Requests	Wavefronts	% Peak	Bank Conflicts
Shared Load	1344788720	1344788720			
Shared Load Matrix	0	0	0		
Shared Store	2103765568	2103765568	3412916428	2.48	3132323
Shared Atomics Global Load	0	0	0	0	0
Shared Atomic	0	0	0	0	0
Other			5365745	0.04	0
Total	3448554288	3448554288	3418682173	2.51	3132323

L1/TEX Cache

	Instructions	Requests	Wavefronts	% Peak	Sectors	Sectors/Req	Hit Rate	Sector Misses to L2 % Peak to L2	Returns to SM	% Peak to SM
Global Load	335544320	335544320	335571085	2.40	1342684018	3.99	0.01	42.95035976	335541707	2.40
Global Load To Shared Byte (access)	0	0	0	0	0	0	0	0	0	0
Global Load To Shared Byte (bypass)	0	0	0	0	0	0	0	0	0	0
Global Store	0	0	0	0	0	0	0	0	0	0
Texture Load	0	0	0	0	0	0	0	0	0	0
Global Store	2621440	2621440	2621440	0.02	10485760	4	0	33554432G	10485760	0.08
Local Store	0	0	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
Texture Reduction	0	0	0	0	0	0	0	0	0	0
Surface Atomic ALU	0	0	0	0	0	0	0	0	0	0
Global Atomic CAS	0	0	0	0	0	0	0	0	0	0
Surface Atomic ALU	0	0	0	0	0	0	0	0	0	0
Surface Atomic CAS	0	0	0	0	0	0	0	0	0	0
Loads	335544320	335544320	335571085	2.40	1342684018	3.99	0.01	42.95035976	1342021107	4.80
Stores	2621440	2621440	2621440	0.02	10485760	4	0	33554432G	10485760	0.08
Atomics & Reductions	0	0	0	0	0	0	0	0	0	0
Total	338165760	338165760	338192525	2.42	1353169778	3.99	0.01	42.86584096	1352506867	4.83

L2 Cache

	Requests	Sectors	Sectors/Req	% Peak	Hit Rate	Bytes	Throughput	Lost Misses to Disc	Lost Misses to Sysmem	Misses to Fbp
L1/TEX Load	83875840	1342021479	1.00	6.77	83.74	42946693728	355669495258.98	49106350	0	0
L1/TEX Store	2621440	10485760	4	0.07	100	335544320	2782169563.53	0	0	0
L1/TEX Atomic ALU	0	0	0	0	0	0	0	0	0	0
L1/TEX Atomic CAS	0	0	0	0	0	0	0	0	0	0
L1/TEX Reduction	0	0	0	0	0	0	0	0	0	0
L1/TEX Total	86497280	1352507239	1.81	6.85	44.19	43289240960	358669058821.8	48948043	0	0
L2 Global Load	0	0	0	0	0	0	0	0	0	0
L2 Fabric Total	14971962	479111708	1.85	3.81	64.67	15499562656	132343135230.37	28519934	0	0
GPU Total	99469242	1831618947	1.85	4.49	48.22	47788803616	408041058051.40	48361104	0	0

L2 Cache Eviction Policies

	First	Hit Rate	Last	Hit Rate	Normal	Hit Rate	Normal Demote	Hit Rate
L1/TEX Load	0	0	0	0	1336438056	43.83	0	0
L1/TEX Store	0	0	0	0	10485760	100	0	0
L1/TEX Atomic	0	0	0	0	0	0	0	0
L1/TEX Total	0	0	0	0	1346923816	43.19	0	0
L2 Fabric Total	0	0	0	0	28951984	29.95	0	0
GPU Total	0	0	0	0	1375875800	43.00	0	0

Device Memory

	Sectors	% Peak	Bytes	Throughput
Load	41117500	0.07	1315760000	10834277595.32
Store	10398956	0.18	332767392	2740097775.89
Total	51516456	5.74	1648527392	13574375371.21

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]	3.97	No Eligible [%]	16.73
Eligible Warps Per Scheduler [warp]	2.08	One or More Eligible [%]	83.27
Issued Warps Per Scheduler	0.83		

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 scheduler fails to issue every cycle. When all warps in the warp pool are stalled, no instruction can be issued. Longer descriptions of the individual states to focus on are shown in the tooltip when hovering over the state's name. These descriptions also include ways to reduce the time spent in each state. Hover over the bars to see more detailed descriptions of the individual states.

Warp Cycles Per Issued Instruction [cycle]	4.77	Avg. Active Threads Per Warp
Warp Cycles Per Executed Instruction [cycle]	4.77	Avg. Not Predicated Off Threads Per Warp

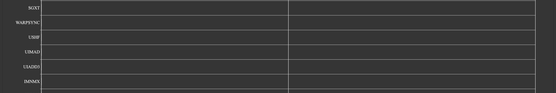
Instruction Statistics

Statistics of the executed low-level assembly instructions (SASS). The instruction mix provides insight into the types and frequency of the executed instructions. A uniform mix of instruction types implies a dependency on few instruction pipelines, while others remain unused. Using multiple pipelines concurrently allows for better instruction throughput and hardware utilization. Note that 'Instructions/Opcode' and 'Executed Instructions' are measured differently, see their tooltips for more details.

Executed Instructions [inst]	3896279200	Avg. Executed Instructions Per Scheduler [inst]	1969458276
Issued Instructions [inst]	4594264064	Avg. Issued Instructions Per Scheduler [inst]	1863487528.24

FP32/64 Instructions This kernel executes 0 fused and 2147483648 non-fused FP32 instructions. By converting pairs of non-fused instructions to their fused equivalent, the achieved FP32 performance could be increased by up to 50% (relative to its current performance). Check the Source page to identify where this kernel executes FP32 instructions.

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.

		Configuration	
Grid Size	128	Block Size	256
Registers Per Thread [register/thread]	128	Static Shared Memory Per Block [Kbyte/block]	24.58
Block Size	256	Dynamic Shared Memory Per Block [Kbyte/block]	0
Threads [thread]	1310720	Driver Shared Memory Per Block [Kbyte/block]	1.02
Warps Per SM	23.70	Shared Memory Configuration Size [Kbyte]	102.40

Occupancy

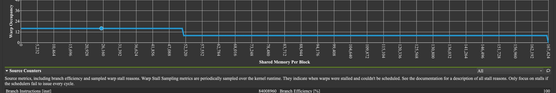
Occupancy is the ratio of the number of active warps per multiprocessor to the maximum number of possible active warps. Another way to view occupancy is the percentage of the hardware's ability to process warps that is actively in use. Higher occupancy does not always result in higher performance, however, 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 [%]	37	Block Limit Registers [block]	2
Theoretical Active Warps per SM [warp]	18	Block Limit Shared Mem [block]	4
Achieved Occupancy [%]	24.81	Block Limit Warps [block]	8
Achieved Active Warps Per SM [warp]	11.91	Block Limit SM [block]	32

Occupancy Limiters This kernel's theoretical occupancy (37.5 [%]) is limited by the number of required registers. See the NVIDIA 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

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 could not be scheduled. See the documentation for a description of all stall reasons. Only focus on stalls if the scheduler fail to issue every cycle.

Branch Instructions [inst]	16039560	Branch Efficiency [%]	100
Branch Instructions Ratio [%]	0.00	Avg. Divergent Branches	

Uncoalesced Shared Access This kernel has uncoalesced shared accesses resulting in a total of 262140 excessive wavefronts (0% of the total 8027967200 wavefronts). Check the L1 Wavefronts Shared Excessive table for the primary source locations. The L1/TEX Best Practices Guide has an example on optimizing shared memory accesses.

L1 Wavefronts Shared Excessive

Location	Value	Value (%)
0x7fb7c7d7f0f0 in ugemm_128x128x8_kernel_nanoxnl +	163,840	
0x7fb7c7d7a7a0 in ugemm_128x128x8_kernel_nanoxnl +	163,840	
0x7fb7c7d6fc60 in ugemm_128x128x8_kernel_nanoxnl +	163,840	
0x7fb7c7d1d0c0 in ugemm_128x128x8_kernel_nanoxnl +	163,840	
0x7fb7c7d7a7a0 in ugemm_128x128x8_kernel_nanoxnl +	163,840	

Warp Stall Sampling (All Cycles)

Location	Value	Value (%)
0x7fb7c7d6fc60 in ugemm_128x128x8_kernel_nanoxnl +	159,600	
0x7fb7c7d6fc70 in ugemm_128x128x8_kernel_nanoxnl +	135,473	
0x7fb7c7d6fc80 in ugemm_128x128x8_kernel_nanoxnl +	81,441	
0x7fb7c7d6fed0 in ugemm_128x128x8_kernel_nanoxnl +	78,125	
0x7fb7c7d6fc10 in ugemm_128x128x8_kernel_nanoxnl +	74,606	

Most Instructions Executed

Location	Value	Value (%)
0x7fb7c7d6d7d0 in ugemm_128x128x8_kernel_nanoxnl +	42,902,080	
0x7fb7c7d6d7e0 in ugemm_128x128x8_kernel_nanoxnl +	42,902,080	
0x7fb7c7d6d7f0 in ugemm_128x128x8_kernel_nanoxnl +	42,902,080	
0x7fb7c7d6d800 in ugemm_128x128x8_kernel_nanoxnl +	42,902,080	
0x7fb7c7d6d810 in ugemm_128x128x8_kernel_nanoxnl +	42,902,080	

Follow the *italic* output to get guidance on how to navigate through the report and quickly discover performance bottlenecks in this kernel.
You could also disable *inline mode* to focus on selected performance aspects and make profiling faster.