

CS553 Cloud Computing

Manual

Shruti Gupta (A20381966)

Sagar Mane (A20379756)

Test Bed Environment:

Chameleon KVM virtual machine m1.medium (2 virtual processors with 4GB RAM and 40GB disk).

Require Tools:

This benchmark requires gcc compiler and nvcc compiler.

Run all benchmark:

Command	Used
Chmod +x start.sh	Make file executable. Start.sh contains code for compiling and automatically running all benchmark file.
./start.sh	Run start.sh file. This will run all test of benchmark. After successfully implementation, cpu, cpuAvx, cpuSample, disk, memory, and network folders are created which contains output of respective benchmark.

Run each benchmark test separately:

CPU

Command	Used
gcc -O -o cpu_bench cpu_bench.c -lpthread	Compile cpu_bench.c file
./cpu_bench \$opt \$thread	Command to run cpu benchmark with AVX instruction. Replace \$opt and \$thread based on test condition. Possible options for \$opt are 1 and 2 where 1 represents iops and 2 represents flops. Possible options for \$thread are 1 2 3 4. Eg. ./cpu_bench 1 4

CPU with AVX Instruction

Command	Used
gcc -O -mavx2 -o cpu_avx cpu_avx.c -lpthread	Compile cpu_avx.c file
./cpu_avx \$opt \$thread	Command to run cpu benchmark. Replace \$opt and \$thread based on test condition. Possible options for \$opt are 1 and 2 where 1 represents iops and 2 represents flops. Possible options for \$thread are 1 2 3 4. Eg. ./cpu_avx 1 4

CPU 600 Samples

Command	Used
gcc -O -o sample_cpu sample_cpu.c -lpthread	Compile sample_cpu.c file
./sample_cpu \$opt \$thread	Command to run cpu benchmark. Replace \$opt and \$thread based on test condition. Possible options for \$opt are 1 and 2 where 1 represents iops and 2 represents flops. Possible options for \$thread are 1 2 3 4. Eg. ./sample_cpu 1 4

Disk

Command	Used
gcc -O -o disk_bench disk_bench.c -lpthread	Compile disk_bench.c file
./disk_bench \$opt \$blocksize \$thread	Command to run disk benchmark. Replace \$opt and \$thread based on test condition. Possible options for \$opt are 1 ,2,3,4,5 where 1 represents write sequential, 2 represents write random, 3 represents read sequential, 4 represents read random and 5 represents read + write. Possible option for \$blocksize are 8B, 8KB, 8MB, 80MB. Possible options for \$thread are 1 2 3 4. Eg. ./disk_bench 1 8B 4

Memory

Command	Used
gcc -O -o memory_bench memory_bench.c -lpthread	Compile memory_bench.c file
./memory_bench \$opt \$blocksize \$thread	<p>Command to run memory benchmark. Replace \$opt and \$thread based on test condition.</p> <p>Possible options for \$opt are 1,2,3,4,5 where 1 represents write sequential, 2 represents write random, 3 represents read sequential, 4 represents read random and 5 represents read + write.</p> <p>Possible option for \$blocksize are 8B, 8KB, 8MB, 80MB.</p> <p>Possible options for \$thread are 1 2 3 4.</p> <p>Eg. ./memory_bench 1 8B 4</p>

Network

Command	Used
gcc -o network_server_bench network_server_bench.c -lpthread	Compile network_server_bench.c file
gcc -o network_client_bench network_client_bench.c -lpthread	Compile network_client_bench.c file
./network_server_bench \$connect \$thread	<p>Command to start server. Replace \$ connect and \$thread based on test condition.</p> <p>Possible options for \$ connect are 1 and 2 where 1 represents TCP and 2 represents UDP. Possible options for \$thread are 1 2 3 4.</p> <p>Eg. ./ network_server_bench 1 4</p>
./network_client_bench \$connect \$thread	<p>Command to start client. Replace \$ connect and \$thread based on test condition.</p> <p>Possible options for \$ connect are 1 and 2 where 1 represents TCP and 2 represents UDP. Possible options for \$thread are 1 2 3 4.</p> <p>Eg. ./ network_client_bench 1 4</p>

GPU Benchmark

Command	Used
nvcc -o gpubenchmark gpubenchmark.cu	Compile gpubenchmark.cu file
./gpubenchmark	Command to run file

GPU Memory Benchmark

Command	Used
nvcc -o gpumemorybenchmark gpumemorybenchmark.cu	Compile gpumemorybenchmark.cu file
./gpumemorybenchmark	Command to run file