SPEC® CINT2006 Result

Copyright 2006-2014 Standard Performance Evaluation Corporation

NEC Corporation

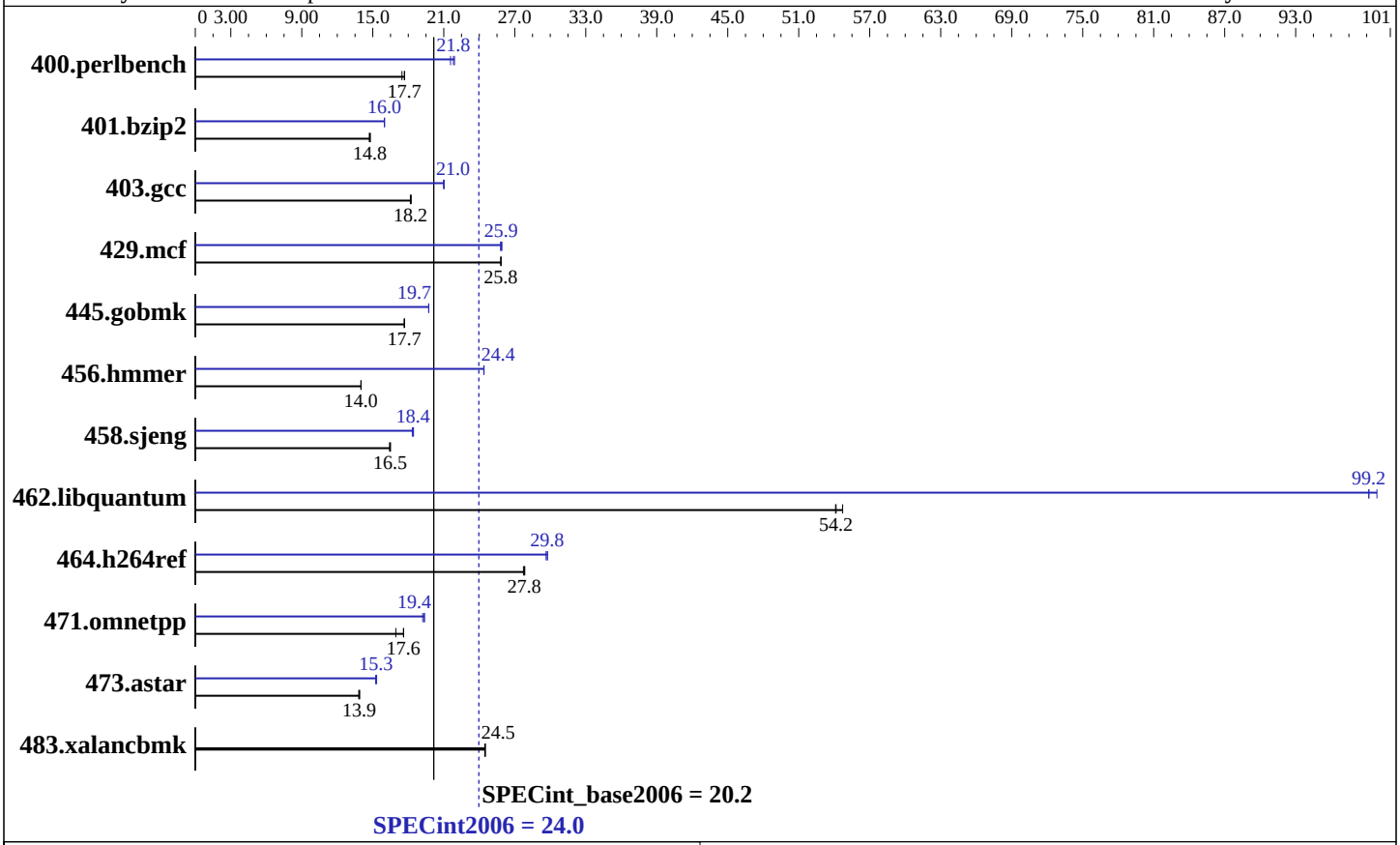
Express5800/110Ri-1
(Intel Xeon X3350)

SPECint®2006 = 24.0

SPECint_base2006 = 20.2

CPU2006 license: 9006
Test sponsor: NEC Corporation
Tested by: NEC Corporation

Test date: Jul-2008
Hardware Availability: Apr-2008
Software Availability: Nov-2007



Hardware

CPU Name:	Intel Xeon X3350
CPU Characteristics:	2.66 GHz, 2x6 MB L2 shared, 1333 MHz bus
CPU MHz:	2667
FPU:	Integrated
CPU(s) enabled:	4 cores, 1 chip, 4 cores/chip
CPU(s) orderable:	1 chip
Primary Cache:	32 KB I + 32 KB D on chip per core
Secondary Cache:	12 MB I+D on chip per chip, 6 MB shared / 2 cores
L3 Cache:	None
Other Cache:	None
Memory:	8 GB (4x2 GB PC2-6400E, 2 rank, CL6-6-6, ECC)
Disk Subsystem:	1x80.0 GB SATAII, 7200RPM
Other Hardware:	None

Software

Operating System:	SUSE Linux Enterprise Server 10 (x86_64) SP1, Kernel 2.6.16.46-0.12-smp
Compiler:	Intel C++ Compiler for Linux32 and Linux64 version 10.1 Build 20070913 Package ID: l_cc_p_10.1.008
Auto Parallel:	Yes
File System:	ReiserFS
System State:	Run level 3 (multi-user)
Base Pointers:	32-bit
Peak Pointers:	32/64-bit
Other Software:	MicroQuill SmartHeap library 8.1 binutils-2.17.tar.gz, Version 2.17

SPEC CINT2006 Result

Copyright 2006-2014 Standard Performance Evaluation Corporation

NEC Corporation

Express5800/110Ri-1
(Intel Xeon X3350)

SPECint2006 = 24.0

SPECint_base2006 = 20.2

CPU2006 license: 9006
Test sponsor: NEC Corporation
Tested by: NEC Corporation

Test date: Jul-2008
Hardware Availability: Apr-2008
Software Availability: Nov-2007

Results Table

Benchmark	Base						Peak					
	Seconds	Ratio	Seconds	Ratio	Seconds	Ratio	Seconds	Ratio	Seconds	Ratio	Seconds	Ratio
400.perlbench	**553**	**17.7**	559	17.5	552	17.7	**448**	**21.8**	453	21.6	446	21.9
401.bzip2	**654**	**14.8**	656	14.7	652	14.8	604	16.0	**604**	**16.0**	603	16.0
403.gcc	441	18.2	443	18.2	**442**	**18.2**	383	21.0	**383**	**21.0**	384	21.0
429.mcf	**353**	**25.8**	353	25.8	353	25.8	354	25.8	**353**	**25.9**	352	25.9
445.gobmk	**594**	**17.7**	594	17.7	594	17.7	**532**	**19.7**	532	19.7	532	19.7
456.hmmer	666	14.0	666	14.0	**666**	**14.0**	**383**	**24.4**	382	24.4	383	24.4
458.sjeng	737	16.4	733	16.5	**735**	**16.5**	656	18.4	660	18.3	**657**	**18.4**
462.libquantum	**383**	**54.2**	379	54.7	383	54.1	207	99.9	**209**	**99.2**	209	99.2
464.h264ref	795	27.8	**796**	**27.8**	798	27.7	744	29.8	747	29.6	**744**	**29.8**
471.omnetpp	369	16.9	355	17.6	**355**	**17.6**	323	19.4	325	19.2	**323**	**19.4**
473.astar	508	13.8	**506**	**13.9**	505	13.9	**460**	**15.3**	460	15.3	458	15.3
483.xalancbmk	282	24.5	282	24.5	**282**	**24.5**	282	24.5	282	24.5	**282**	**24.5**

Results appear in the order in which they were run. Bold underlined text indicates a median measurement.

Operating System Notes

```
'ulimit -s unlimited' was used to set the stacksize to unlimited prior to run
 OMP_NUM_THREADS set to number of cores
```

Platform Notes

```
Bios settings:
Hardware Prefetcher:          Enabled
Adjacent Cache Line Prefetch: Enabled
```

General Notes

```
All benchmarks compiled in 32-bit mode except 401.bzip2 and 456.hmmer,
for peak, are compiled in 64-bit mode
```

Base Compiler Invocation

C benchmarks:
icc

C++ benchmarks:
icpc

Base Portability Flags

400.perlbench: -DSPEC_CPU_LINUX_IA32
462.libquantum: -DSPEC_CPU_LINUX
483.xalancbmk: -DSPEC_CPU_LINUX

Base Optimization Flags

C benchmarks:
-fast -vec-guard-write -parallel -par-runtime-control

C++ benchmarks:
-xT -ipo -O3 -no-prec-div -Wl,-z,muldefs
-L/opt/SmartHeap_8.1/lib -lsmartheap

Base Other Flags

C benchmarks:

403.gcc: -Dalloca=_alloca

Peak Compiler Invocation

C benchmarks (except as noted below):
icc

401.bzip2: /opt/intel/cce/10.1.008/bin/icc
-L/opt/intel/cce/10.1.008/lib
-I/opt/intel/cce/10.1.008/include

456.hmmer: /opt/intel/cce/10.1.008/bin/icc
-L/opt/intel/cce/10.1.008/lib
-I/opt/intel/cce/10.1.008/include

C++ benchmarks:
icpc

Peak Portability Flags

400.perlbench: -DSPEC_CPU_LINUX_IA32
401.bzip2: -DSPEC_CPU_LP64
456.hmmer: -DSPEC_CPU_LP64
462.libquantum: -DSPEC_CPU_LINUX

Continued on next page

Peak Portability Flags (Continued)

483.xalancbmk: -DSPEC_CPU_LINUX

Peak Optimization Flags

C benchmarks:

400.perlbench: -prof-gen(pass 1) -prof-use(pass 2) -fast -ansi-alias -prefetch

401.bzip2: -prof-gen(pass 1) -prof-use(pass 2) -fast -prefetch -auto-ilp32

403.gcc: -fast -inline-calloc -opt-malloc-options=3

429.mcf: -fast -prefetch

445.gobmk: -prof-gen(pass 1) -prof-use(pass 2) -xT -O2 -ipo -no-prec-div -ansi-alias

456.hmmer: -fast -unroll2 -ansi-alias -opt-multi-version-aggressive -auto-ilp32

458.sjeng: -prof-gen(pass 1) -prof-use(pass 2) -fast -unroll4

462.libquantum: -fast -unroll4 -Ob0 -prefetch -opt-streaming-stores always -vec-guard-write -opt-malloc-options=3 -parallel -par-runtime-control

464.h264ref: -prof-gen(pass 1) -prof-use(pass 2) -fast -unroll2 -ansi-alias

C++ benchmarks:

471.omnetpp: -prof-gen(pass 1) -prof-use(pass 2) -xT -O3 -ipo -no-prec-div -ansi-alias -opt-ra-region-strategy=block -Wl,-z,muldefs -L/opt/SmartHeap_8.1/lib -lsmartheap

473.astar: -prof-gen(pass 1) -prof-use(pass 2) -xT -O3 -ipo -no-prec-div -ansi-alias -opt-ra-region-strategy=routine -Wl,-z,muldefs -L/opt/SmartHeap_8.1/lib -lsmartheap

483.xalancbmk: basepeak = yes

Peak Other Flags

C benchmarks:

403.gcc: `-Dalloca=_alloca`

The flags file that was used to format this result can be browsed at
`http://www.spec.org/cpu2006/flags/NEC-Intel-ic10.1-INT-ia32-linux-flags.html`

You can also download the XML flags source by saving the following link:
`http://www.spec.org/cpu2006/flags/NEC-Intel-ic10.1-INT-ia32-linux-flags.xml`

SPEC and SPECint are registered trademarks of the Standard Performance Evaluation Corporation. All other brand and product names appearing in this result are trademarks or registered trademarks of their respective holders.

For questions about this result, please contact the tester.
For other inquiries, please contact webmaster@spec.org.

Tested with SPEC CPU2006 v1.0.
Report generated on Tue Jul 22 20:02:04 2014 by SPEC CPU2006 PS/PDF formatter v6932.
Originally published on 22 July 2008.