

www.phoronix-test-suite.com

7742 ep

2 x AMD EPYC 7742 64-Core testing with a Supermicro H11DSi-NT v2.00 (2.1 BIOS) and ASPEED on Ubuntu 21.10 via the Phoronix Test Suite.

Automated Executive Summary

A had the most wins, coming in first place for 59% of the tests.

Based on the geometric mean of all complete results, the fastest (C) was 1.022x the speed of the slowest (BB). A was 0.993x the speed of C, D was 1x the speed of A, B was 0.995x the speed of D, BB was 0.991x the speed of B.

The results with the greatest spread from best to worst included:

Redis Memtier / Redis Benchmark (Test: GET) at 1.948x
Stargate Digital Audio Workstation (Sample Rate: 96000 - Buffer Size: 512) at 1.588x
Stargate Digital Audio Workstation (Sample Rate: 44100 - Buffer Size: 1024) at 1.562x
Redis Memtier / Redis Benchmark (Test: MIX) at 1.558x
Stargate Digital Audio Workstation (Sample Rate: 480000 - Buffer Size: 512) at 1.558x
ONNX Runtime (Model: super-resolution-10 - Device: CPU) at 1.508x
Stargate Digital Audio Workstation (Sample Rate: 96000 - Buffer Size: 1024) at 1.424x
ONNX Runtime (Model: shufflenet-v2-10 - Device: CPU) at 1.363x
PyHPC Benchmarks (Device: CPU - Backend: JAX - Project Size: 4194304 - Benchmark: Isoneutral Mixing) at 1.226x

Stargate Digital Audio Workstation (Sample Rate: 44100 - Buffer Size: 512) at 1.225x.

Test Systems:

A

B

BB

C

D

Processor: 2 x AMD EPYC 7742 64-Core @ 2.25GHz (128 Cores / 256 Threads), Motherboard: Supermicro H11DSi-NT v2.00 (2.1 BIOS), Chipset: AMD Starship/Matisse, Memory: 126GB, Disk: 280GB INTEL SSDPE21D280GA, Graphics: ASPEED, Monitor: VE228, Network: 2 x Intel 10G X550T

OS: Ubuntu 21.10, Kernel: 5.15.0-rc4-cluster-sched (x86_64), Desktop: GNOME Shell 40.5, Display Server: X Server, Vulkan: 1.1.182, Compiler: GCC 11.2.0, File-System: ext4, Screen Resolution: 1920x1080

Kernel Notes: Transparent Huge Pages: madvise
Compiler Notes: --build=x86_64-linux-gnu --disable-vtable-verify --disable-werror --enable-bootstrap --enable-cet --enable-checking=release --enable-clocale=gnu --enable-default-pie --enable-gnu-unique-object --enable-languages=c,ada,c++,go,brig,d,fortran,objc,obj-c++,m2 --enable-libphobos-checking=release --enable-libstdcxx-debug --enable-libstdcxx-time=yes --enable-link-serialization=2 --enable-multiarch --enable-multilib --enable-nls --enable-objc-gc=auto --enable-offload-targets=nvptx-none=/build/gcc-11-ZPT0kp/gcc-11-11.2.0/debian/tmp-nvptx/usr,amdgcn-amdhsa=/build/gcc-11-ZPT0kp/gcc-11-11.2.0/debian/tmp-gcn/usr --enable-plugin --enable-shared --enable-threads=posix --host=x86_64-linux-gnu --program-prefix=x86_64-linux-gnu- --target=x86_64-linux-gnu --with-abi=m64 --with-arch-32=i686 --with-build-config=bootstrap-lto-lean --with-default-libstdcxx-abi=new --with-gcc-major-version-only --with-multilib-list=m32,m64,mx32 --with-target-system-zlib=auto --with-tune=generic --without-cuda-driver -v
Processor Notes: Scaling Governor: acpi-cpufreq schedutil (Boost: Enabled) - CPU Microcode: 0x8301034
Python Notes: Python 3.9.7
Security Notes: itlb_multihit: Not affected + l1tf: Not affected + mds: Not affected + meltdown: Not affected + spec_store_bypass: Mitigation of SSB disabled via prctl and seccomp + spectre_v1: Mitigation of usercopy/swapgs barriers and __user pointer sanitization + spectre_v2: Mitigation of Full AMD retpoline IBPB: conditional IBRS_FW STIBP: conditional RSB filling + srbds: Not affected + tsx_async_abort: Not affected

	A	B	BB	C	D
JPEG XL libjxl - PNG - 5 (MP/s)	52.81	**52.09**	53.03	**53.34**	52.57
Normalized	99.01%	97.66%	99.42%	100%	98.56%
Standard Deviation		1%			
JPEG XL libjxl - PNG - 7 (MP/s)	**7.35**	7.15	7.12	7.07	**7.01**
Normalized	100%	97.28%	96.87%	96.19%	95.37%
Standard Deviation		0.1%			
JPEG XL libjxl - PNG - 8 (MP/s)	**0.7**	**0.7**	**0.69**	**0.69**	**0.69**
Normalized	100%	100%	98.57%	98.57%	98.57%
Standard Deviation		0%			
JPEG XL libjxl - JPEG - 5 (MP/s)	59.7	**60.04**	58.8	59.24	**58.46**
Normalized	99.43%	100%	97.93%	98.67%	97.37%

7742 ep

Standard Deviation		2.3%			
JPEG XL libjxl - JPEG - 7 (MP/s)	61.15	58.55	58.26	58.4	60.32
Normalized	100%	95.75%	95.27%	95.5%	98.64%
Standard Deviation		1.6%			
JPEG XL libjxl - JPEG - 8 (MP/s)	21.87	21.65	21.43	21.62	21.74
Normalized	100%	98.99%	97.99%	98.86%	99.41%
Standard Deviation		0.3%			
JPEG XL Decoding libjxl - 1	43.06	43.32	41.55	43.09	43.06
Normalized	99.4%	100%	95.91%	99.47%	99.4%
Standard Deviation		0.1%			
JPEG XL Decoding libjxl - All (MP/s)	151.02	153.13	147.14	155.51	147.64
Normalized	97.11%	98.47%	94.62%	100%	94.94%
Standard Deviation		3.8%			
rav1e - 1 (FPS)	0.372	0.372	0.371	0.371	0.371
Normalized	100%	100%	99.73%	99.73%	99.73%
Standard Deviation		0.2%			
rav1e - 5 (FPS)	1.646	1.653	1.666	1.644	1.658
Normalized	98.8%	99.22%	100%	98.68%	99.52%
Standard Deviation		0.6%			
rav1e - 6 (FPS)	2.176	2.198	2.185	2.18	2.171
Normalized	99%	100%	99.41%	99.18%	98.77%
Standard Deviation		0.7%			
rav1e - 10 (FPS)	4.748	4.670	4.708	4.619	4.61
Normalized	100%	98.36%	99.16%	97.28%	97.09%
Standard Deviation		1%			
Stargate Digital Audio Workstation - 44100 - 512	2.761709	2.759399	2.743255	3.360503	3.211419
Normalized	82.18%	82.11%	81.63%	100%	95.56%
Standard Deviation		1.1%			
Stargate Digital Audio Workstation - 96000 - 512	2.033194		1.968436	2.011051	3.125304
Normalized	65.06%		62.98%	64.35%	100%
Stargate Digital Audio Workstation - 192000 - 512	1.375502		1.349265	1.365165	1.396307
Normalized	98.51%		96.63%	97.77%	100%
Stargate Digital Audio Workstation - 44100 - 1024	2.901815		2.872484	2.876094	4.48783
Normalized	64.66%		64.01%	64.09%	100%
Stargate Digital Audio Workstation - 480000 - 512	2.652187		2.673665	2.664475	4.130945
Normalized	64.2%		64.72%	64.5%	100%
Stargate Digital Audio Workstation - 96000 - 1024	2.10949		2.729096	2.974576	2.089359
Normalized	70.92%		91.75%	100%	70.24%
Stargate Digital Audio Workstation - 192000 - 1024	1.439159		1.434837	1.424212	1.425348
Normalized	100%		99.7%	98.96%	99.04%
Stargate Digital Audio Workstation - 480000 - 1024	2.789309		2.773968	2.787997	2.74212
Normalized	100%		99.45%	99.95%	98.31%

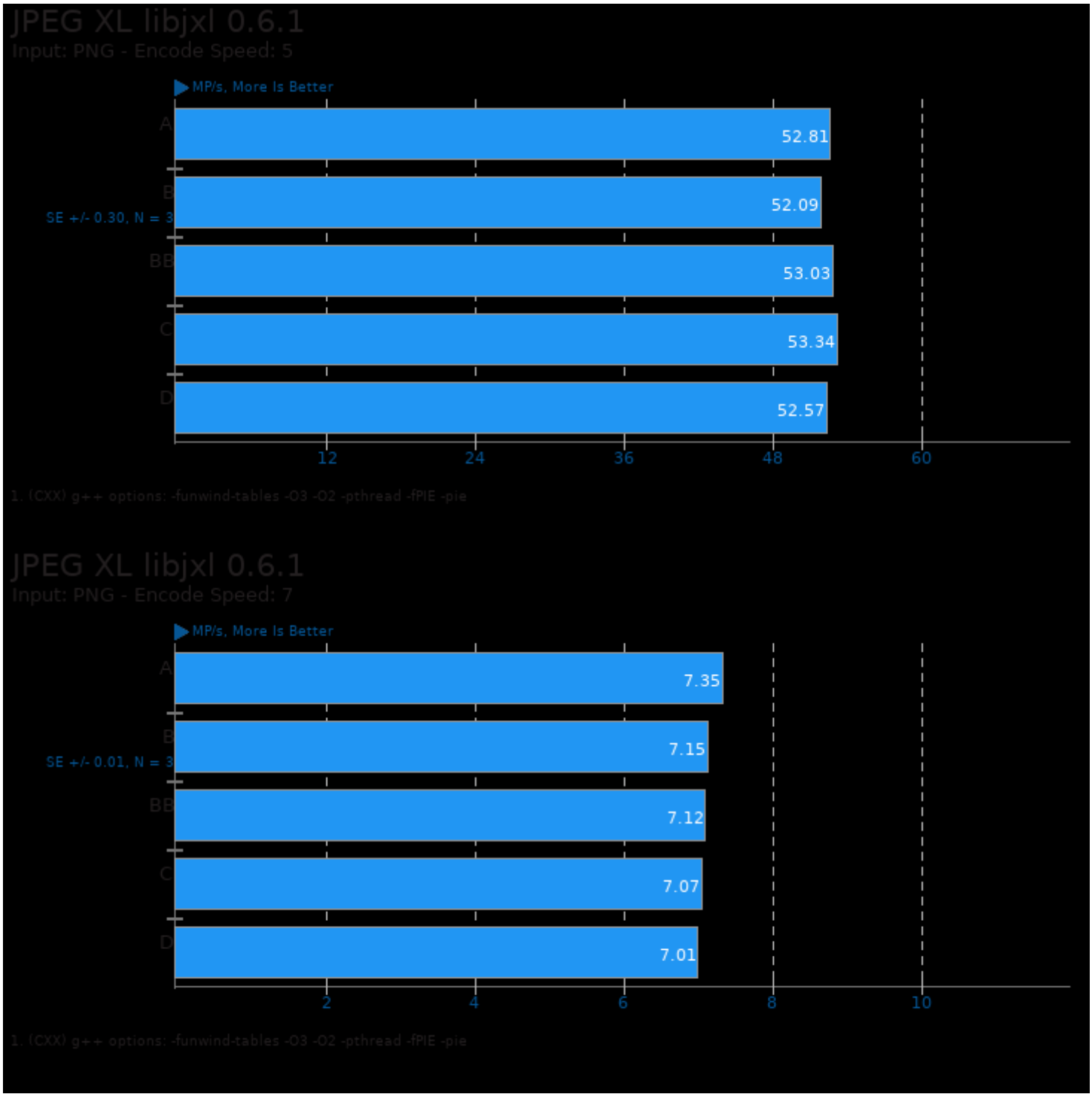
Redis Memtier / Redis Benchmark - L.a.L.l (Reqs/sec)	**919118**	931015	**978295**	946614
Normalized	93.95%	95.17%	100%	96.76%
Redis Memtier / Redis Benchmark - L.a.L.l (Reqs/sec)	**923792**	**951214**	940912	943307
Normalized	97.12%	100%	98.92%	99.17%
Redis Memtier / Redis Benchmark - GET	**445704**	**228760**	231366	232656
Normalized	100%	51.33%	51.91%	52.2%
Redis Memtier / Redis Benchmark - MIX	**367026**	**235553**	240458	236336
Normalized	100%	64.18%	65.52%	64.39%
Redis Memtier / Redis Benchmark - SET	235373	232637	**236157**	**232380**
Normalized	99.67%	98.51%	100%	98.4%
Blender - BMW27 - CPU-Only (sec)	24.02	23.87	**23.49**	**24.45**
Normalized	97.79%	98.41%	100%	96.07%
Blender - Classroom - CPU-Only (sec)	**46.37**	**46.61**	46.58	46.6
Normalized	100%	99.49%	99.55%	99.51%
Blender - Fishy Cat - CPU-Only (sec)	**34.86**	**35.71**	35.66	35.13
Normalized	100%	97.62%	97.76%	99.23%
Blender - Barbershop - CPU-Only (sec)	78.08	**78.54**	**77.81**	78.07
Normalized	99.65%	99.07%	100%	99.67%
Blender - Pabellon Barcelona - CPU-Only (sec)	**63.88**	63.49	**63.48**	63.6
Normalized	99.37%	99.98%	100%	99.81%
ONNX Runtime - yolov4 - CPU (Inferences/min)	**212**	228	215	**229**
Normalized	92.58%	99.56%	93.89%	100%
ONNX Runtime - fcn-resnet101-11 - CPU	**178**	**192**	186	179
Normalized	92.71%	100%	96.88%	93.23%
ONNX Runtime - shufflenet-v2-10 - CPU (Inferences/min)	5612	**6563**	**4814**	5305
Normalized	85.51%	100%	73.35%	80.83%
ONNX Runtime - super-resolution-10 - CPU (Inferences/min)	**4488**	5430	6750	**6766**
Normalized	66.33%	80.25%	99.76%	100%
PyHPC Benchmarks - CPU - JAX - 16384 - Equation of State (sec)	0.001	0.001	0.001	0.001
PyHPC Benchmarks - CPU - JAX - 16384 - Isoneutral Mixing (sec)	0.003	0.003	0.003	0.003
PyHPC Benchmarks - CPU - JAX - 65536 - Equation of State (sec)	0.001	0.001	0.001	0.001

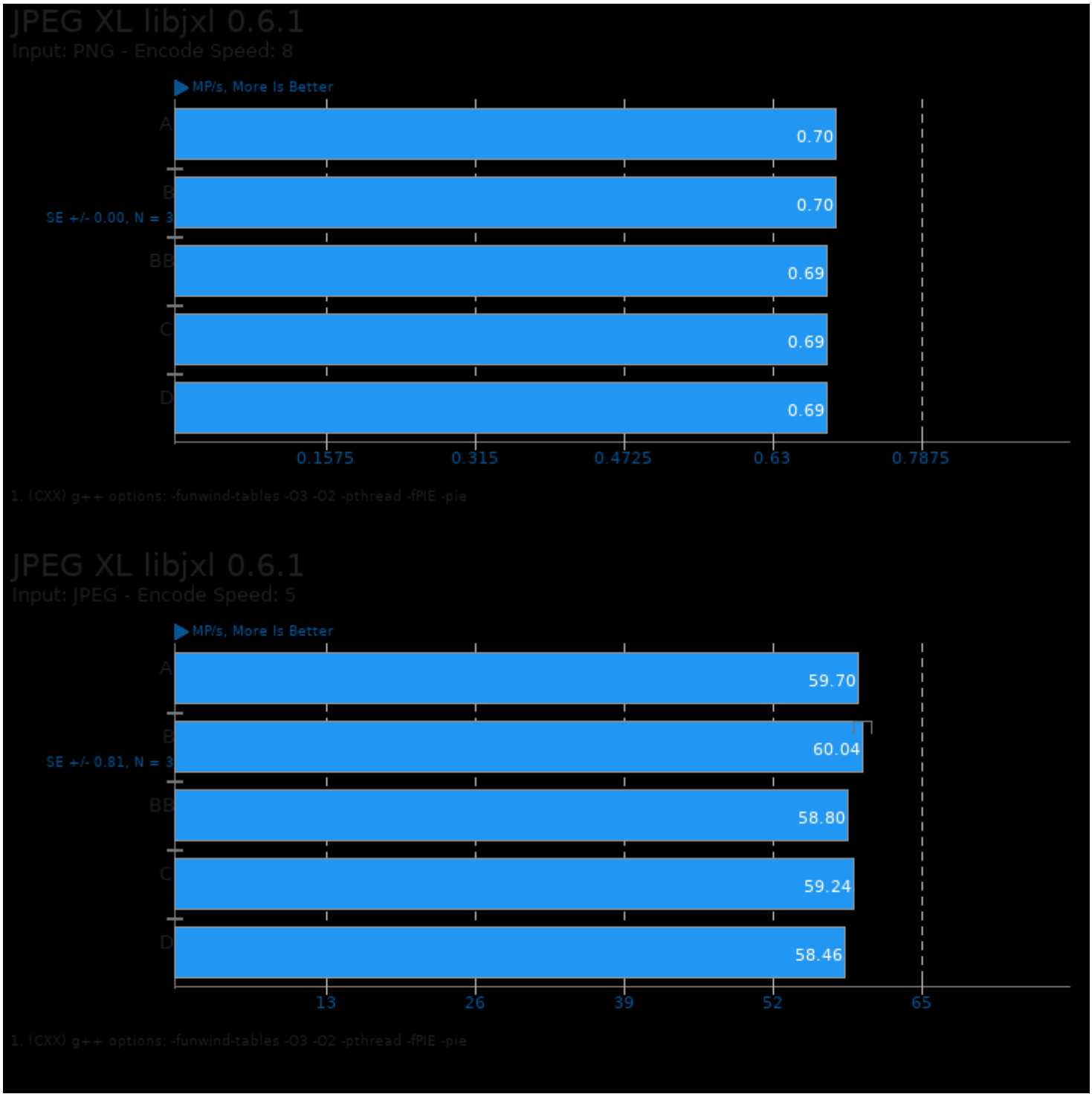
PyHPC Benchmarks - CPU - JAX - 65536 - Isoneutral Mixing (sec)	0.024	0.024	0.023	0.024
Normalized	95.83%	95.83%	100%	95.83%
PyHPC Benchmarks - CPU - JAX - 262144 - Equation of State (sec)	0.002	0.002	0.002	0.002
PyHPC Benchmarks - CPU - JAX - 262144 - Isoneutral Mixing (sec)	0.122	0.118	0.128	0.127
Normalized	96.72%	100%	92.19%	92.91%
PyHPC Benchmarks - CPU - JAX - 1048576 - Equation of State	0.012	0.013	0.013	0.012
Normalized	100%	92.31%	92.31%	100%
PyHPC Benchmarks - CPU - JAX - 1048576 - Isoneutral Mixing	0.207	0.206	0.212	0.201
Normalized	97.1%	97.57%	94.81%	100%
PyHPC Benchmarks - CPU - JAX - 4194304 - Equation of State	0.097	0.1	0.101	0.101
Normalized	100%	97%	96.04%	96.04%
PyHPC Benchmarks - CPU - JAX - 4194304 - Isoneutral Mixing	0.818	1.003	0.836	0.835
Normalized	100%	81.56%	97.85%	97.96%
PyHPC Benchmarks - CPU - Numba - 16384 - Equation of State (sec)	0.001	0.001	0.001	0.001
PyHPC Benchmarks - CPU - Numba - 16384 - Isoneutral Mixing (sec)	0.005	0.005	0.005	0.005
PyHPC Benchmarks - CPU - Numba - 65536 - Equation of State (sec)	0.005	0.005	0.005	0.005
PyHPC Benchmarks - CPU - Numba - 65536 - Isoneutral Mixing (sec)	0.021	0.021	0.021	0.021
PyHPC Benchmarks - CPU - Numpy - 16384 - Equation of State (sec)	0.003	0.003	0.003	0.003
PyHPC Benchmarks - CPU - Numpy - 16384 - Isoneutral Mixing (sec)	0.009	0.009	0.009	0.009
PyHPC Benchmarks - CPU - Numpy - 65536 - Equation of State (sec)	0.018	0.018	0.018	0.018
PyHPC Benchmarks - CPU - Numpy - 65536 - Isoneutral Mixing (sec)	0.039	0.04	0.04	0.04
Normalized	100%	97.5%	97.5%	97.5%

PyHPC Benchmarks - CPU - Aesara - 16384 - Equation of State (sec)	0.002	0.002	0.002	0.002
PyHPC Benchmarks - CPU - Aesara - 16384 - Isoneutral Mixing (sec)	0.006	0.006	0.006	0.006
PyHPC Benchmarks - CPU - Aesara - 65536 - Equation of State (sec)	0.007	0.007	0.007	0.007
PyHPC Benchmarks - CPU - Aesara - 65536 - Isoneutral Mixing (sec)	0.024	0.024	0.025	0.024
Normalized	100%	100%	96%	100%
PyHPC Benchmarks - CPU - Numba - 262144 - Equation of State (sec)	0.019	0.019	0.019	0.019
PyHPC Benchmarks - CPU - Numba - 262144 - Isoneutral Mixing (sec)	0.079	0.079	0.08	0.079
Normalized	100%	100%	98.75%	100%
PyHPC Benchmarks - CPU - Numpy - 262144 - Equation of State (sec)	0.07	0.072	0.074	0.071
Normalized	100%	97.22%	94.59%	98.59%
PyHPC Benchmarks - CPU - Numpy - 262144 - Isoneutral Mixing (sec)	0.156	0.159	0.159	0.158
Normalized	100%	98.11%	98.11%	98.73%
PyHPC Benchmarks - CPU - Aesara - 262144 - Equation of State (sec)	0.024	0.024	0.024	0.024
PyHPC Benchmarks - CPU - Aesara - 262144 - Isoneutral Mixing (sec)	0.092	0.092	0.092	0.092
PyHPC Benchmarks - CPU - Numba - 1048576 - Equation of State (sec)	0.078	0.079	0.078	0.078
Normalized	100%	98.73%	100%	100%
PyHPC Benchmarks - CPU - Numba - 1048576 - Isoneutral Mixing (sec)	0.36	0.372	0.362	0.362
Normalized	100%	96.77%	99.45%	99.45%
PyHPC Benchmarks - CPU - Numba - 4194304 - Equation of State (sec)	0.313	0.312	0.313	0.313
Normalized	99.68%	100%	99.68%	99.68%

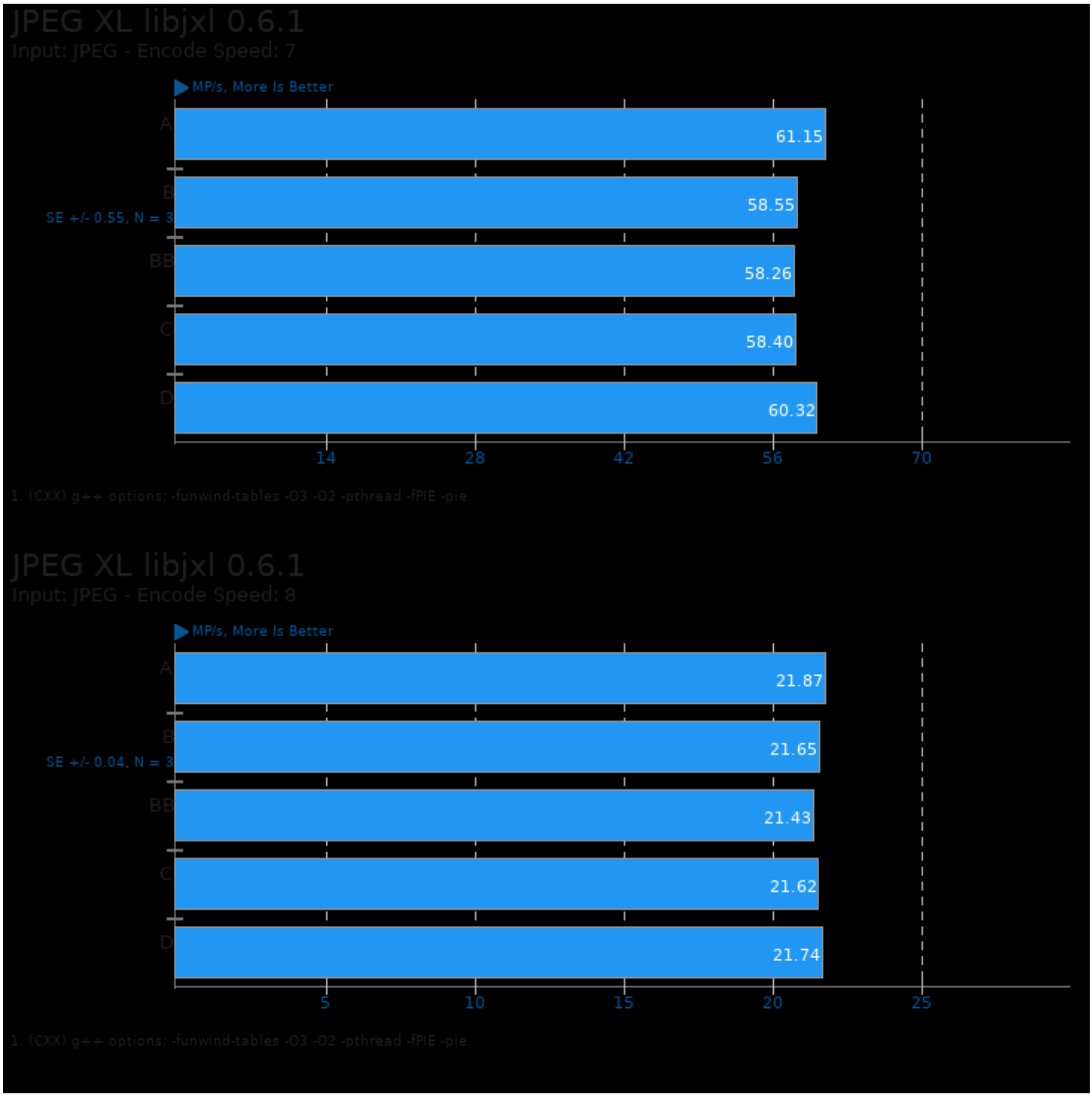
PyHPC Benchmarks - CPU - Numba - 4194304 - Isoneutral Mixing (sec)	**1.518**	**1.583**	1.526	1.528
Normalized	100%	95.89%	99.48%	99.35%
PyHPC Benchmarks - CPU - Numpy - 1048576 - Equation of State (sec)	0.307	**0.315**	0.311	**0.305**
Normalized	99.35%	96.83%	98.07%	100%
PyHPC Benchmarks - CPU - Numpy - 1048576 - Isoneutral Mixing (sec)	0.731	**0.751**	0.733	**0.73**
Normalized	99.86%	97.2%	99.59%	100%
PyHPC Benchmarks - CPU - Numpy - 4194304 - Equation of State (sec)	**1.57**	**1.604**	1.587	1.574
Normalized	100%	97.88%	98.93%	99.75%
PyHPC Benchmarks - CPU - Numpy - 4194304 - Isoneutral Mixing (sec)	3.004	**3.075**	3.002	**3.001**
Normalized	99.9%	97.59%	99.97%	100%
PyHPC Benchmarks - CPU - PyTorch - 16384 - Equation of State (sec)	0.001	0.001	0.001	0.001
PyHPC Benchmarks - CPU - PyTorch - 16384 - Isoneutral Mixing (sec)	0.006	0.006	0.006	0.006
PyHPC Benchmarks - CPU - PyTorch - 65536 - Equation of State (sec)	0.002	0.002	0.002	0.002
PyHPC Benchmarks - CPU - PyTorch - 65536 - Isoneutral Mixing (sec)	0.022	0.022	0.022	0.022
PyHPC Benchmarks - CPU - Aesara - 1048576 - Equation of State (sec)	**0.1**	**0.1**	**0.1**	**0.102**
Normalized	100%	100%	100%	98.04%
PyHPC Benchmarks - CPU - Aesara - 1048576 - Isoneutral Mixing (sec)	0.439	**0.441**	**0.438**	0.439
Normalized	99.77%	99.32%	100%	99.77%
PyHPC Benchmarks - CPU - Aesara - 4194304 - Equation of State (sec)	**0.396**	**0.396**	**0.397**	**0.396**
Normalized	100%	100%	99.75%	100%
PyHPC Benchmarks - CPU - Aesara - 4194304 - Isoneutral Mixing (sec)	1.844	**1.855**	**1.843**	1.847
Normalized	99.95%	99.35%	100%	99.78%

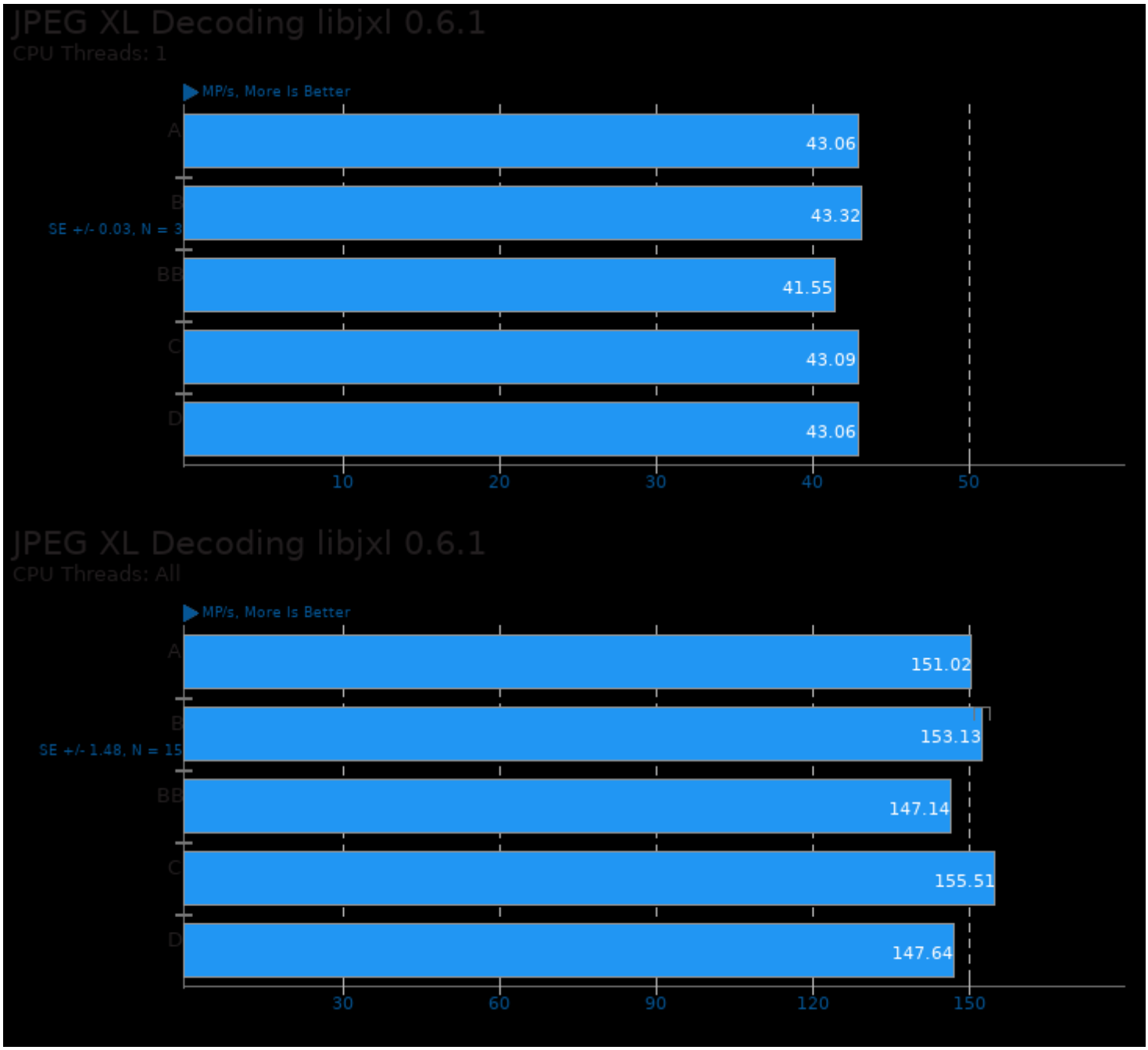
PyHPC Benchmarks - CPU - PyTorch - 262144 - Equation of State (sec)	0.007	0.006	0.006	0.007
Normalized	85.71%	100%	100%	85.71%
PyHPC Benchmarks - CPU - PyTorch - 262144 - Isoneutral Mixing (sec)	0.087	0.089	0.088	0.088
Normalized	100%	97.75%	98.86%	98.86%
PyHPC Benchmarks - CPU - PyTorch - 1048576 - Equation of State (sec)	0.029	0.028	0.028	0.029
Normalized	96.55%	100%	100%	96.55%
PyHPC Benchmarks - CPU - PyTorch - 1048576 - Isoneutral Mixing (sec)	0.456	0.453	0.454	0.455
Normalized	99.34%	100%	99.78%	99.56%
PyHPC Benchmarks - CPU - PyTorch - 4194304 - Equation of State (sec)	0.115	0.115	0.115	0.115
PyHPC Benchmarks - CPU - PyTorch - 4194304 - Isoneutral Mixing (sec)	2.014	1.989	2.014	1.971
Normalized	97.86%	99.1%	97.86%	100%
PyHPC Benchmarks - CPU - TensorFlow - 16384 - Equation of State (sec)	0.002	0.002	0.002	0.002
PyHPC Benchmarks - CPU - TensorFlow - 65536 - Equation of State (sec)	0.005	0.006	0.006	0.005
Normalized	100%	83.33%	83.33%	100%
PyHPC Benchmarks - CPU - TensorFlow - 262144 - Equation of State (sec)	0.017	0.017	0.016	0.017
Normalized	94.12%	94.12%	100%	94.12%
PyHPC Benchmarks - CPU - TensorFlow - 1048576 - Equation of State (sec)	0.036	0.04	0.036	0.037
Normalized	100%	90%	100%	97.3%
PyHPC Benchmarks - CPU - TensorFlow - 4194304 - Equation of State (sec)	0.167	0.172	0.169	0.17
Normalized	100%	97.09%	98.82%	98.24%


JPEG XL libjxl 0.6.1
Input: PNG - Encode Speed: 5
MP/s, More Is Better
A
B
BB
C
D
52.81
52.09
53.03
53.34
52.57
SE +/- 0.30, N = 3
12
24
36
48
60
1. (CXX) g++ options: -funwind-tables -O3 -O2 -pthread -fPIE -pie
JPEG XL libjxl 0.6.1
Input: PNG - Encode Speed: 7
MP/s, More Is Better
A
B
BB
C
D
7.35
7.15
7.12
7.07
7.01
SE +/- 0.01, N = 3
2
4
6
8
10
1. (CXX) g++ options: -funwind-tables -O3 -O2 -pthread -fPIE -pie

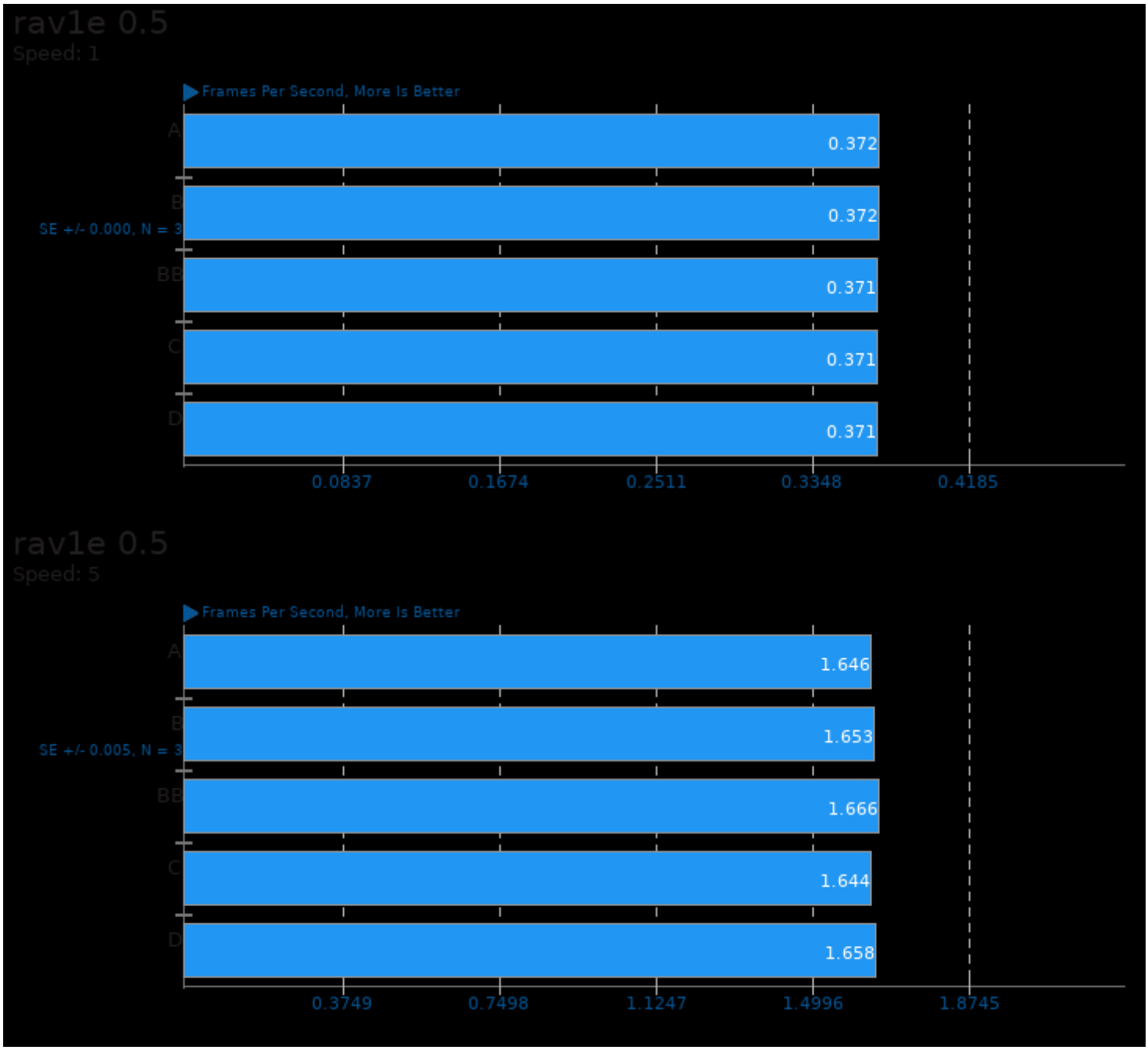

JPEG XL libjxl 0.6.1
Input: PNG - Encode Speed: 8
MP/s, More Is Better
A
B
BB
C
D
SE +/- 0.00, N = 3
0.70
0.70
0.69
0.69
0.69
0.1575
0.315
0.4725
0.63
0.7875
1. (CXX) g++ options: -funwind-tables -O3 -O2 -pthread -fPIE -pie

JPEG XL libjxl 0.6.1
Input: JPEG - Encode Speed: 5
MP/s, More Is Better
A
B
BB
C
D
SE +/- 0.81, N = 3
59.70
60.04
58.80
59.24
58.46
13
26
39
52
65
1. (CXX) g++ options: -funwind-tables -O3 -O2 -pthread -fPIE -pie

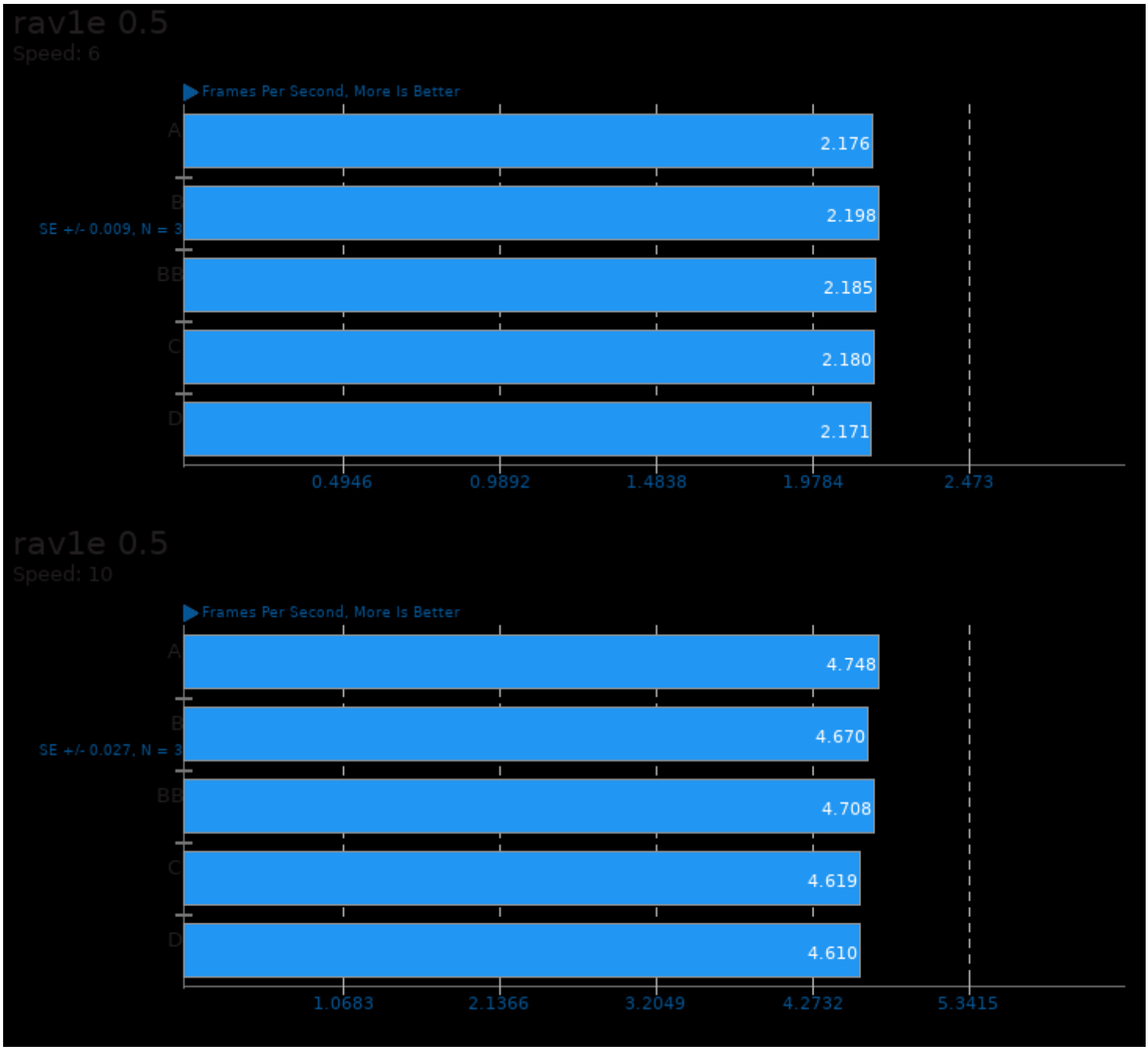

JPEG XL libjxl 0.6.1
Input: JPEG - Encode Speed: 7
MP/s, More Is Better
A
B
BB
C
D
61.15
58.55
58.26
58.40
60.32
SE +/- 0.55, N = 3
14
28
42
56
70
1. (CXX) g++ options: -funwind-tables -O3 -O2 -pthread -fPIE -pie
JPEG XL libjxl 0.6.1
Input: JPEG - Encode Speed: 8
MP/s, More Is Better
A
B
BB
C
D
21.87
21.65
21.43
21.62
21.74
SE +/- 0.04, N = 3
5
10
15
20
25
1. (CXX) g++ options: -funwind-tables -O3 -O2 -pthread -fPIE -pie

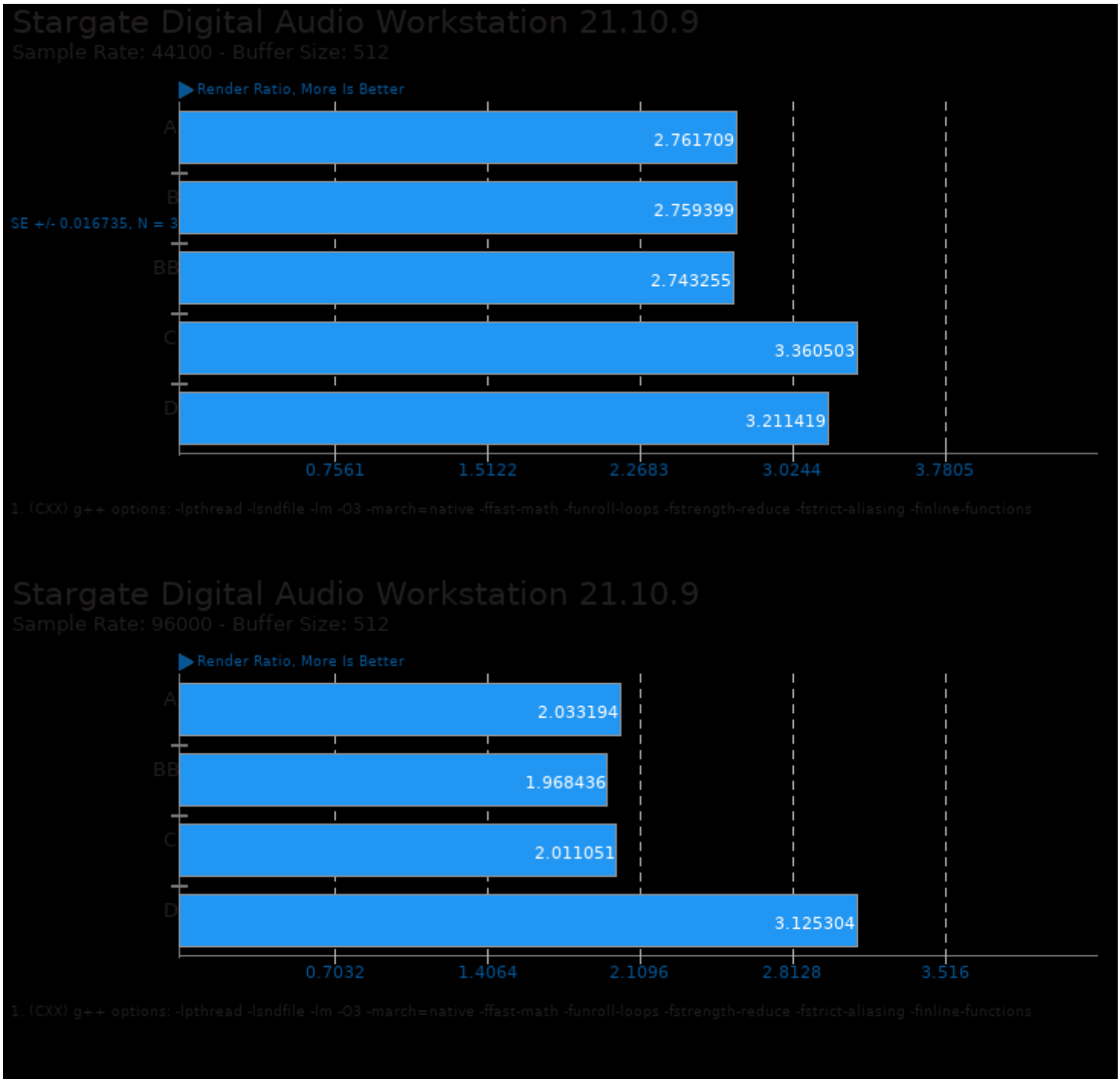

JPEG XL Decoding libjxl 0.6.1
CPU Threads: 1
MP/s, More Is Better
A
B
BB
C
D
43.06
43.32
41.55
43.09
43.06
SE +/- 0.03, N = 3
10
20
30
40
50
JPEG XL Decoding libjxl 0.6.1
CPU Threads: All
MP/s, More Is Better
A
B
BB
C
D
151.02
153.13
147.14
155.51
147.64
SE +/- 1.48, N = 15
30
60
90
120
150

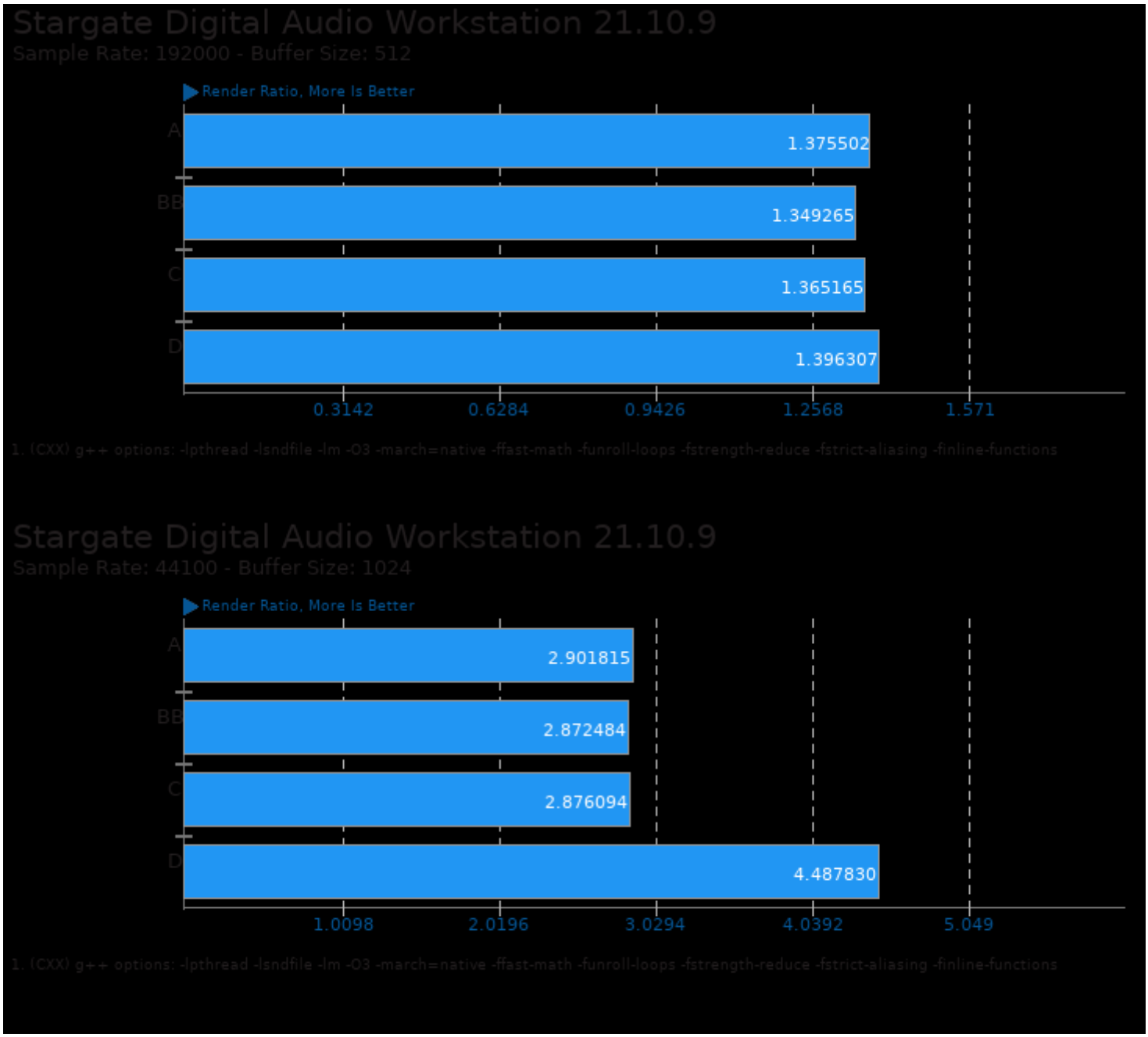
7742 ep

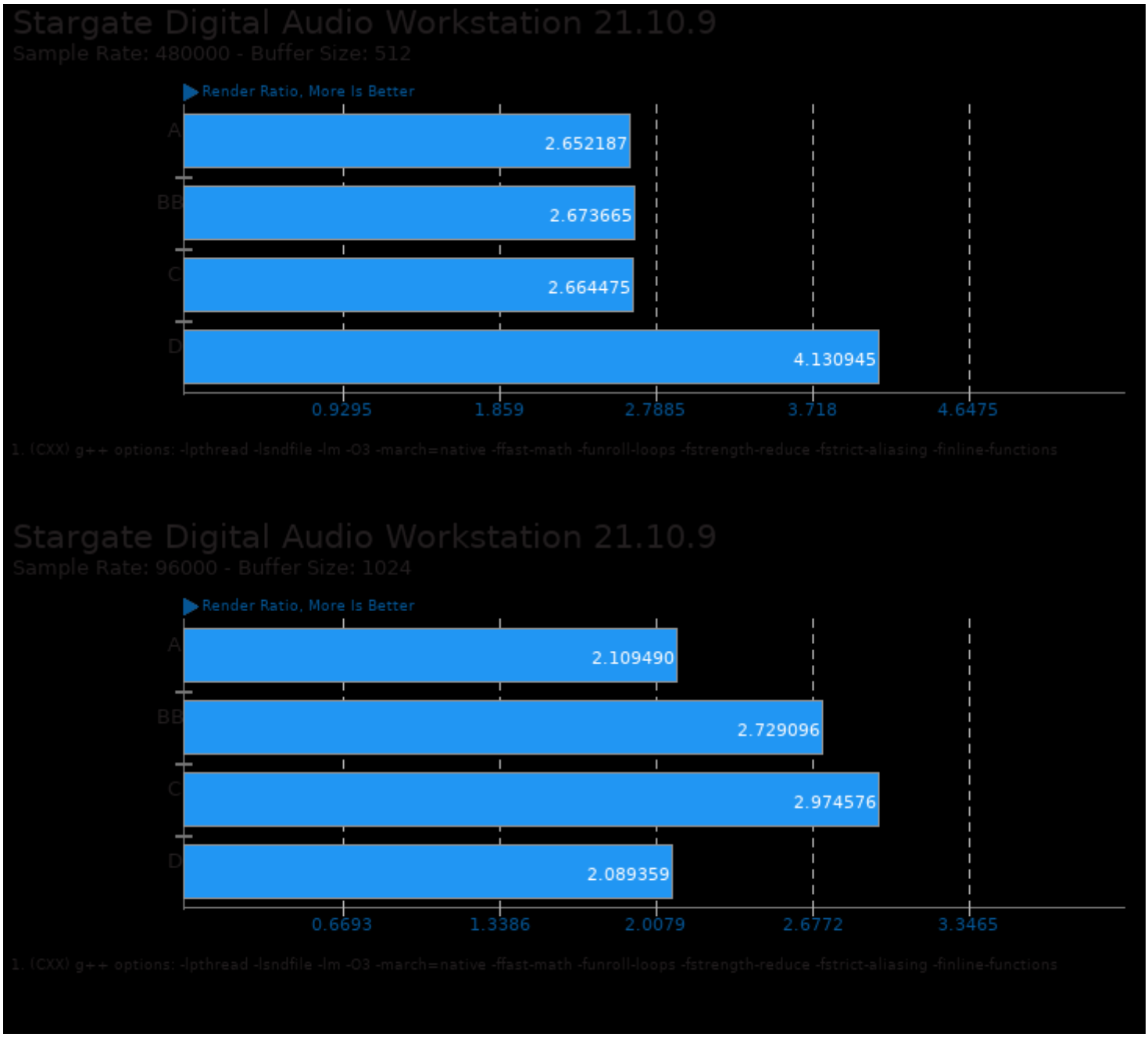

rav1e 0.5
Speed: 1
Frames Per Second, More Is Better
A
B
BB
C
D
SE +/- 0.000, N = 3
0.372
0.372
0.371
0.371
0.371
0.0837
0.1674
0.2511
0.3348
0.4185
rav1e 0.5
Speed: 5
Frames Per Second, More Is Better
A
B
BB
C
D
SE +/- 0.005, N = 3
1.646
1.653
1.666
1.644
1.658
0.3749
0.7498
1.1247
1.4996
1.8745

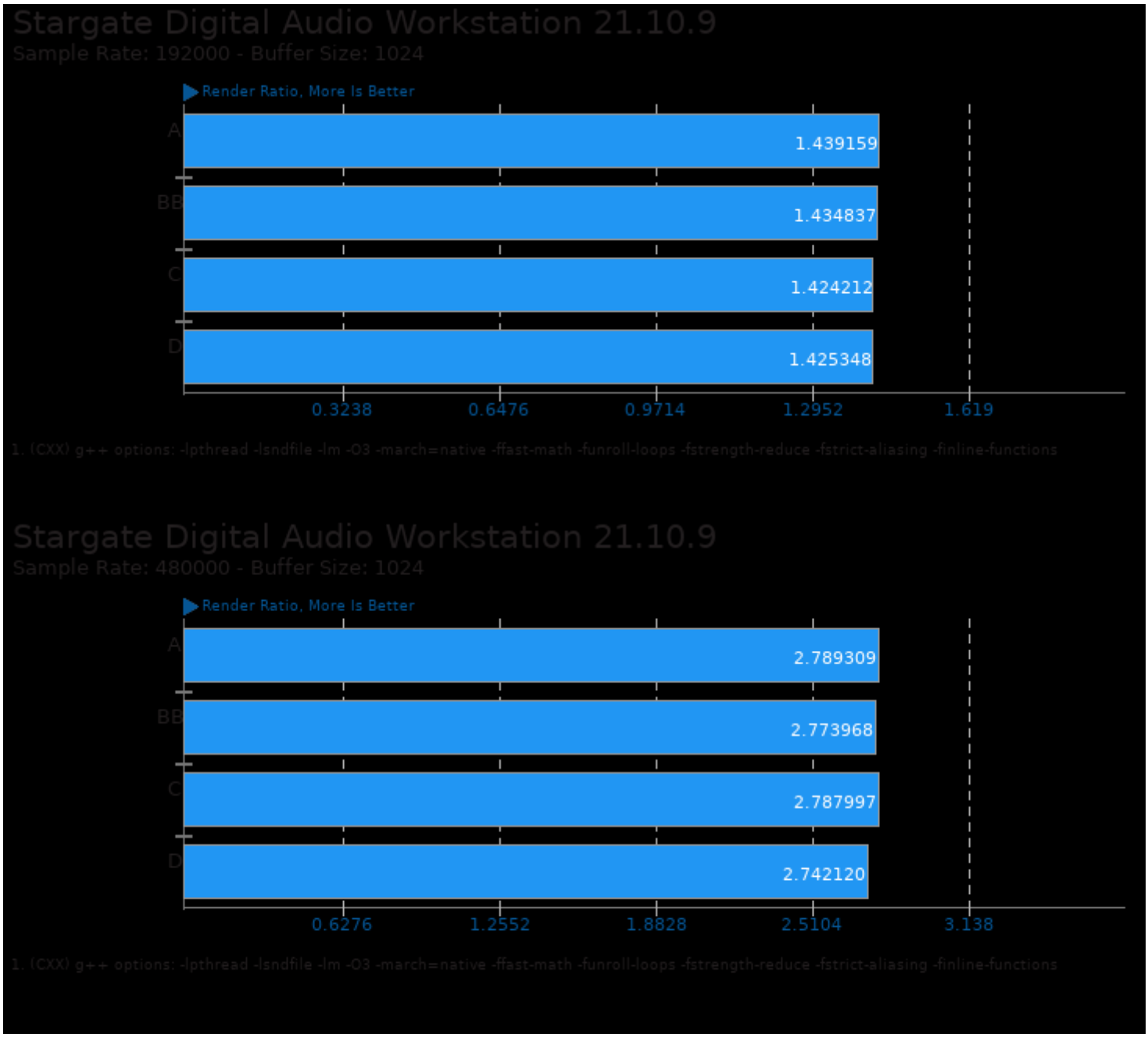
7742 ep

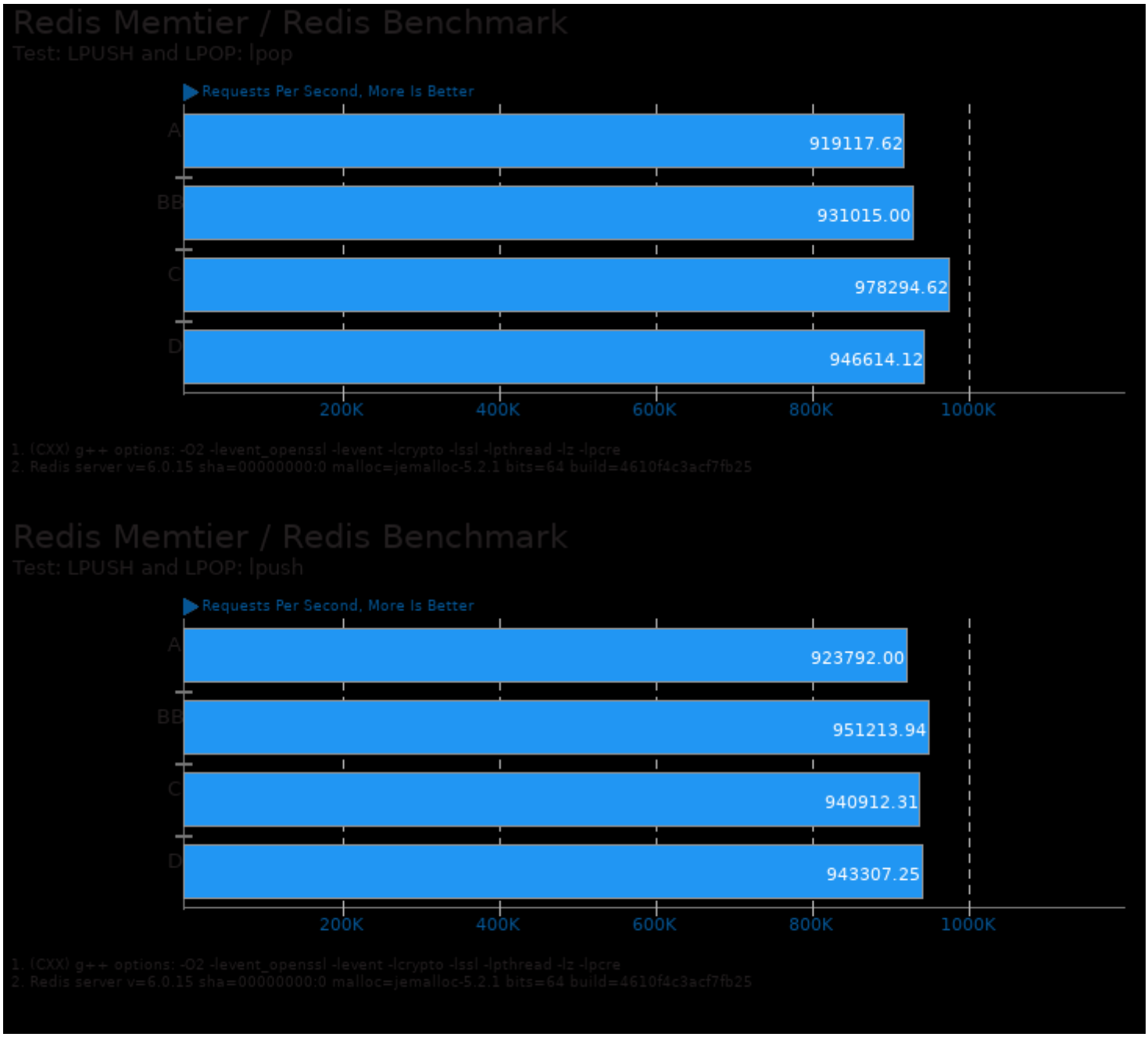

rav1e 0.5
Speed: 6
Frames Per Second, More Is Better
A
B
BB
C
D
2.176
2.198
2.185
2.180
2.171
SE +/- 0.009, N = 3
0.4946
0.9892
1.4838
1.9784
2.473
rav1e 0.5
Speed: 10
Frames Per Second, More Is Better
A
B
BB
C
D
4.748
4.670
4.708
4.619
4.610
SE +/- 0.027, N = 3
1.0683
2.1366
3.2049
4.2732
5.3415

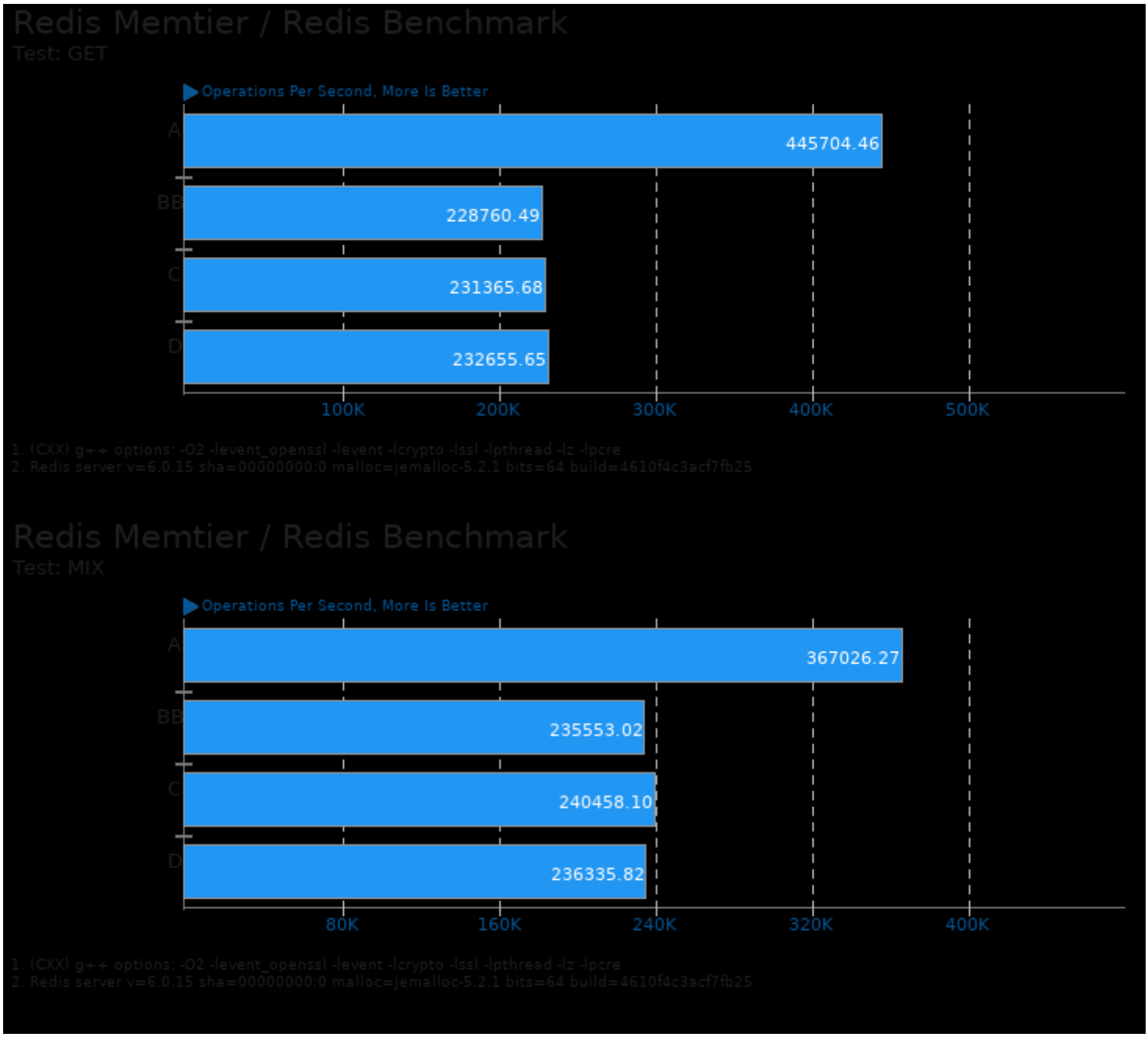

Stargate Digital Audio Workstation 21.10.9
Sample Rate: 44100 - Buffer Size: 512
Render Ratio, More Is Better
A
B
BB
C
D
2.761709
2.759399
2.743255
3.360503
3.211419
SE +/- 0.016735, N = 3
0.7561
1.5122
2.2683
3.0244
3.7805
1. (CXX) g++ options: -lpthread -lsndfile -lm -O3 -march=native -ffast-math -funroll-loops -fstrength-reduce -fstrict-aliasing -finline-functions
Stargate Digital Audio Workstation 21.10.9
Sample Rate: 96000 - Buffer Size: 512
Render Ratio, More Is Better
A
BB
C
D
2.033194
1.968436
2.011051
3.125304
0.7032
1.4064
2.1096
2.8128
3.516
1. (CXX) g++ options: -lpthread -lsndfile -lm -O3 -march=native -ffast-math -funroll-loops -fstrength-reduce -fstrict-aliasing -finline-functions


Stargate Digital Audio Workstation 21.10.9
Sample Rate: 192000 - Buffer Size: 512
Render Ratio, More Is Better
A
BB
C
D
1.375502
1.349265
1.365165
1.396307
0.3142
0.6284
0.9426
1.2568
1.571
1. (CXX) g++ options: -lpthread -lsndfile -lm -O3 -march=native -ffast-math -funroll-loops -fstrength-reduce -fstrict-aliasing -finline-functions
Stargate Digital Audio Workstation 21.10.9
Sample Rate: 44100 - Buffer Size: 1024
Render Ratio, More Is Better
A
BB
C
D
2.901815
2.872484
2.876094
4.487830
1.0098
2.0196
3.0294
4.0392
5.049
1. (CXX) g++ options: -lpthread -lsndfile -lm -O3 -march=native -ffast-math -funroll-loops -fstrength-reduce -fstrict-aliasing -finline-functions


Stargate Digital Audio Workstation 21.10.9
Sample Rate: 480000 - Buffer Size: 512
Render Ratio, More Is Better
A
BB
C
D
2.652187
2.673665
2.664475
4.130945
0.9295
1.859
2.7885
3.718
4.6475
1. (CXX) g++ options: -lpthread -lsndfile -lm -O3 -march=native -ffast-math -funroll-loops -fstrength-reduce -fstrict-aliasing -finline-functions
Stargate Digital Audio Workstation 21.10.9
Sample Rate: 96000 - Buffer Size: 1024
Render Ratio, More Is Better
A
BB
C
D
2.109490
2.729096
2.974576
2.089359
0.6693
1.3386
2.0079
2.6772
3.3465
1. (CXX) g++ options: -lpthread -lsndfile -lm -O3 -march=native -ffast-math -funroll-loops -fstrength-reduce -fstrict-aliasing -finline-functions

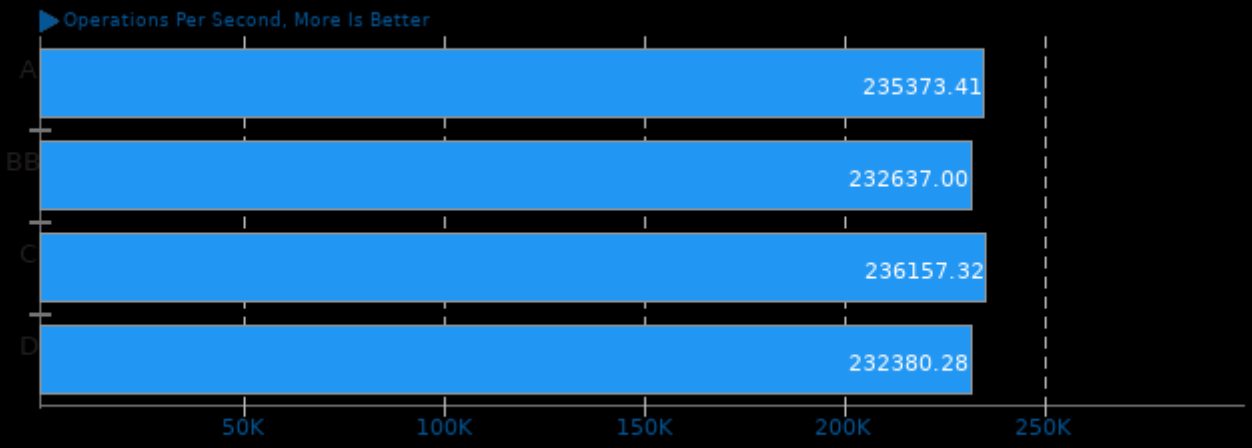

Stargate Digital Audio Workstation 21.10.9
Sample Rate: 192000 - Buffer Size: 1024
Render Ratio, More Is Better
A
BB
C
D
1.439159
1.434837
1.424212
1.425348
0.3238
0.6476
0.9714
1.2952
1.619
1. (CXX) g++ options: -lpthread -lsndfile -lm -O3 -march=native -ffast-math -funroll-loops -fstrength-reduce -fstrict-aliasing -finline-functions
Stargate Digital Audio Workstation 21.10.9
Sample Rate: 480000 - Buffer Size: 1024
Render Ratio, More Is Better
A
BB
C
D
2.789309
2.773968
2.787997
2.742120
0.6276
1.2552
1.8828
2.5104
3.138
1. (CXX) g++ options: -lpthread -lsndfile -lm -O3 -march=native -ffast-math -funroll-loops -fstrength-reduce -fstrict-aliasing -finline-functions


Redis Memtier / Redis Benchmark
Test: LPUSH and LPOP: lpop
Requests Per Second, More Is Better
A
BB
C
D
919117.62
931015.00
978294.62
946614.12
200K
400K
600K
800K
1000K
1. (CXX) g++ options: -O2 -levent_openssl -levent -lcrypto -lssl -lpthread -lz -lpcre
2. Redis server v=6.0.15 sha=00000000:0 malloc=jemalloc-5.2.1 bits=64 build=4610f4c3acf7fb25
Redis Memtier / Redis Benchmark
Test: LPUSH and LPOP: lpush
Requests Per Second, More Is Better
A
BB
C
D
923792.00
951213.94
940912.31
943307.25
200K
400K
600K
800K
1000K
1. (CXX) g++ options: -O2 -levent_openssl -levent -lcrypto -lssl -lpthread -lz -lpcre
2. Redis server v=6.0.15 sha=00000000:0 malloc=jemalloc-5.2.1 bits=64 build=4610f4c3acf7fb25


Redis Memtier / Redis Benchmark
Test: GET
Operations Per Second, More Is Better
A
445704.46
BB
228760.49
C
231365.68
D
232655.65
100K
200K
300K
400K
500K
1. (CXX) g++ options: -O2 -levent_openssl -levent -lcrypto -lssl -lpthread -lz -lpcre
2. Redis server v=6.0.15 sha=00000000:0 malloc=jemalloc-5.2.1 bits=64 build=4610f4c3acf7fb25
Redis Memtier / Redis Benchmark
Test: MIX
Operations Per Second, More Is Better
A
367026.27
BB
235553.02
C
240458.10
D
236335.82
80K
160K
240K
320K
400K
1. (CXX) g++ options: -O2 -levent_openssl -levent -lcrypto -lssl -lpthread -lz -lpcre
2. Redis server v=6.0.15 sha=00000000:0 malloc=jemalloc-5.2.1 bits=64 build=4610f4c3acf7fb25

Redis Memtier / Redis Benchmark

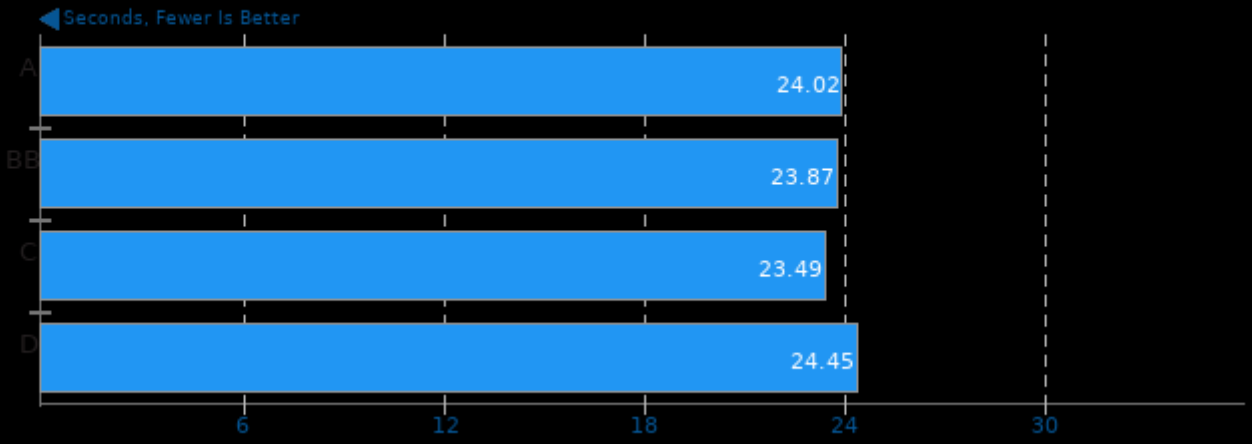
Test: SET



1. (CXX) g++ options: -O2 -levent_openssl -levent -lcrypto -lssl -lpthread -lz -lpcre
2. Redis server v=6.0.15 sha=00000000:0 malloc=jemalloc-5.2.1 bits=64 build=4610f4c3acf7fb25

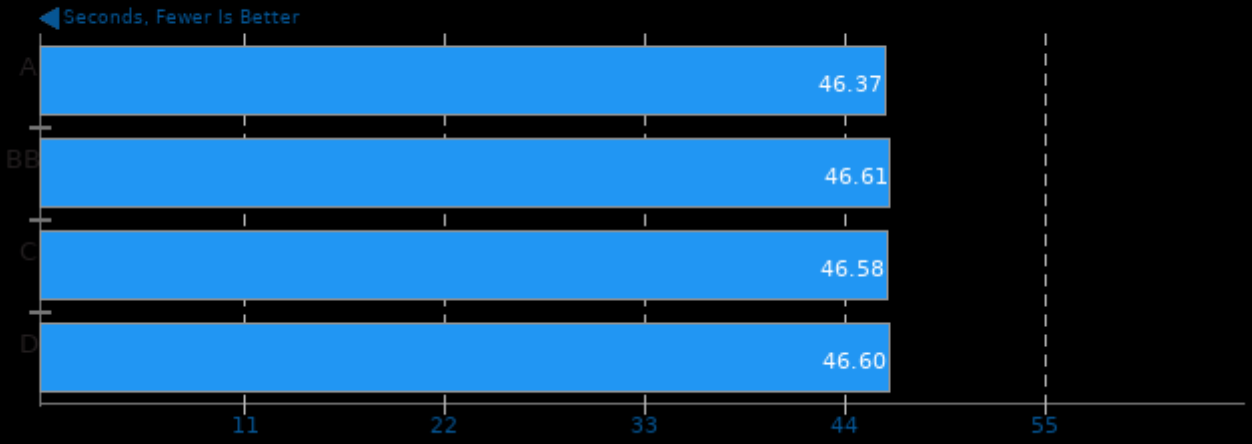
Blender 2.92

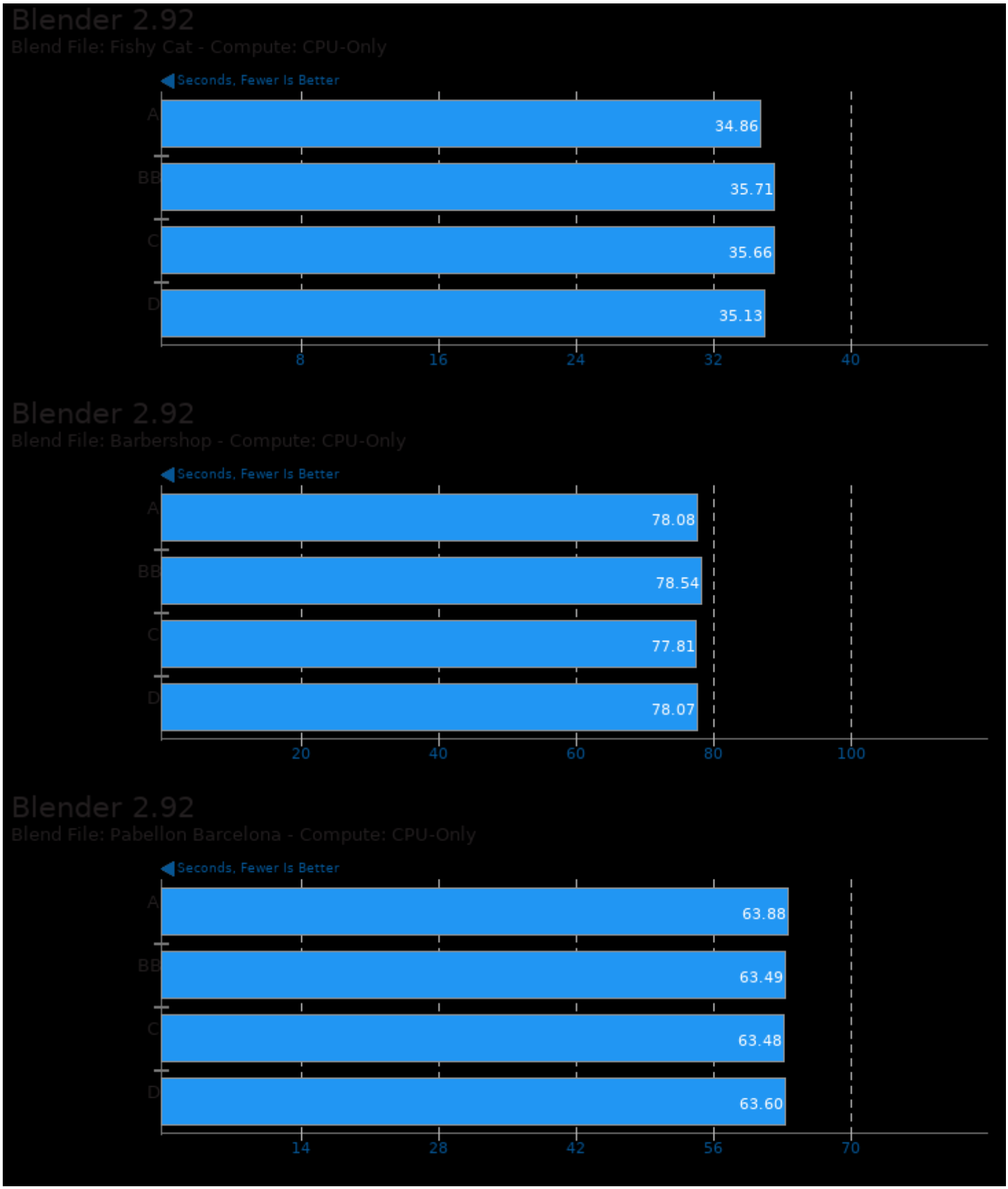
Blend File: BMW27 - Compute: CPU-Only

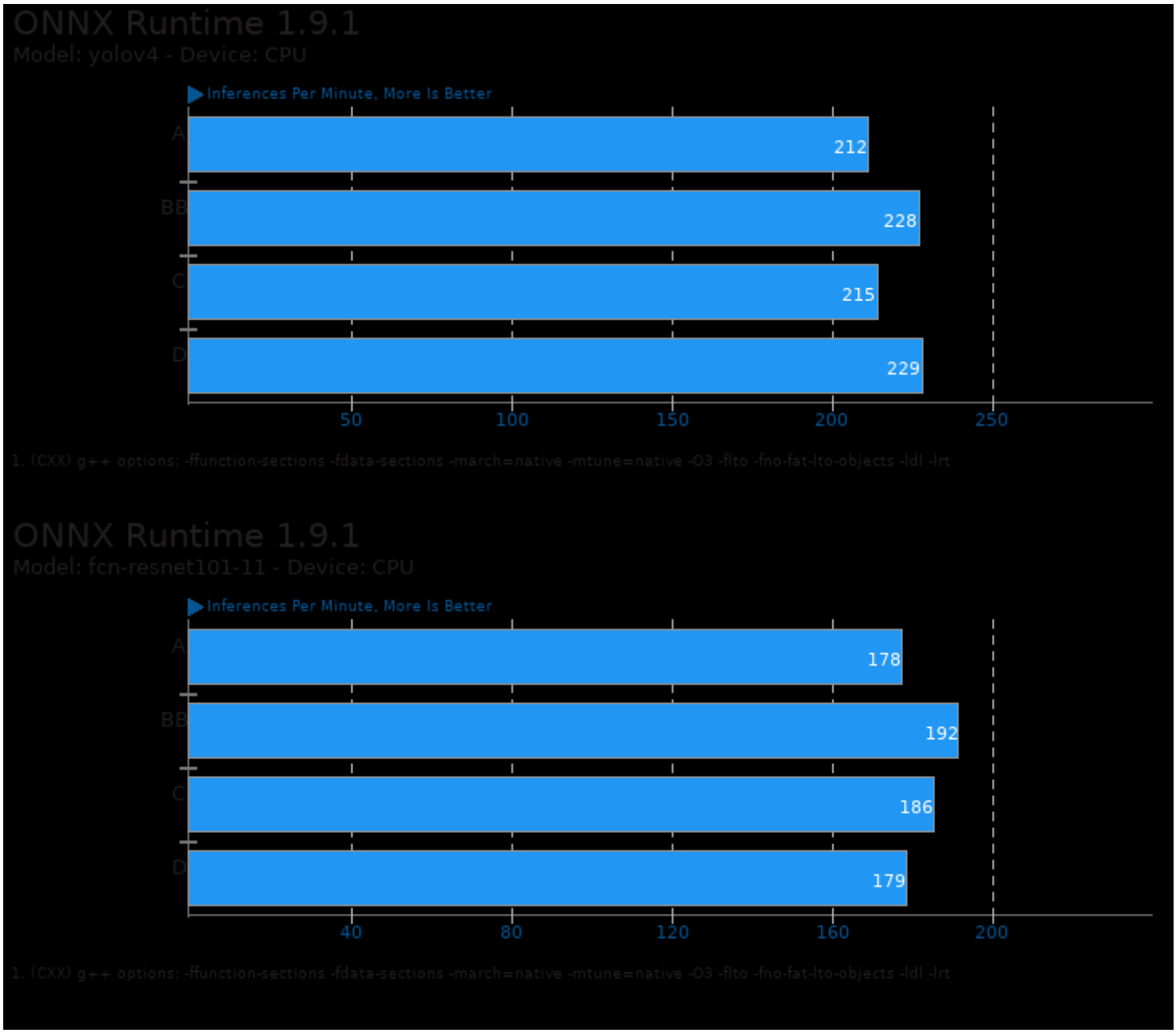


Blender 2.92

Blend File: Classroom - Compute: CPU-Only

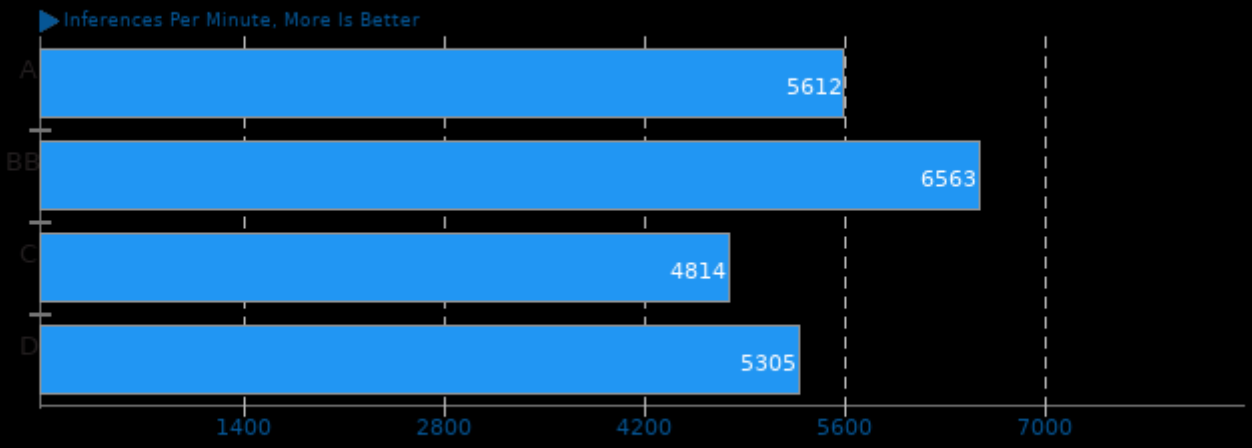



Blender 2.92
Blend File: Fishy Cat - Compute: CPU-Only
Seconds, Fewer Is Better
A
BB
C
D
34.86
35.71
35.66
35.13
8
16
24
32
40
Blender 2.92
Blend File: Barbershop - Compute: CPU-Only
Seconds, Fewer Is Better
A
BB
C
D
78.08
78.54
77.81
78.07
20
40
60
80
100
Blender 2.92
Blend File: Pabellon Barcelona - Compute: CPU-Only
Seconds, Fewer Is Better
A
BB
C
D
63.88
63.49
63.48
63.60
14
28
42
56
70


ONNX Runtime 1.9.1
Model: yolov4 - Device: CPU
Inferences Per Minute, More Is Better
A
BB
C
D
212
228
215
229
50
100
150
200
250
1. (CXX) g++ options: -ffunction-sections -fdata-sections -march=native -mtune=native -O3 -flto -fno-fat-lto-objects -ldl -lrt
ONNX Runtime 1.9.1
Model: fcn-resnet101-11 - Device: CPU
Inferences Per Minute, More Is Better
A
BB
C
D
178
192
186
179
40
80
120
160
200
1. (CXX) g++ options: -ffunction-sections -fdata-sections -march=native -mtune=native -O3 -flto -fno-fat-lto-objects -ldl -lrt

ONNX Runtime 1.9.1

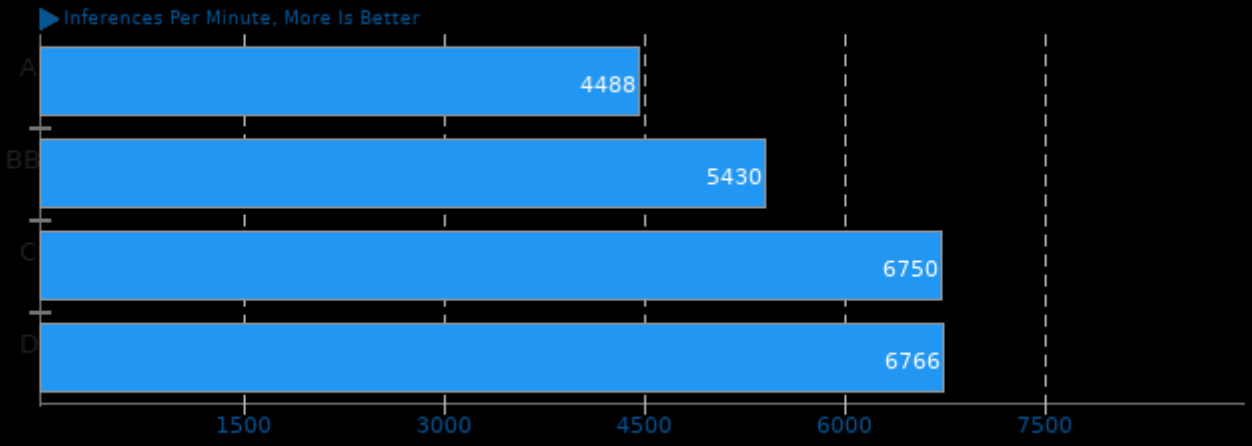
Model: shufflenet-v2-10 - Device: CPU



1. (CXX) g++ options: -ffunction-sections -fdata-sections -march=native -mtune=native -O3 -flto -fno-fat-lto-objects -ldl -lrt

ONNX Runtime 1.9.1

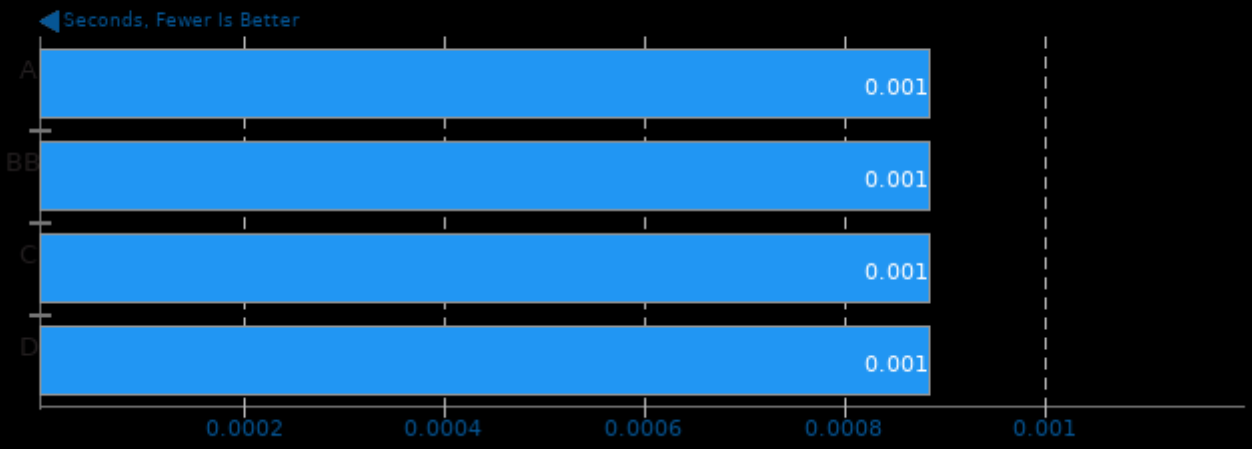
Model: super-resolution-10 - Device: CPU



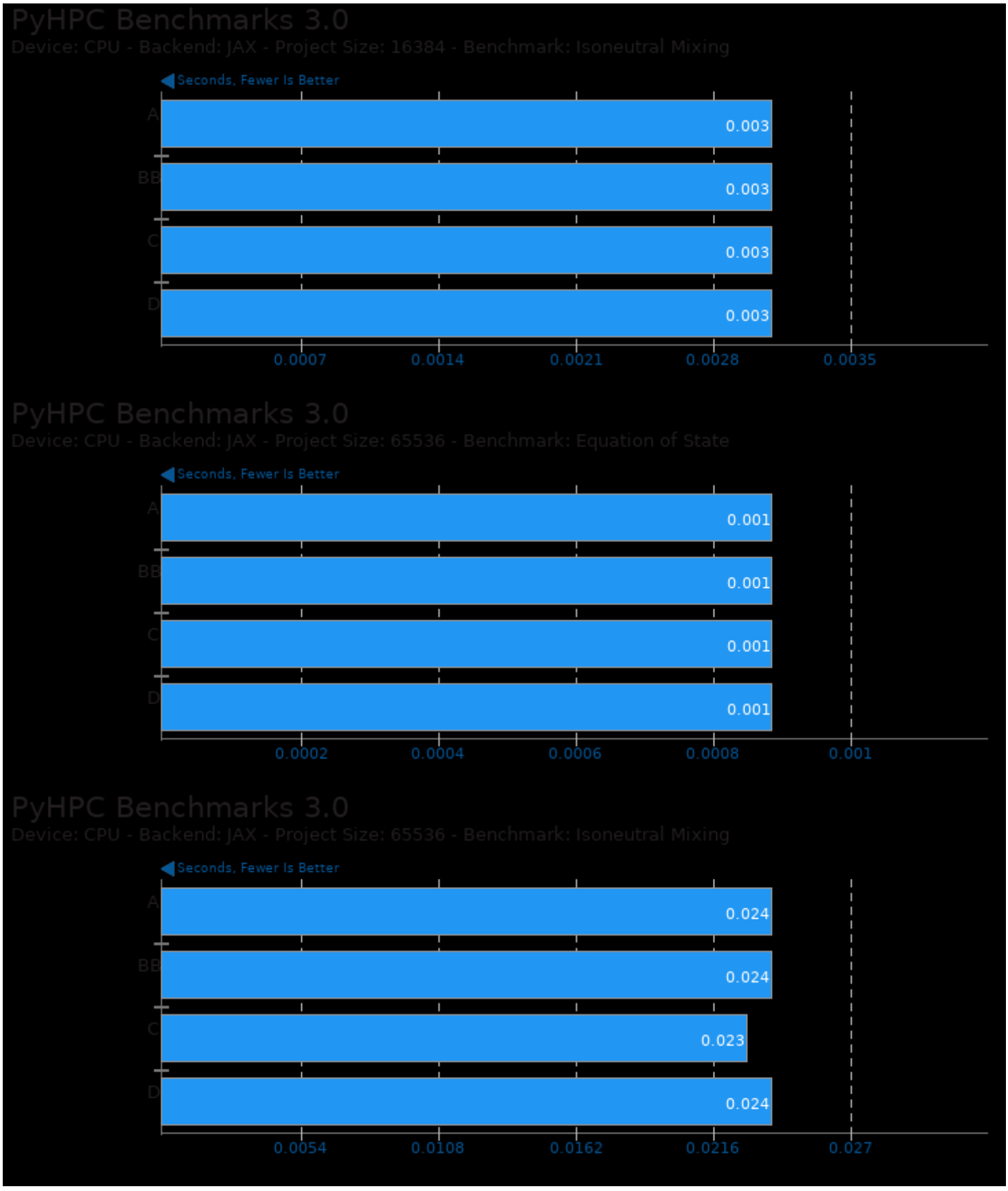
1. (CXX) g++ options: -ffunction-sections -fdata-sections -march=native -mtune=native -O3 -flto -fno-fat-lto-objects -ldl -lrt

PyHPC Benchmarks 3.0

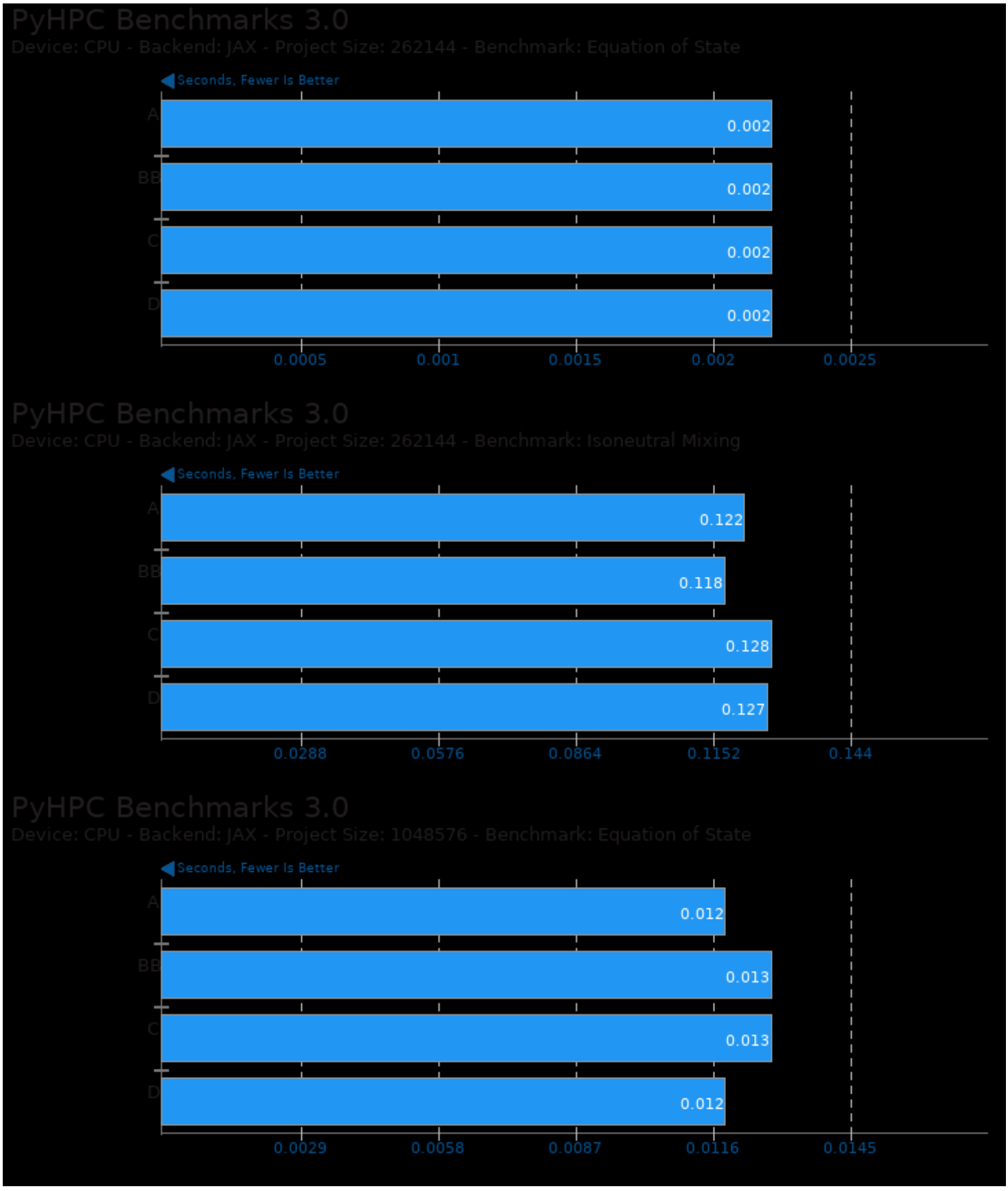
Device: CPU - Backend: JAX - Project Size: 16384 - Benchmark: Equation of State

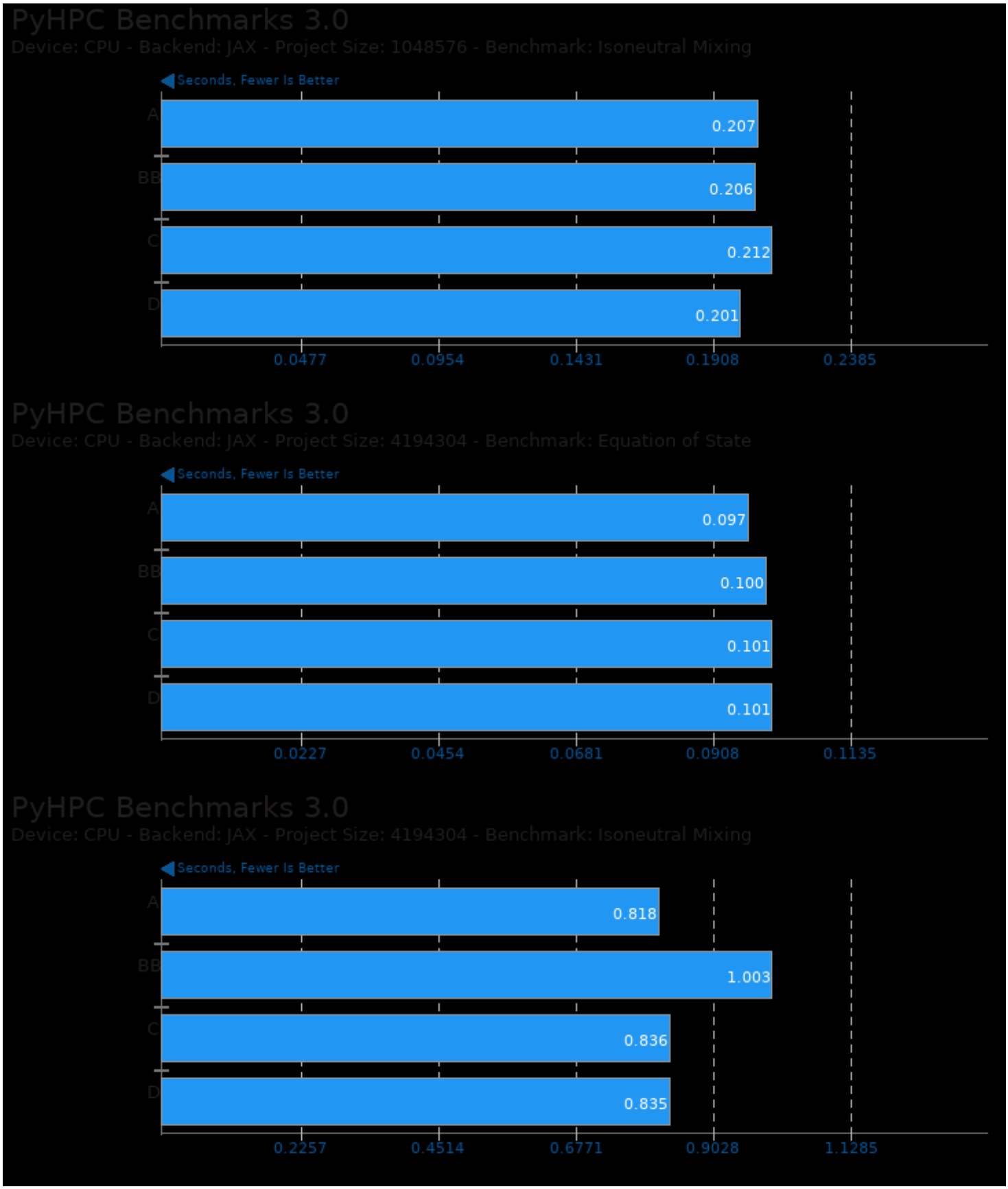


7742 ep

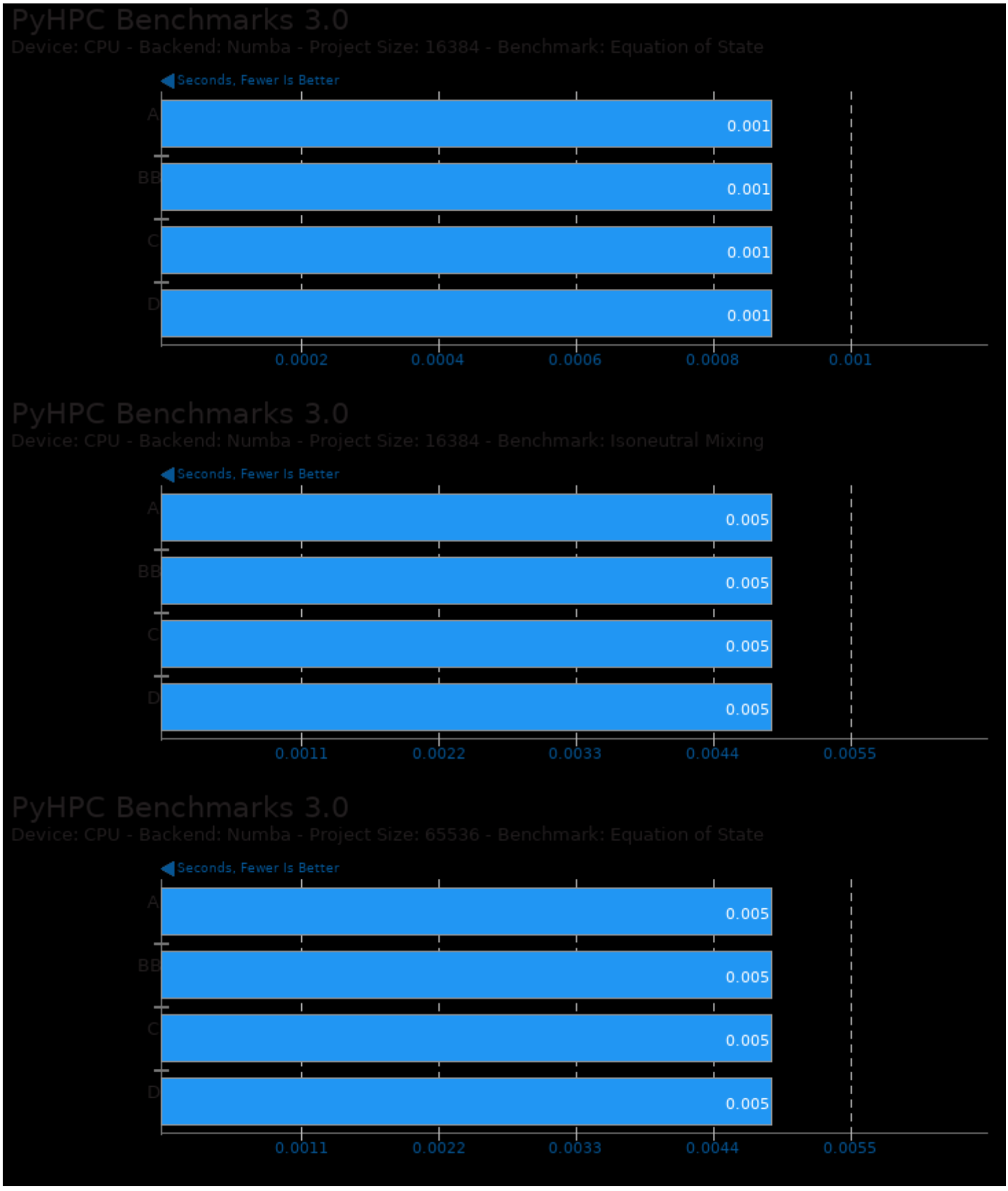

PyHPC Benchmarks 3.0
Device: CPU - Backend: JAX - Project Size: 16384 - Benchmark: Isoneutral Mixing
Seconds, Fewer Is Better
A
BB
C
D
0.003
0.003
0.003
0.003
0.0007
0.0014
0.0021
0.0028
0.0035
PyHPC Benchmarks 3.0
Device: CPU - Backend: JAX - Project Size: 65536 - Benchmark: Equation of State
Seconds, Fewer Is Better
A
BB
C
D
0.001
0.001
0.001
0.001
0.0002
0.0004
0.0006
0.0008
0.001
PyHPC Benchmarks 3.0
Device: CPU - Backend: JAX - Project Size: 65536 - Benchmark: Isoneutral Mixing
Seconds, Fewer Is Better
A
BB
C
D
0.024
0.024
0.023
0.024
0.0054
0.0108
0.0162
0.0216
0.027

7742 ep

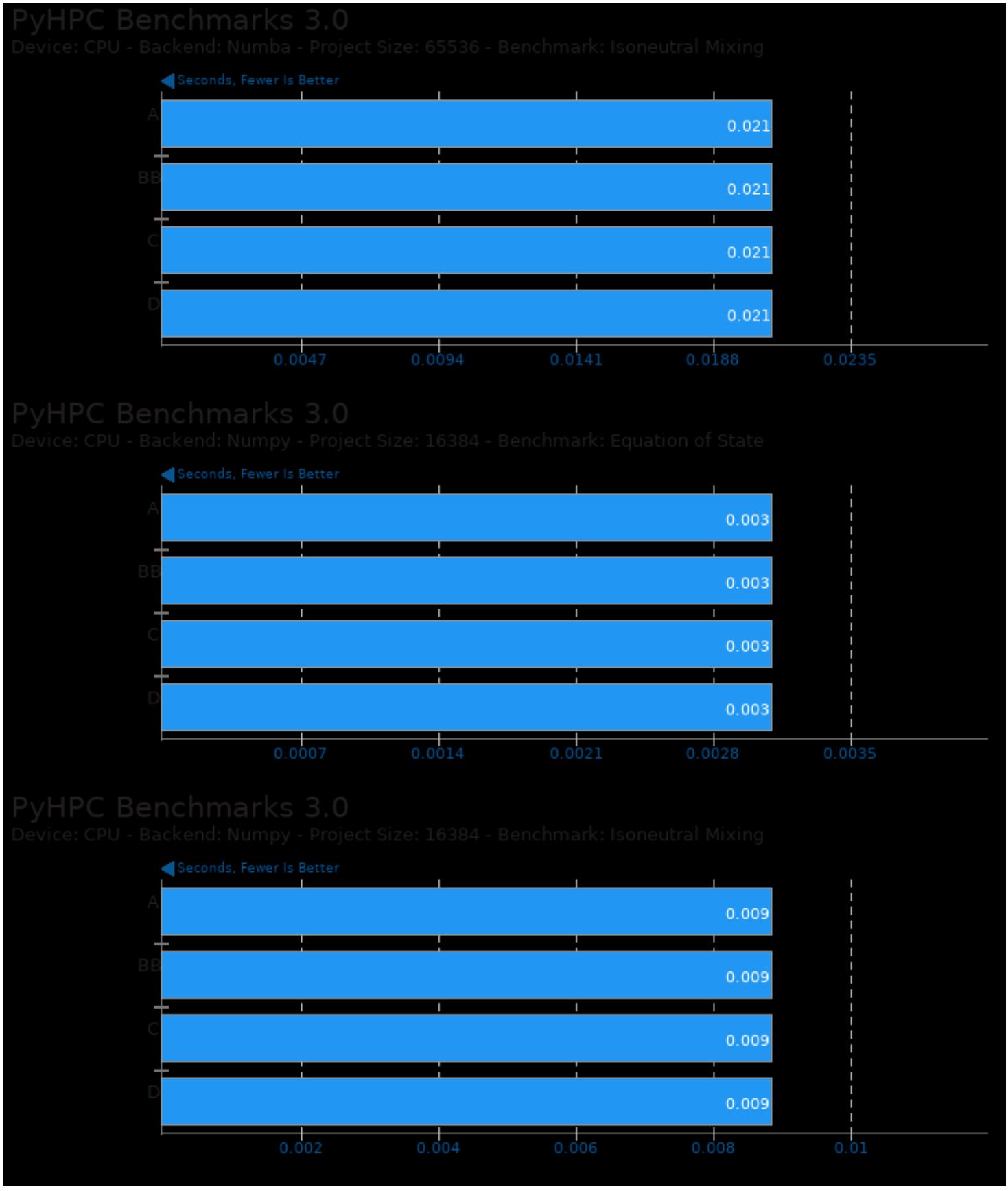

PyHPC Benchmarks 3.0
Device: CPU - Backend: JAX - Project Size: 262144 - Benchmark: Equation of State
Seconds, Fewer Is Better
A
BB
C
D
0.002
0.002
0.002
0.002
0.0005
0.001
0.0015
0.002
0.0025
PyHPC Benchmarks 3.0
Device: CPU - Backend: JAX - Project Size: 262144 - Benchmark: Isoneutral Mixing
Seconds, Fewer Is Better
A
BB
C
D
0.122
0.118
0.128
0.127
0.0288
0.0576
0.0864
0.1152
0.144
PyHPC Benchmarks 3.0
Device: CPU - Backend: JAX - Project Size: 1048576 - Benchmark: Equation of State
Seconds, Fewer Is Better
A
BB
C
D
0.012
0.013
0.013
0.012
0.0029
0.0058
0.0087
0.0116
0.0145

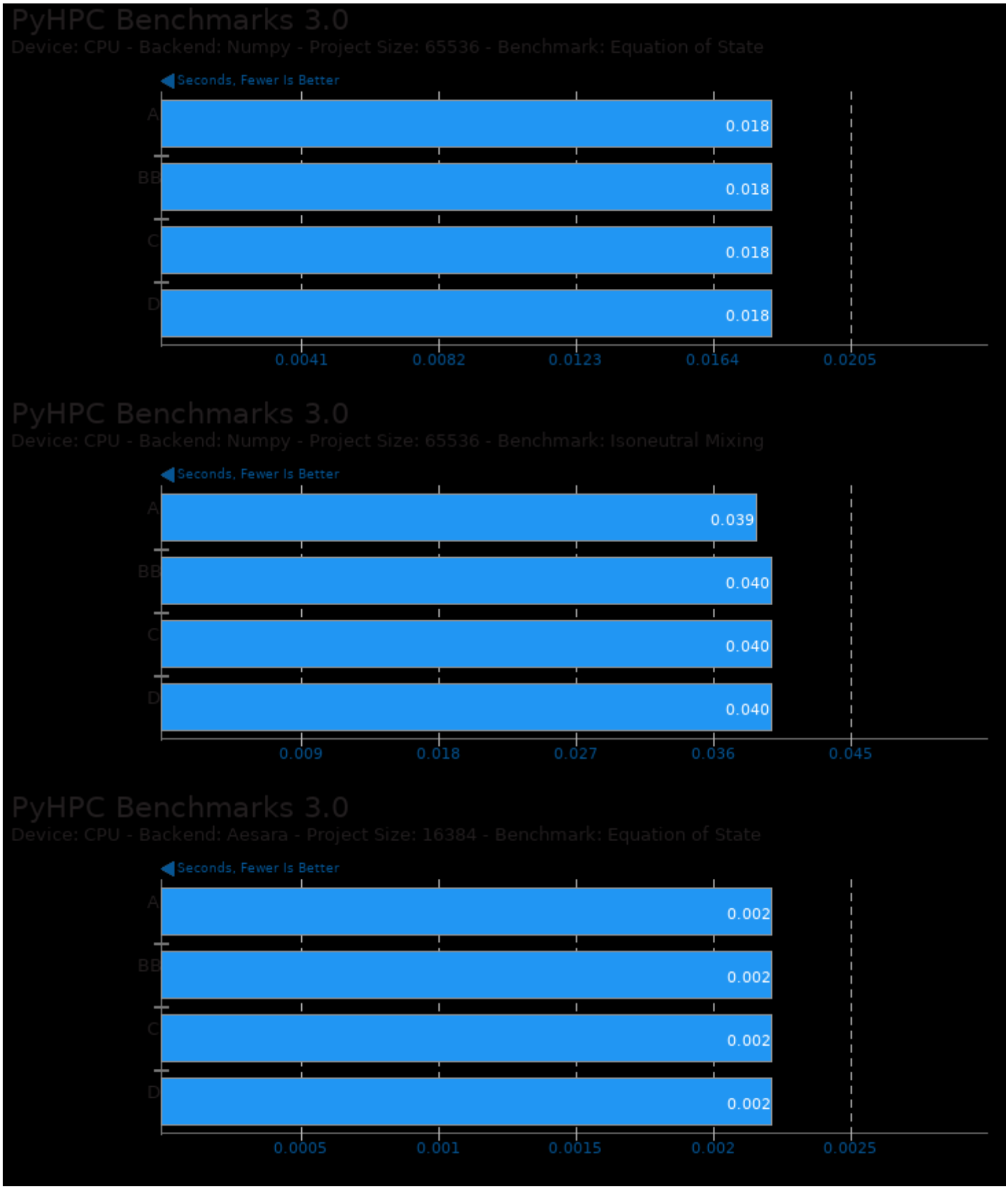

PyHPC Benchmarks 3.0
Device: CPU - Backend: JAX - Project Size: 1048576 - Benchmark: Isoneutral Mixing
Seconds, Fewer Is Better
A
BB
C
D
0.207
0.206
0.212
0.201
0.0477
0.0954
0.1431
0.1908
0.2385
PyHPC Benchmarks 3.0
Device: CPU - Backend: JAX - Project Size: 4194304 - Benchmark: Equation of State
Seconds, Fewer Is Better
A
BB
C
D
0.097
0.100
0.101
0.101
0.0227
0.0454
0.0681
0.0908
0.1135
PyHPC Benchmarks 3.0
Device: CPU - Backend: JAX - Project Size: 4194304 - Benchmark: Isoneutral Mixing
Seconds, Fewer Is Better
A
BB
C
D
0.818
1.003
0.836
0.835
0.2257
0.4514
0.6771
0.9028
1.1285

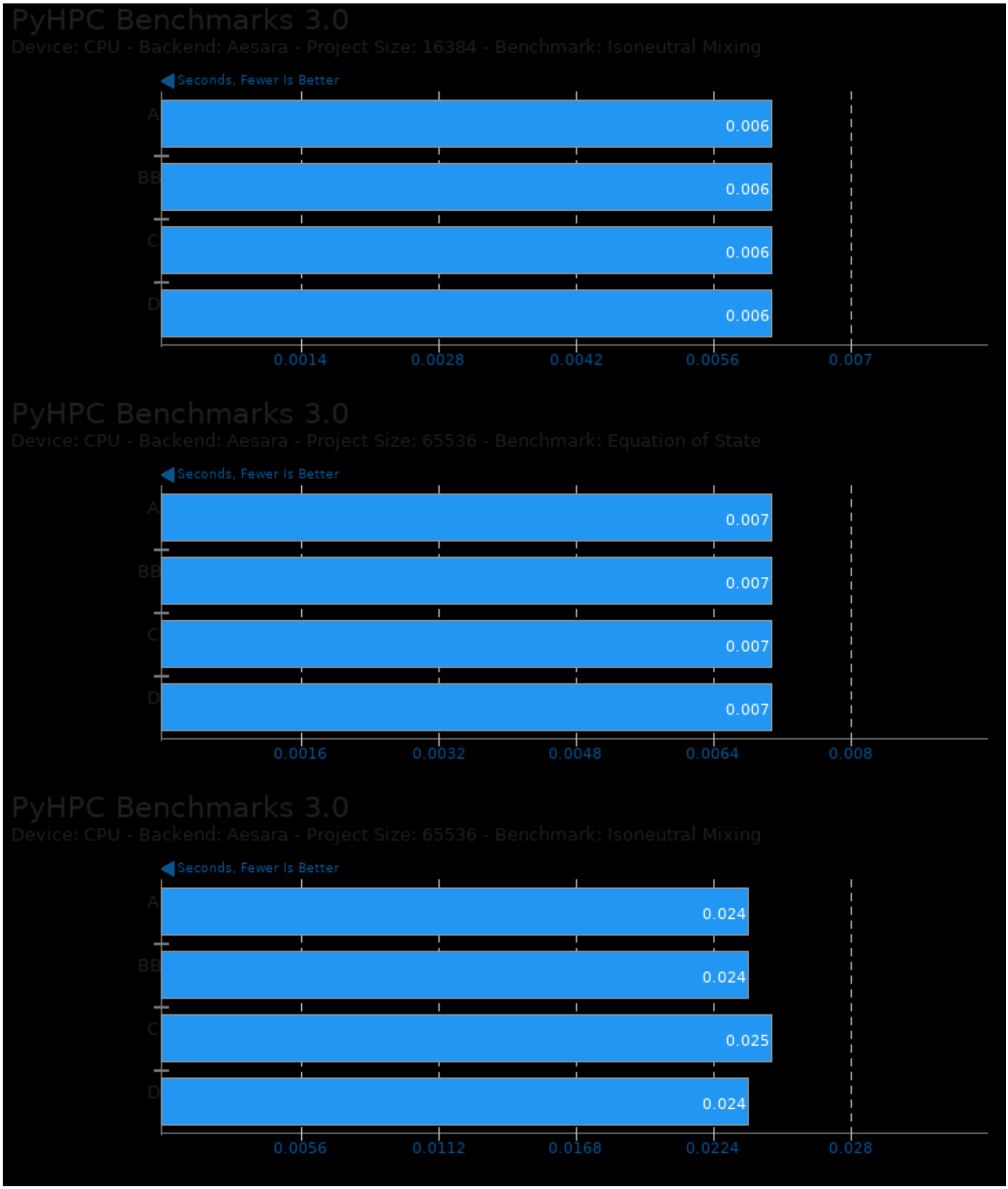
7742 ep

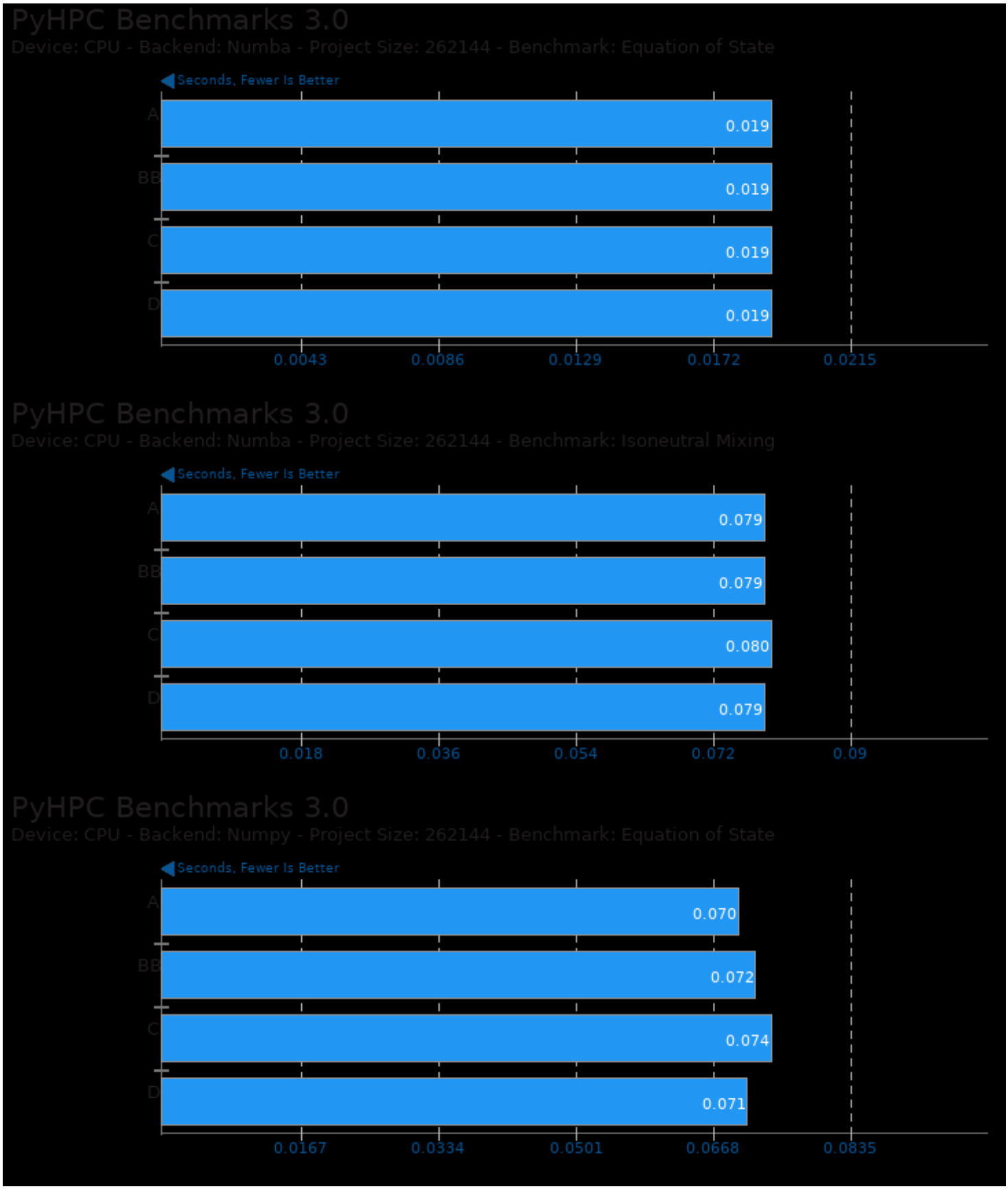

PyHPC Benchmarks 3.0
Device: CPU - Backend: Numba - Project Size: 16384 - Benchmark: Equation of State
Seconds, Fewer Is Better
A
BB
C
D
0.001
0.001
0.001
0.001
0.0002
0.0004
0.0006
0.0008
0.001
PyHPC Benchmarks 3.0
Device: CPU - Backend: Numba - Project Size: 16384 - Benchmark: Isoneutral Mixing
Seconds, Fewer Is Better
A
BB
C
D
0.005
0.005
0.005
0.005
0.0011
0.0022
0.0033
0.0044
0.0055
PyHPC Benchmarks 3.0
Device: CPU - Backend: Numba - Project Size: 65536 - Benchmark: Equation of State
Seconds, Fewer Is Better
A
BB
C
D
0.005
0.005
0.005
0.005
0.0011
0.0022
0.0033
0.0044
0.0055

7742 ep

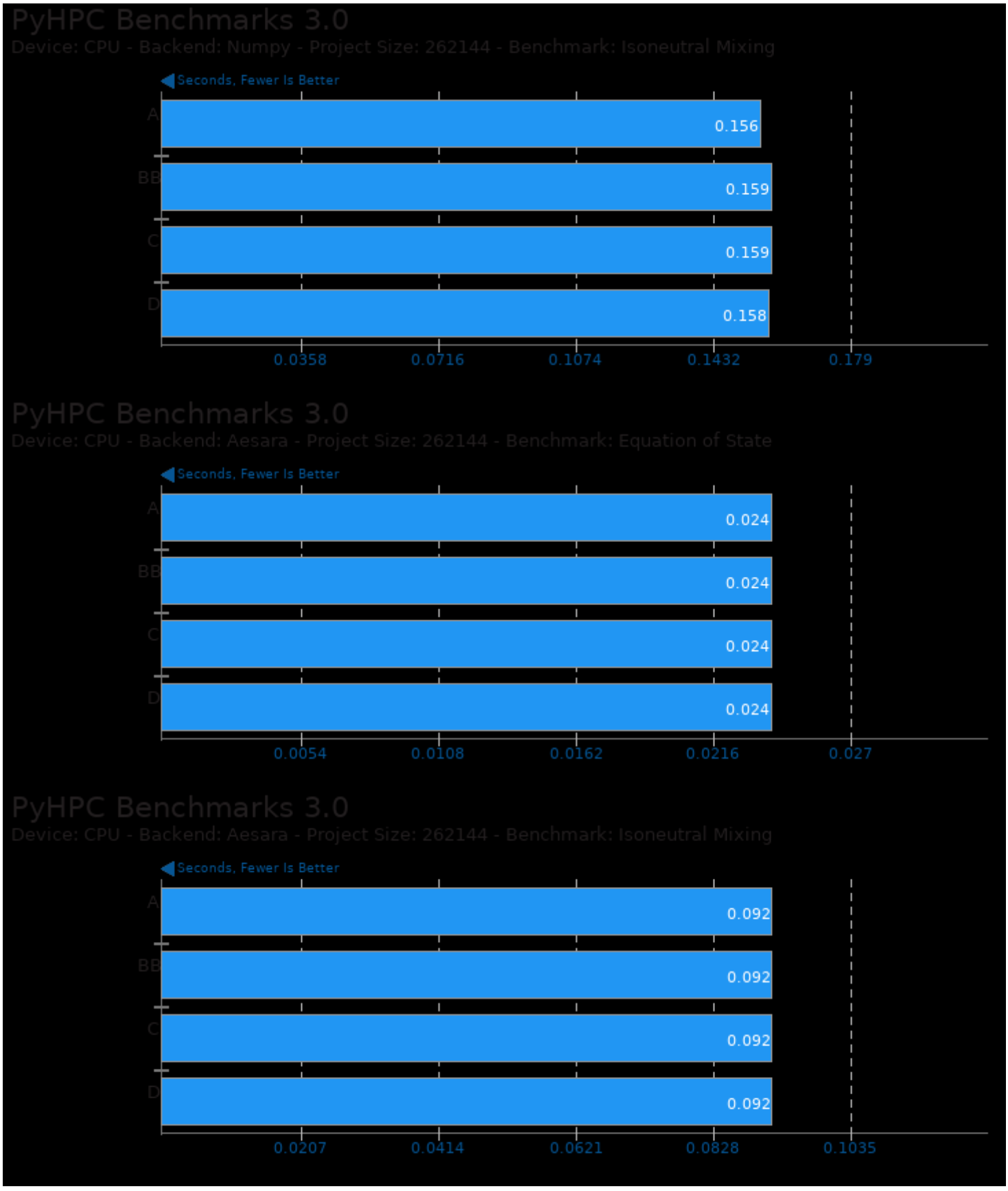

PyHPC Benchmarks 3.0
Device: CPU - Backend: Numba - Project Size: 65536 - Benchmark: Isoneutral Mixing
Seconds, Fewer Is Better
A
BB
C
D
0.021
0.021
0.021
0.021
0.0047
0.0094
0.0141
0.0188
0.0235
PyHPC Benchmarks 3.0
Device: CPU - Backend: Numpy - Project Size: 16384 - Benchmark: Equation of State
Seconds, Fewer Is Better
A
BB
C
D
0.003
0.003
0.003
0.003
0.0007
0.0014
0.0021
0.0028
0.0035
PyHPC Benchmarks 3.0
Device: CPU - Backend: Numpy - Project Size: 16384 - Benchmark: Isoneutral Mixing
Seconds, Fewer Is Better
A
BB
C
D
0.009
0.009
0.009
0.009
0.002
0.004
0.006
0.008
0.01


PyHPC Benchmarks 3.0
Device: CPU - Backend: Numpy - Project Size: 65536 - Benchmark: Equation of State
Seconds, Fewer Is Better
A
BB
C
D
0.018
0.018
0.018
0.018
0.0041
0.0082
0.0123
0.0164
0.0205
PyHPC Benchmarks 3.0
Device: CPU - Backend: Numpy - Project Size: 65536 - Benchmark: Isoneutral Mixing
Seconds, Fewer Is Better
A
BB
C
D
0.039
0.040
0.040
0.040
0.009
0.018
0.027
0.036
0.045
PyHPC Benchmarks 3.0
Device: CPU - Backend: Aesara - Project Size: 16384 - Benchmark: Equation of State
Seconds, Fewer Is Better
A
BB
C
D
0.002
0.002
0.002
0.002
0.0005
0.001
0.0015
0.002
0.0025

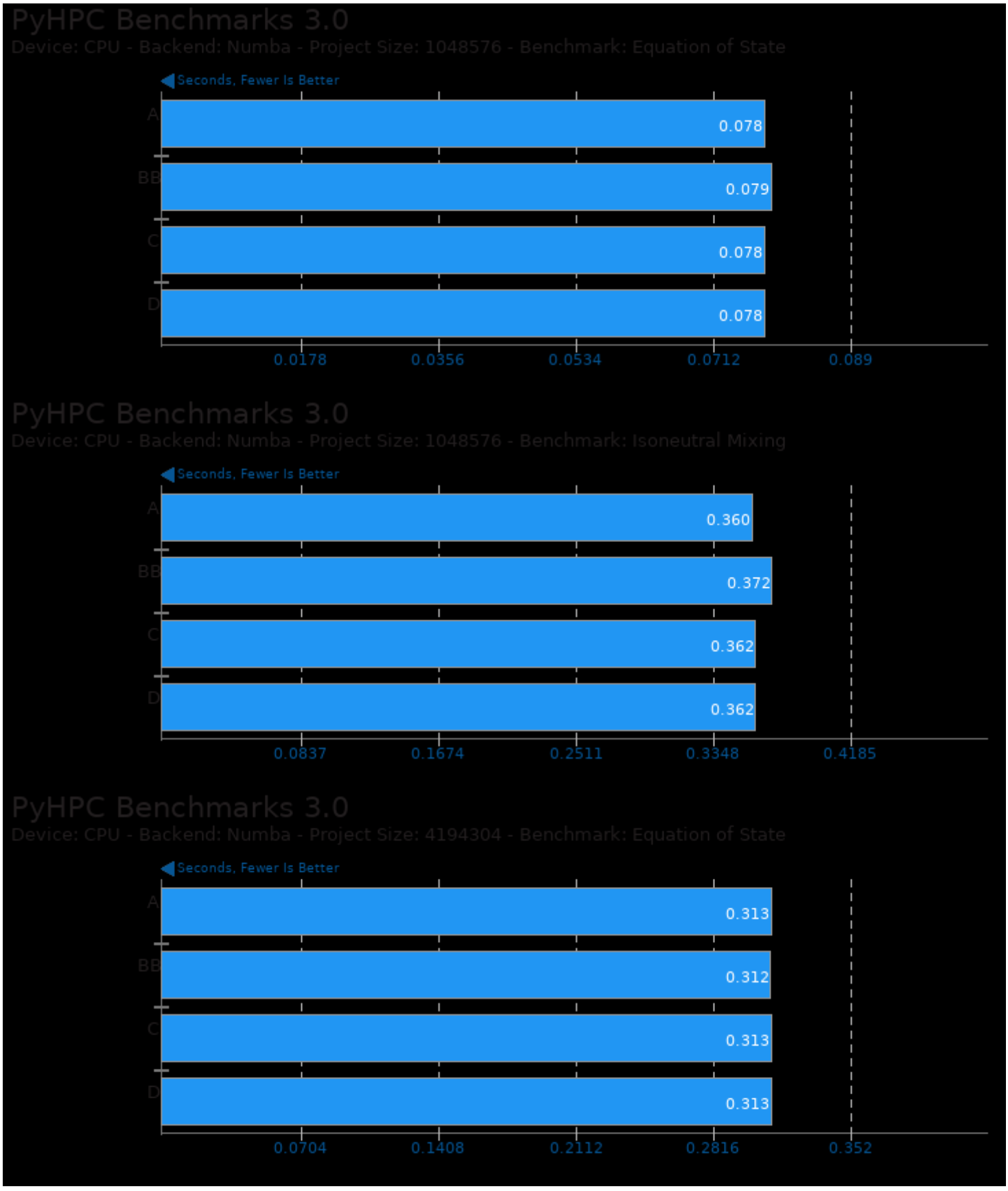

PyHPC Benchmarks 3.0
Device: CPU - Backend: Aesara - Project Size: 16384 - Benchmark: Isoneutral Mixing
Seconds, Fewer Is Better
A
BB
C
D
0.006
0.006
0.006
0.006
0.0014
0.0028
0.0042
0.0056
0.007
PyHPC Benchmarks 3.0
Device: CPU - Backend: Aesara - Project Size: 65536 - Benchmark: Equation of State
Seconds, Fewer Is Better
A
BB
C
D
0.007
0.007
0.007
0.007
0.0016
0.0032
0.0048
0.0064
0.008
PyHPC Benchmarks 3.0
Device: CPU - Backend: Aesara - Project Size: 65536 - Benchmark: Isoneutral Mixing
Seconds, Fewer Is Better
A
BB
C
D
0.024
0.024
0.025
0.024
0.0056
0.0112
0.0168
0.0224
0.028

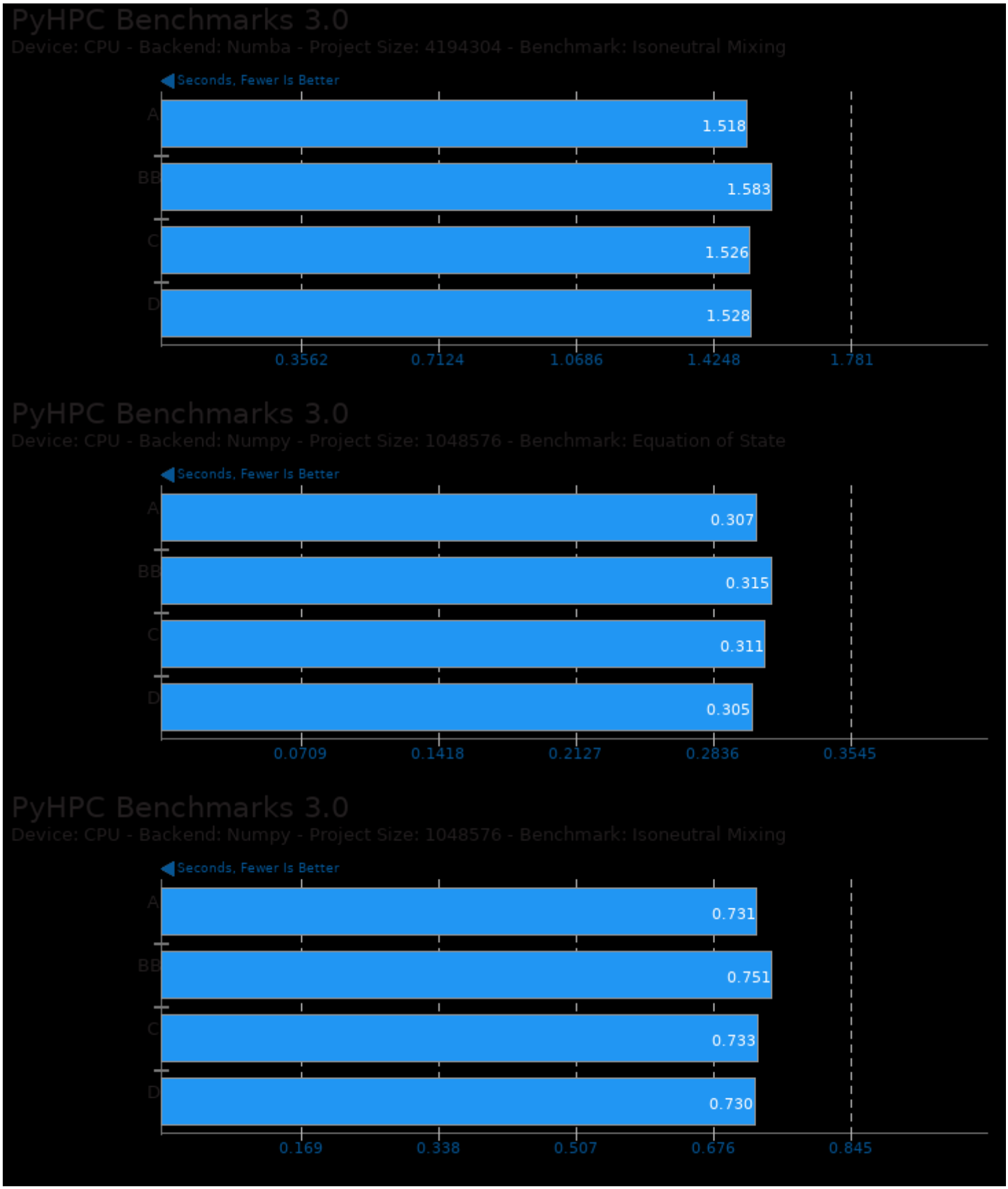

PyHPC Benchmarks 3.0
Device: CPU - Backend: Numba - Project Size: 262144 - Benchmark: Equation of State
Seconds, Fewer Is Better
A
BB
C
D
0.019
0.019
0.019
0.019
0.0043
0.0086
0.0129
0.0172
0.0215
PyHPC Benchmarks 3.0
Device: CPU - Backend: Numba - Project Size: 262144 - Benchmark: Isoneutral Mixing
Seconds, Fewer Is Better
A
BB
C
D
0.079
0.079
0.080
0.079
0.018
0.036
0.054
0.072
0.09
PyHPC Benchmarks 3.0
Device: CPU - Backend: Numpy - Project Size: 262144 - Benchmark: Equation of State
Seconds, Fewer Is Better
A
BB
C
D
0.070
0.072
0.074
0.071
0.0167
0.0334
0.0501
0.0668
0.0835

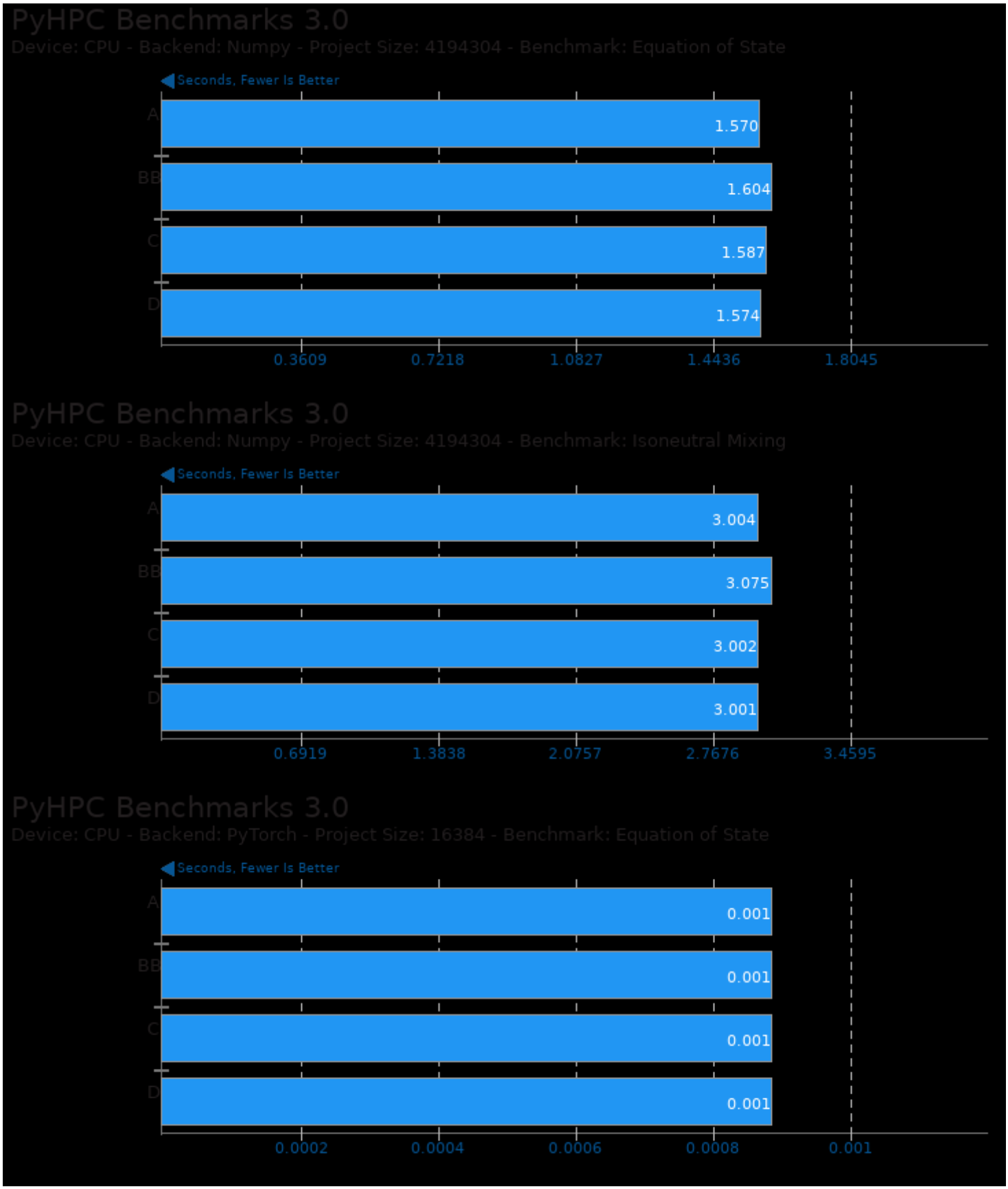
7742 ep


PyHPC Benchmarks 3.0
Device: CPU - Backend: Numpy - Project Size: 262144 - Benchmark: Isoneutral Mixing
Seconds, Fewer Is Better
A
BB
C
D
0.156
0.159
0.159
0.158
0.0358
0.0716
0.1074
0.1432
0.179
PyHPC Benchmarks 3.0
Device: CPU - Backend: Aesara - Project Size: 262144 - Benchmark: Equation of State
Seconds, Fewer Is Better
A
BB
C
D
0.024
0.024
0.024
0.024
0.0054
0.0108
0.0162
0.0216
0.027
PyHPC Benchmarks 3.0
Device: CPU - Backend: Aesara - Project Size: 262144 - Benchmark: Isoneutral Mixing
Seconds, Fewer Is Better
A
BB
C
D
0.092
0.092
0.092
0.092
0.0207
0.0414
0.0621
0.0828
0.1035

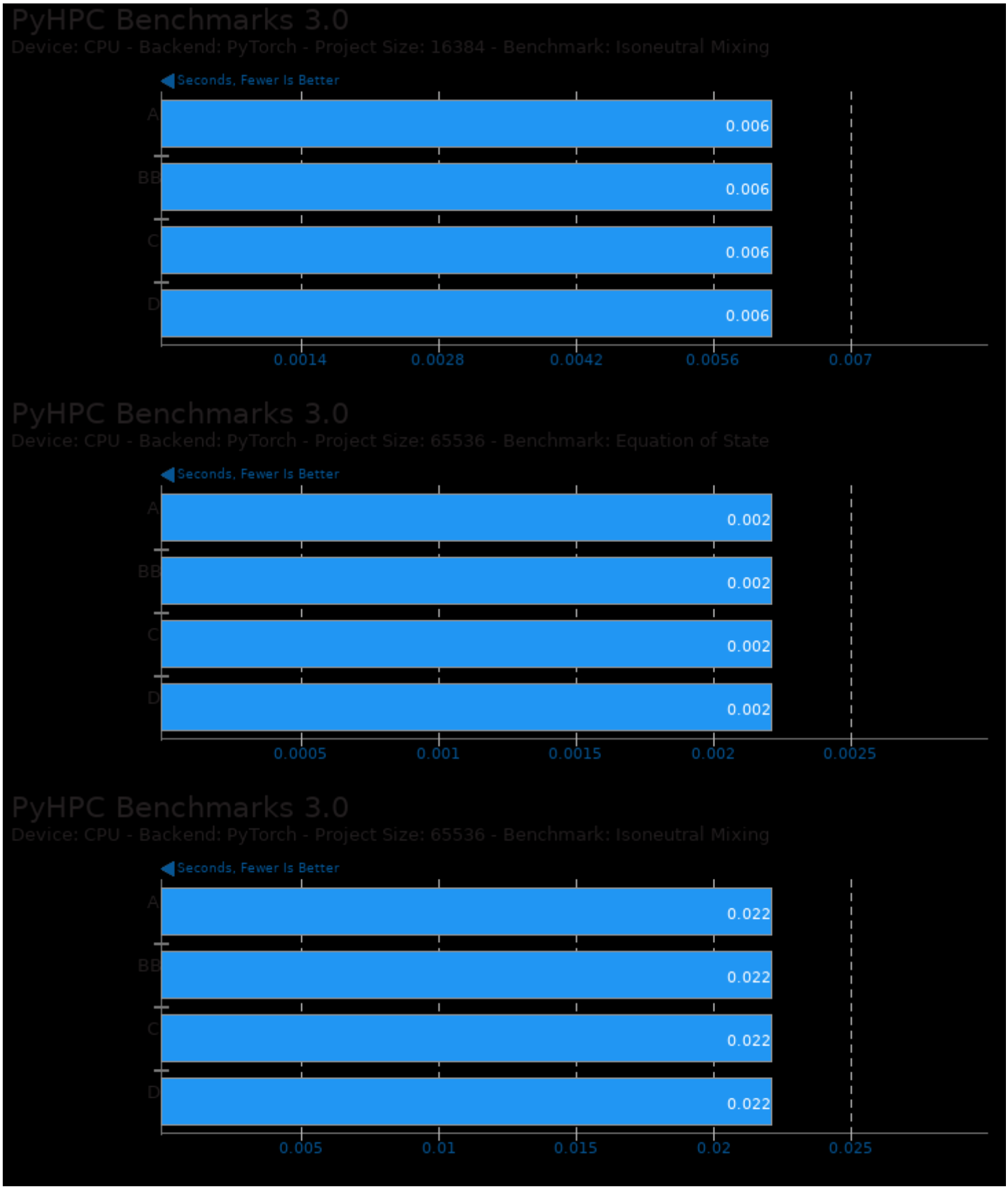
7742 ep

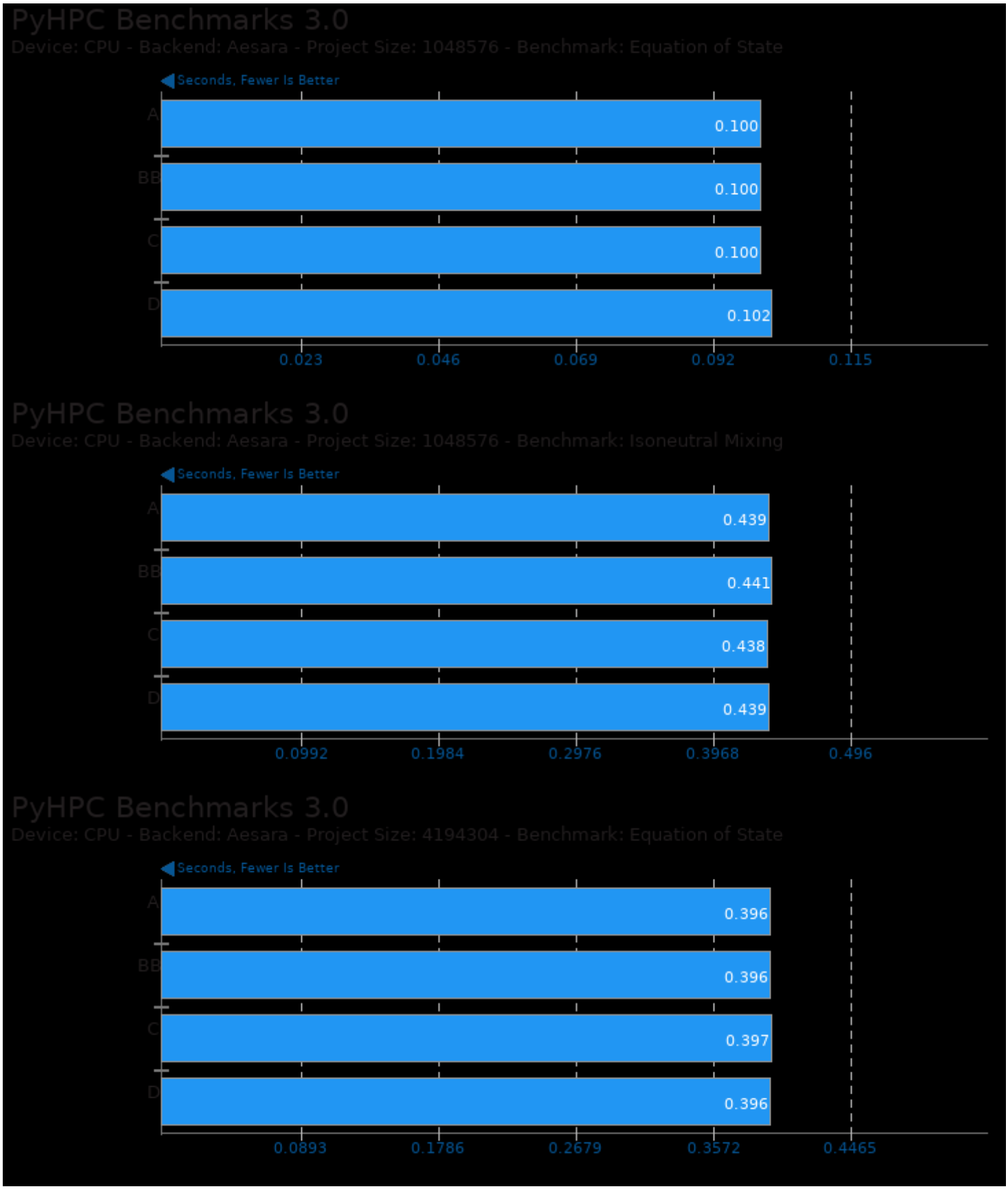

PyHPC Benchmarks 3.0
Device: CPU - Backend: Numba - Project Size: 1048576 - Benchmark: Equation of State
Seconds, Fewer Is Better
A
0.078
BB
0.079
C
0.078
D
0.078
0.0178
0.0356
0.0534
0.0712
0.089
PyHPC Benchmarks 3.0
Device: CPU - Backend: Numba - Project Size: 1048576 - Benchmark: Isoneutral Mixing
Seconds, Fewer Is Better
A
0.360
BB
0.372
C
0.362
D
0.362
0.0837
0.1674
0.2511
0.3348
0.4185
PyHPC Benchmarks 3.0
Device: CPU - Backend: Numba - Project Size: 4194304 - Benchmark: Equation of State
Seconds, Fewer Is Better
A
0.313
BB
0.312
C
0.313
D
0.313
0.0704
0.1408
0.2112
0.2816
0.352

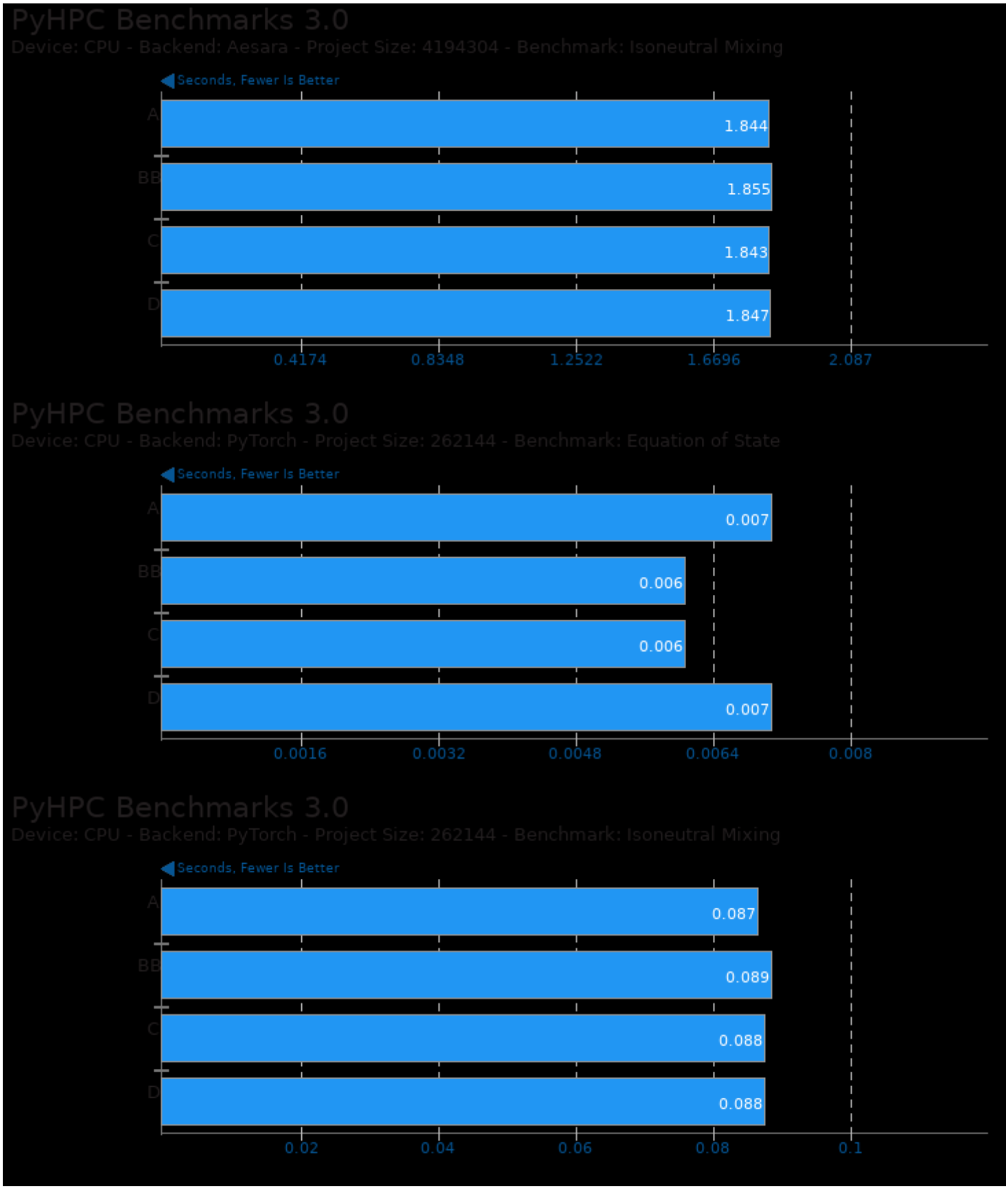

PyHPC Benchmarks 3.0
Device: CPU - Backend: Numba - Project Size: 4194304 - Benchmark: Isoneutral Mixing
Seconds, Fewer Is Better
A
BB
C
D
1.518
1.583
1.526
1.528
0.3562
0.7124
1.0686
1.4248
1.781
PyHPC Benchmarks 3.0
Device: CPU - Backend: Numpy - Project Size: 1048576 - Benchmark: Equation of State
Seconds, Fewer Is Better
A
BB
C
D
0.307
0.315
0.311
0.305
0.0709
0.1418
0.2127
0.2836
0.3545
PyHPC Benchmarks 3.0
Device: CPU - Backend: Numpy - Project Size: 1048576 - Benchmark: Isoneutral Mixing
Seconds, Fewer Is Better
A
BB
C
D
0.731
0.751
0.733
0.730
0.169
0.338
0.507
0.676
0.845

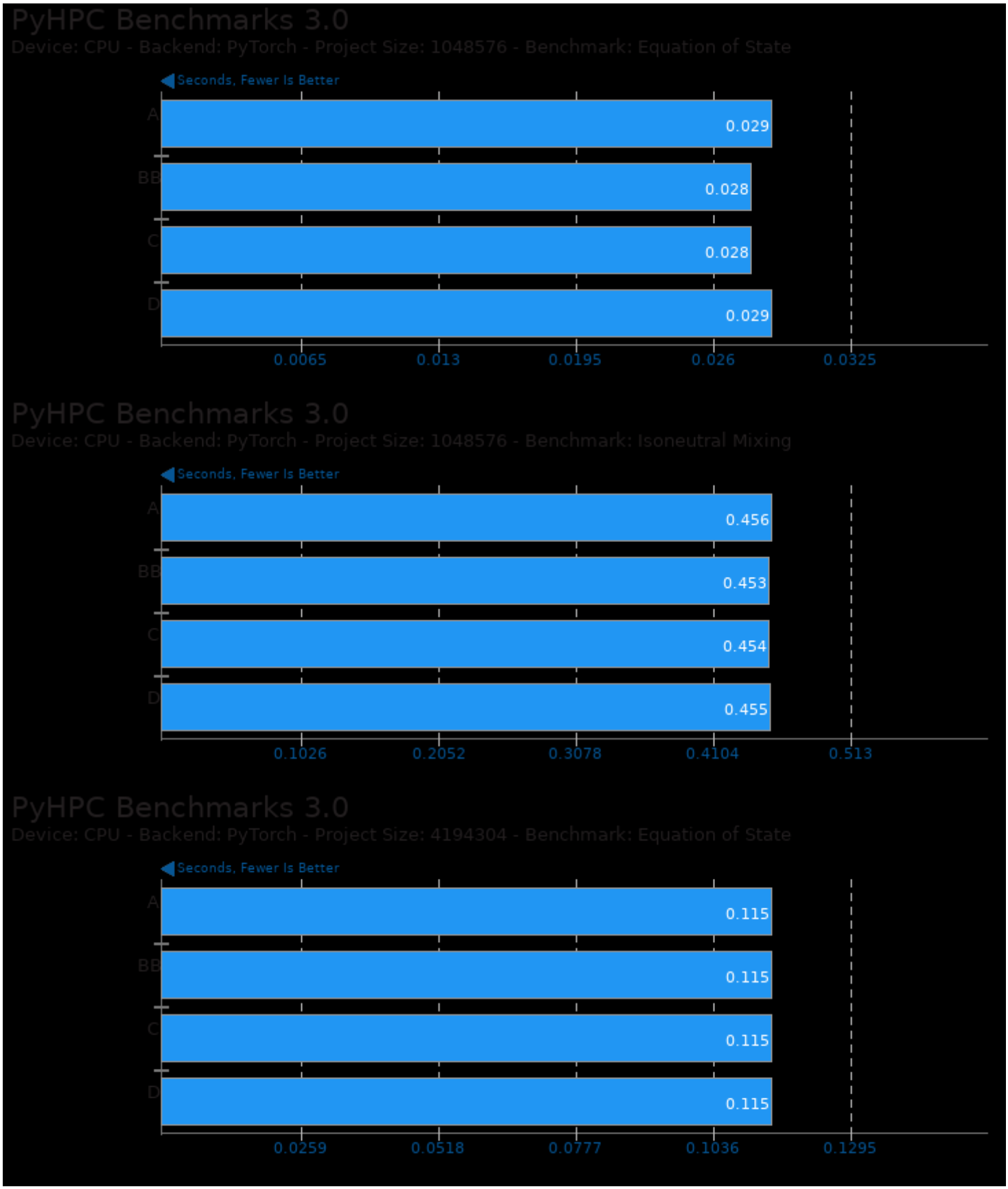

PyHPC Benchmarks 3.0
Device: CPU - Backend: Numpy - Project Size: 4194304 - Benchmark: Equation of State
Seconds, Fewer Is Better
A
BB
C
D
1.570
1.604
1.587
1.574
0.3609
0.7218
1.0827
1.4436
1.8045
PyHPC Benchmarks 3.0
Device: CPU - Backend: Numpy - Project Size: 4194304 - Benchmark: Isoneutral Mixing
Seconds, Fewer Is Better
A
BB
C
D
3.004
3.075
3.002
3.001
0.6919
1.3838
2.0757
2.7676
3.4595
PyHPC Benchmarks 3.0
Device: CPU - Backend: PyTorch - Project Size: 16384 - Benchmark: Equation of State
Seconds, Fewer Is Better
A
BB
C
D
0.001
0.001
0.001
0.001
0.0002
0.0004
0.0006
0.0008
0.001

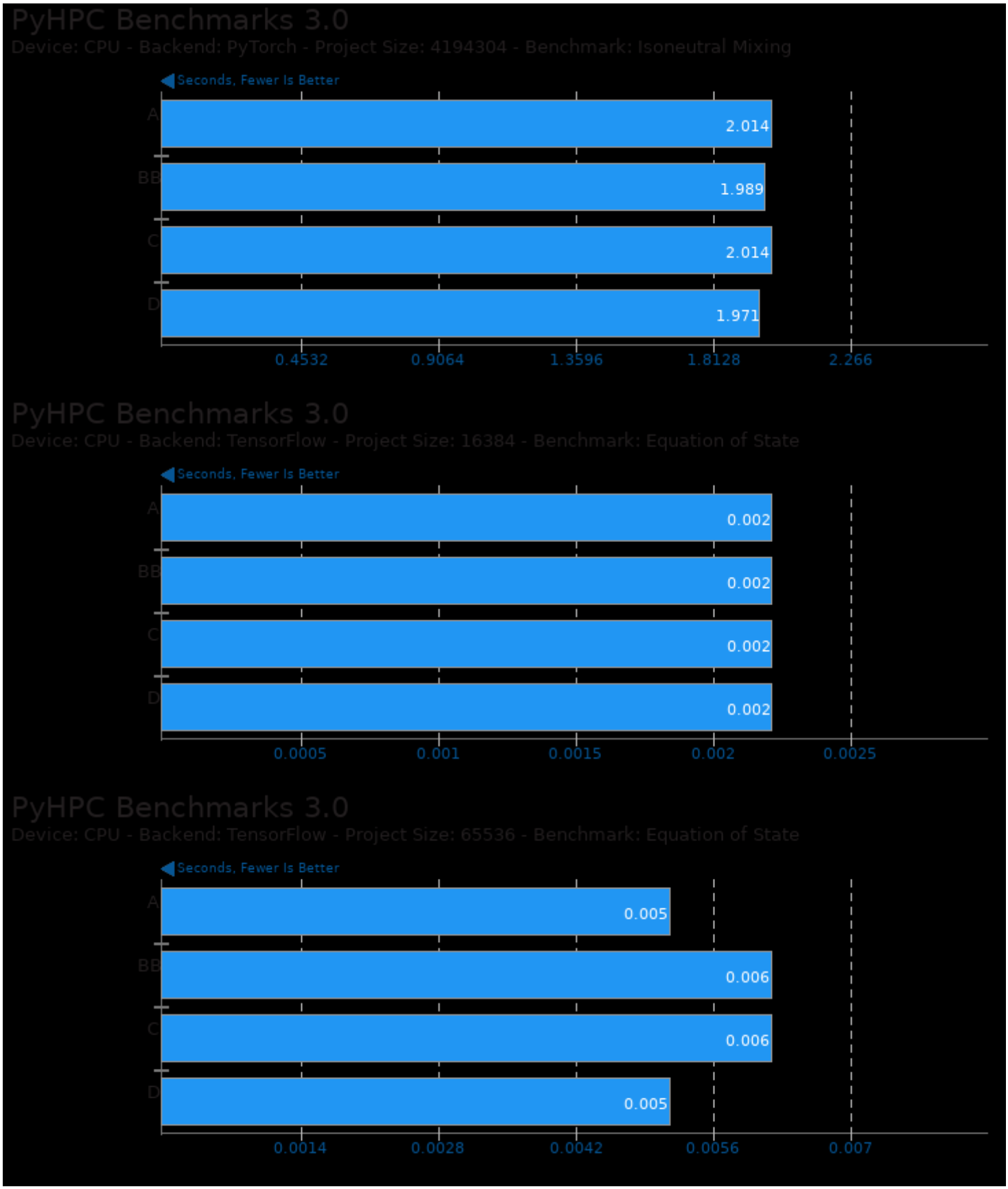
7742 ep

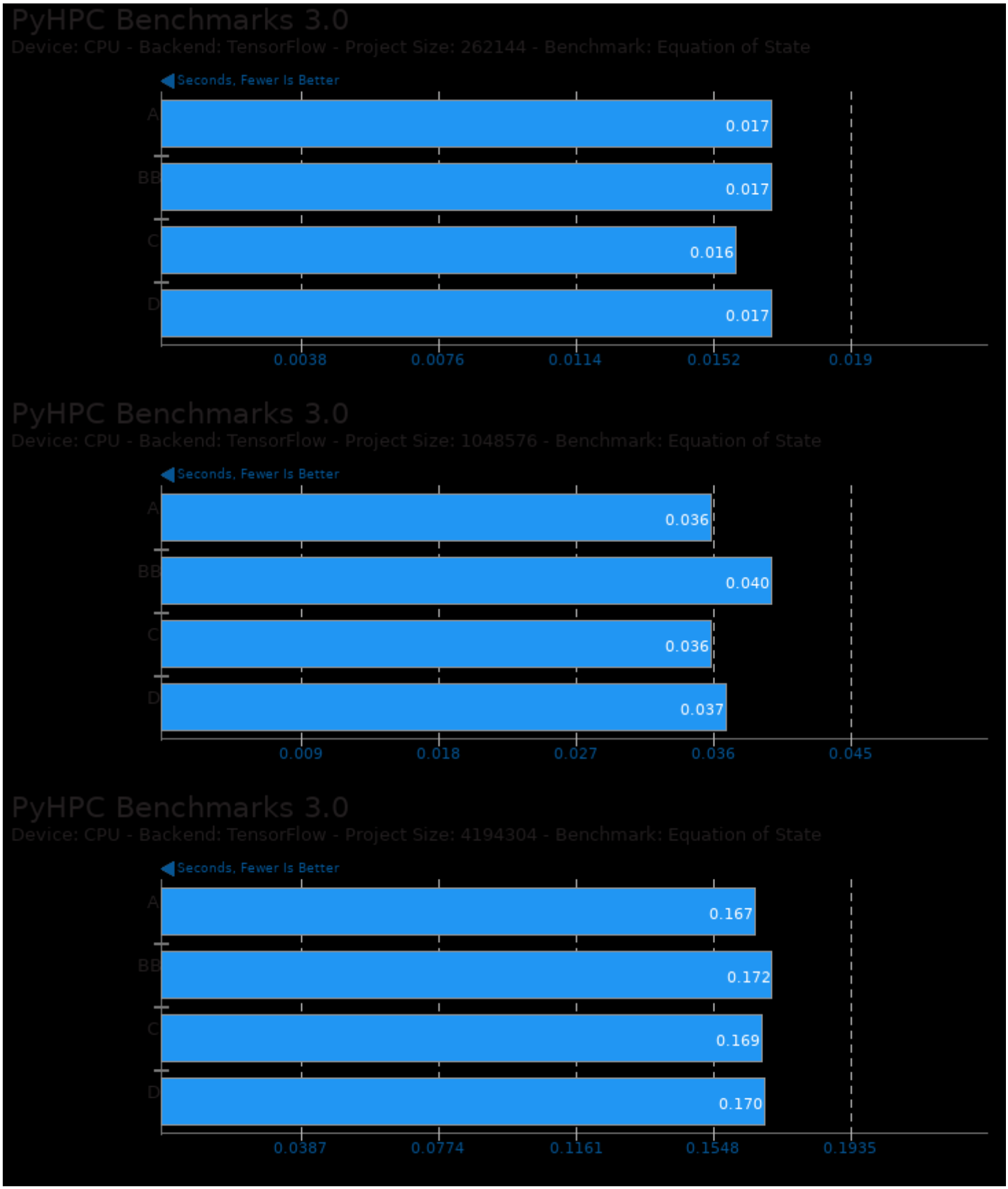

PyHPC Benchmarks 3.0
Device: CPU - Backend: PyTorch - Project Size: 16384 - Benchmark: Isoneutral Mixing
Seconds, Fewer Is Better
A
BB
C
D
0.006
0.006
0.006
0.006
0.0014
0.0028
0.0042
0.0056
0.007
PyHPC Benchmarks 3.0
Device: CPU - Backend: PyTorch - Project Size: 65536 - Benchmark: Equation of State
Seconds, Fewer Is Better
A
BB
C
D
0.002
0.002
0.002
0.002
0.0005
0.001
0.0015
0.002
0.0025
PyHPC Benchmarks 3.0
Device: CPU - Backend: PyTorch - Project Size: 65536 - Benchmark: Isoneutral Mixing
Seconds, Fewer Is Better
A
BB
C
D
0.022
0.022
0.022
0.022
0.005
0.01
0.015
0.02
0.025


PyHPC Benchmarks 3.0
Device: CPU - Backend: Aesara - Project Size: 1048576 - Benchmark: Equation of State
Seconds, Fewer Is Better
A
BB
C
D
0.100
0.100
0.100
0.102
0.023
0.046
0.069
0.092
0.115
PyHPC Benchmarks 3.0
Device: CPU - Backend: Aesara - Project Size: 1048576 - Benchmark: Isoneutral Mixing
Seconds, Fewer Is Better
A
BB
C
D
0.439
0.441
0.438
0.439
0.0992
0.1984
0.2976
0.3968
0.496
PyHPC Benchmarks 3.0
Device: CPU - Backend: Aesara - Project Size: 4194304 - Benchmark: Equation of State
Seconds, Fewer Is Better
A
BB
C
D
0.396
0.396
0.397
0.396
0.0893
0.1786
0.2679
0.3572
0.4465


PyHPC Benchmarks 3.0
Device: CPU - Backend: Aesara - Project Size: 4194304 - Benchmark: Isoneutral Mixing
Seconds, Fewer Is Better
A
BB
C
D
1.844
1.855
1.843
1.847
0.4174
0.8348
1.2522
1.6696
2.087
PyHPC Benchmarks 3.0
Device: CPU - Backend: PyTorch - Project Size: 262144 - Benchmark: Equation of State
Seconds, Fewer Is Better
A
BB
C
D
0.007
0.006
0.006
0.007
0.0016
0.0032
0.0048
0.0064
0.008
PyHPC Benchmarks 3.0
Device: CPU - Backend: PyTorch - Project Size: 262144 - Benchmark: Isoneutral Mixing
Seconds, Fewer Is Better
A
BB
C
D
0.087
0.089
0.088
0.088
0.02
0.04
0.06
0.08
0.1

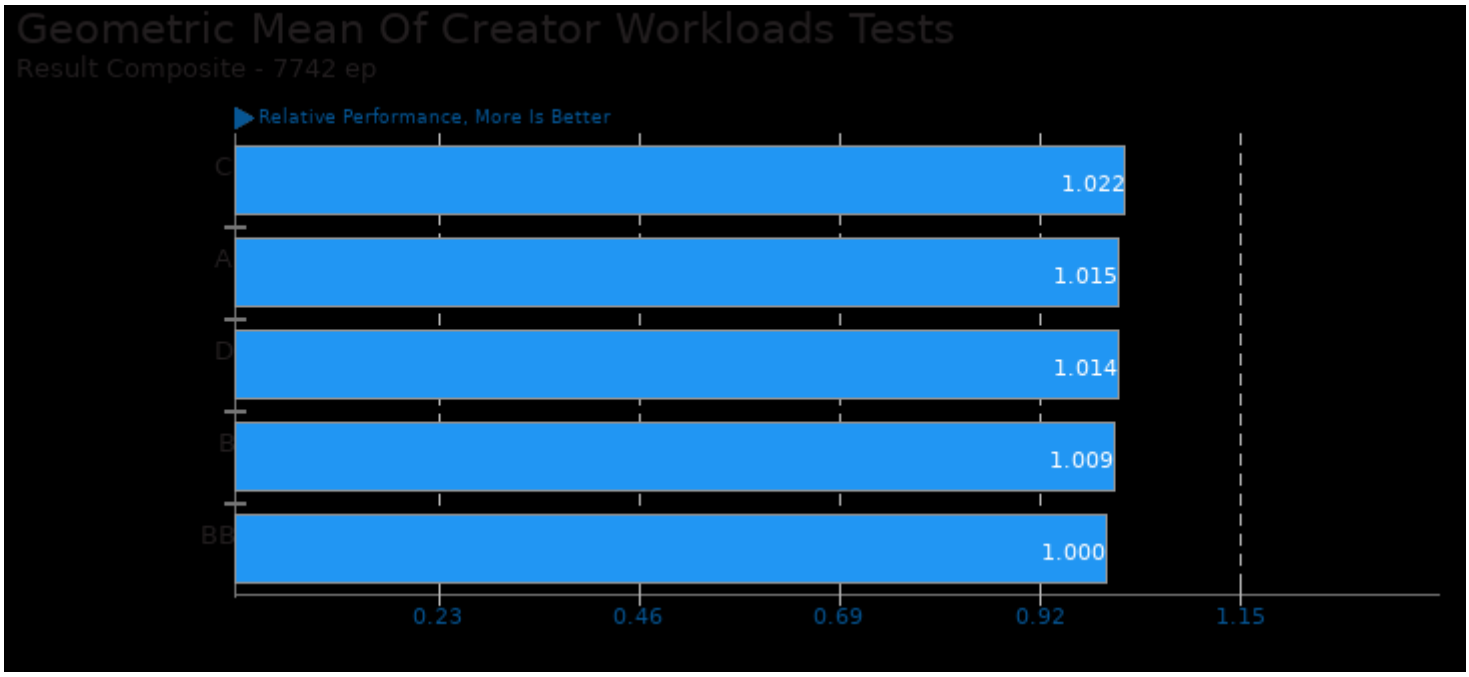

PyHPC Benchmarks 3.0
Device: CPU - Backend: PyTorch - Project Size: 1048576 - Benchmark: Equation of State
Seconds, Fewer Is Better
A
BB
C
D
0.029
0.028
0.028
0.029
0.0065
0.013
0.0195
0.026
0.0325
PyHPC Benchmarks 3.0
Device: CPU - Backend: PyTorch - Project Size: 1048576 - Benchmark: Isoneutral Mixing
Seconds, Fewer Is Better
A
BB
C
D
0.456
0.453
0.454
0.455
0.1026
0.2052
0.3078
0.4104
0.513
PyHPC Benchmarks 3.0
Device: CPU - Backend: PyTorch - Project Size: 4194304 - Benchmark: Equation of State
Seconds, Fewer Is Better
A
BB
C
D
0.115
0.115
0.115
0.115
0.0259
0.0518
0.0777
0.1036
0.1295


PyHPC Benchmarks 3.0
Device: CPU - Backend: PyTorch - Project Size: 4194304 - Benchmark: Isoneutral Mixing
Seconds, Fewer Is Better
A
BB
C
D
2.014
1.989
2.014
1.971
0.4532
0.9064
1.3596
1.8128
2.266
PyHPC Benchmarks 3.0
Device: CPU - Backend: TensorFlow - Project Size: 16384 - Benchmark: Equation of State
Seconds, Fewer Is Better
A
BB
C
D
0.002
0.002
0.002
0.002
0.0005
0.001
0.0015
0.002
0.0025
PyHPC Benchmarks 3.0
Device: CPU - Backend: TensorFlow - Project Size: 65536 - Benchmark: Equation of State
Seconds, Fewer Is Better
A
BB
C
D
0.005
0.006
0.006
0.005
0.0014
0.0028
0.0042
0.0056
0.007

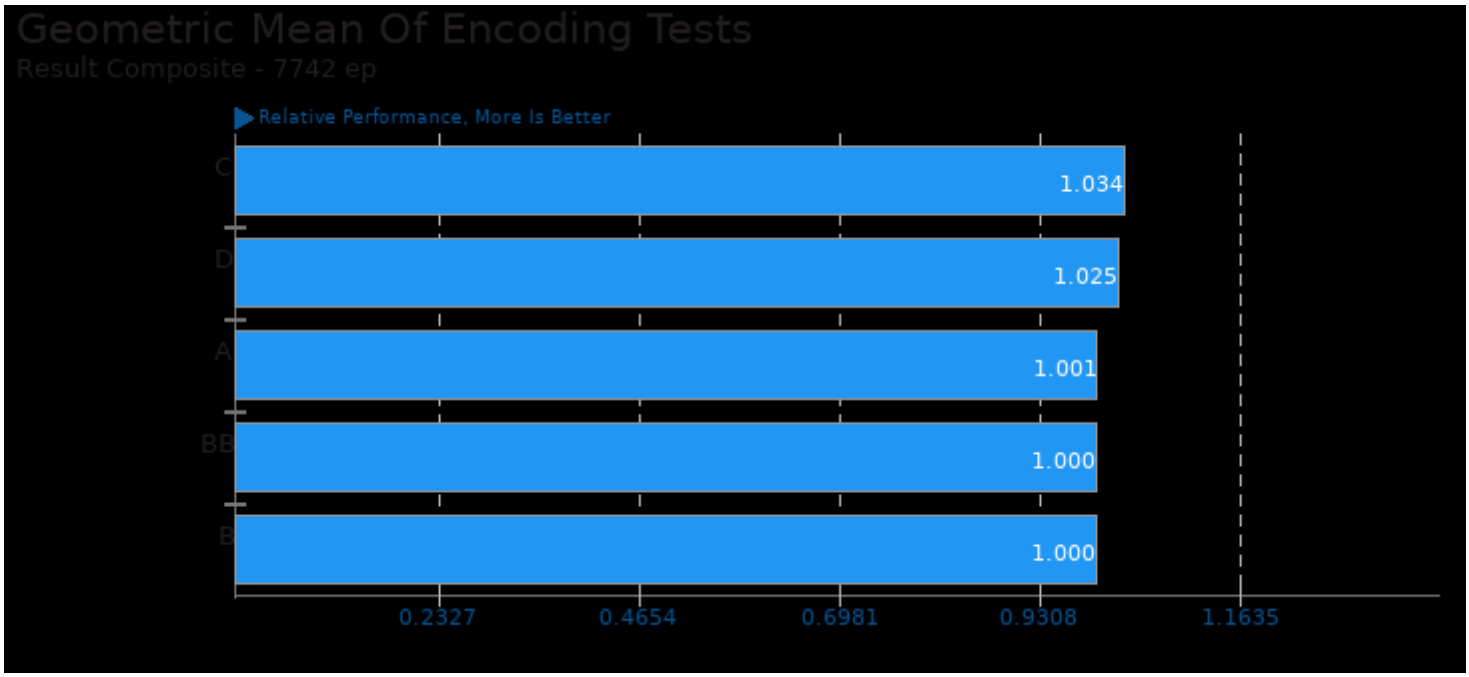

PyHPC Benchmarks 3.0
Device: CPU - Backend: TensorFlow - Project Size: 262144 - Benchmark: Equation of State
Seconds, Fewer Is Better
A
BB
C
D
0.017
0.017
0.016
0.017
0.0038
0.0076
0.0114
0.0152
0.019
PyHPC Benchmarks 3.0
Device: CPU - Backend: TensorFlow - Project Size: 1048576 - Benchmark: Equation of State
Seconds, Fewer Is Better
A
BB
C
D
0.036
0.040
0.036
0.037
0.009
0.018
0.027
0.036
0.045
PyHPC Benchmarks 3.0
Device: CPU - Backend: TensorFlow - Project Size: 4194304 - Benchmark: Equation of State
Seconds, Fewer Is Better
A
BB
C
D
0.167
0.172
0.169
0.170
0.0387
0.0774
0.1161
0.1548
0.1935

7742 ep

These geometric means are based upon test groupings / test suites for this result file.

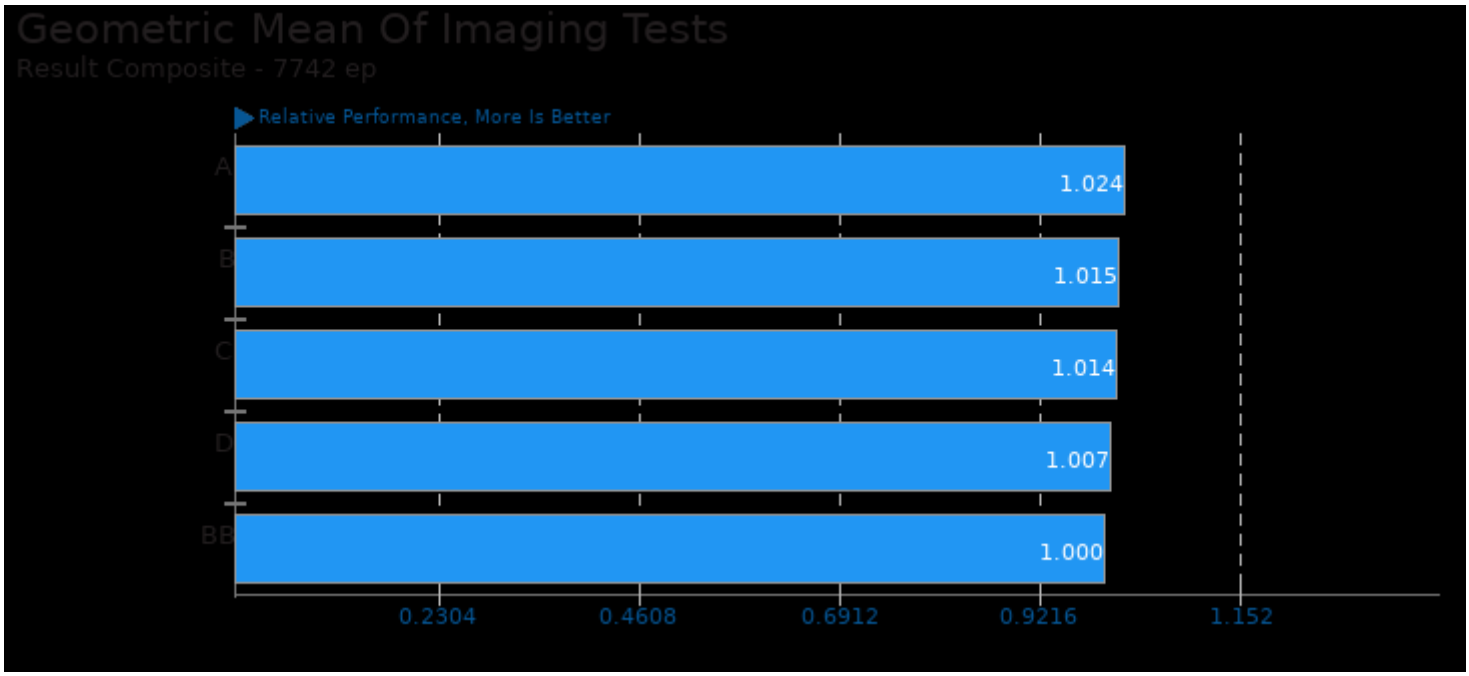


Geometric mean based upon tests: pts/blender, pts/rav1e, pts/stargate, pts/jpegxl and pts/jpegxl-decode

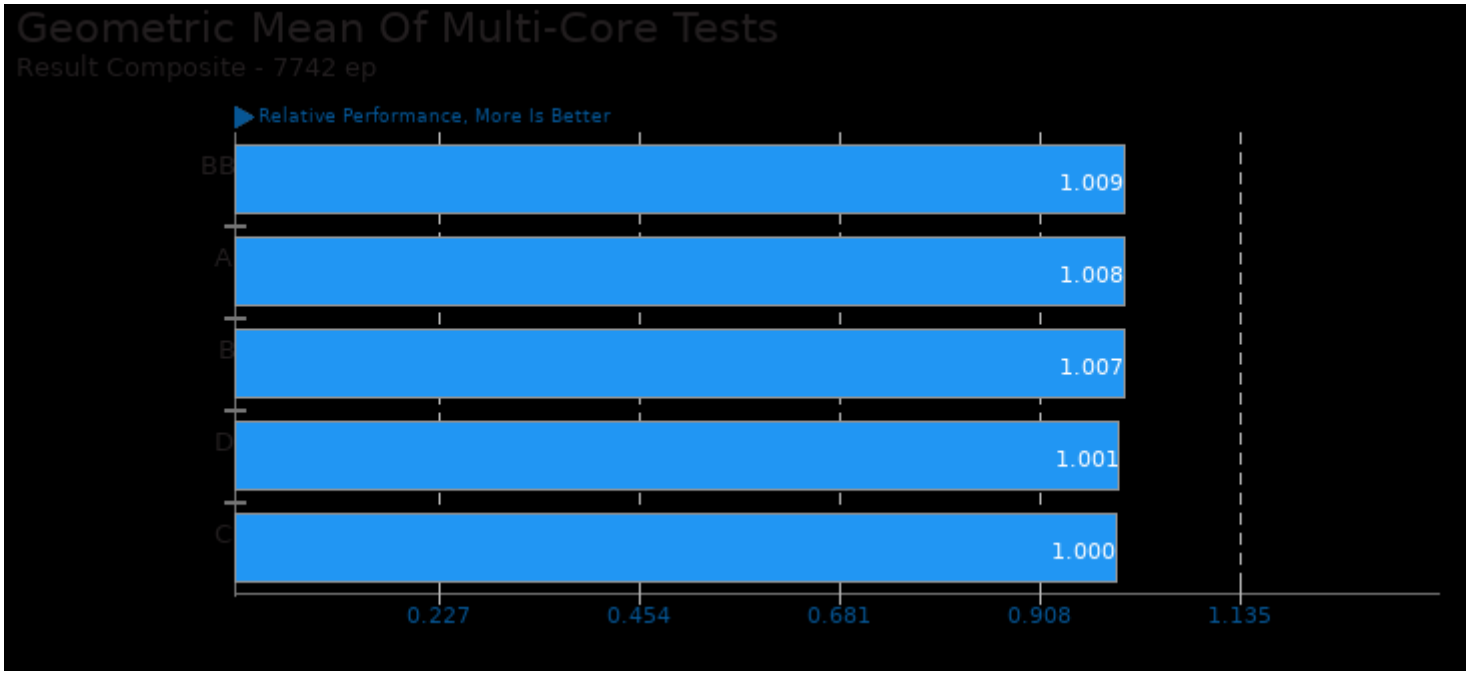


Geometric mean based upon tests: pts/stargate and pts/rav1e

7742 ep



Geometric mean based upon tests: pts/jpegxl and pts/jpegxl-decode



Geometric mean based upon tests: pts/blender and pts/rav1e

This file was automatically generated via the Phoronix Test Suite benchmarking software on Thursday, 28 March 2024 20:28.