SPEC® CINT2006 Result

Copyright 2006-2014 Standard Performance Evaluation Corporation

NEC Corporation

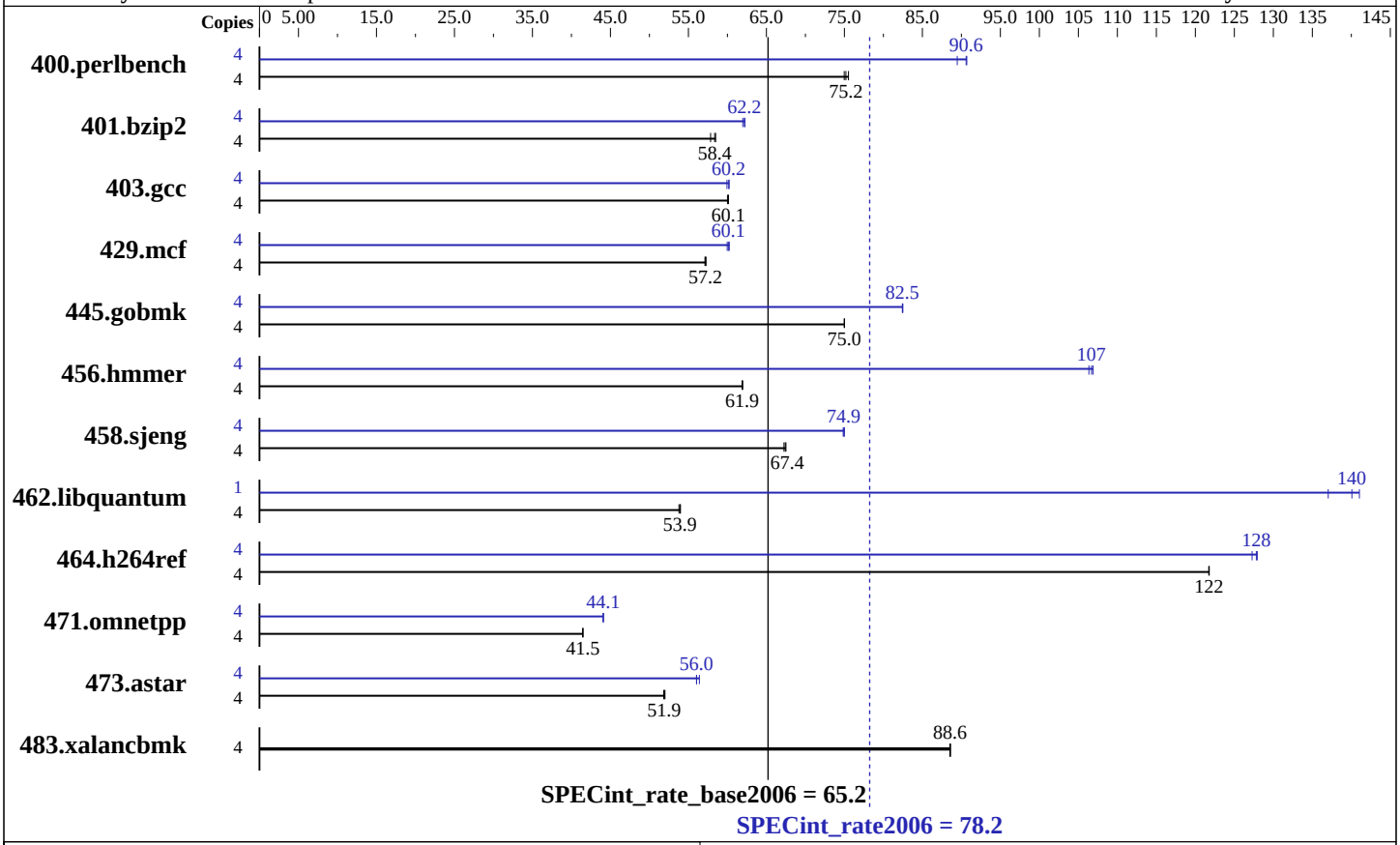
Express5800/140Ba-10
(Intel Xeon E7220)

SPECint®_rate2006 = 78.2

SPECint_rate_base2006 = 65.2

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

Test date: Mar-2008
Hardware Availability: Sep-2007
Software Availability: Nov-2007



Hardware

CPU Name:	Intel Xeon E7220
CPU Characteristics:	2.93 GHz, 1066 MHz bus
CPU MHz:	2933
FPU:	Integrated
CPU(s) enabled:	4 cores, 2 chips, 2 cores/chip
CPU(s) orderable:	1,2,3,4 chips
Primary Cache:	32 KB I + 32 KB D on chip per core
Secondary Cache:	4 MB I+D on chip per core
L3 Cache:	None
Other Cache:	None
Memory:	16 GB (16x1 GB PC2-5300F, 2 rank, CL5-5-5, ECC)
Disk Subsystem:	1x73.2 GB SAS, 10000RPM
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:	ext2
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

Express5800/140Ba-10
(Intel Xeon E7220)

SPECint_rate2006 = 78.2

SPECint_rate_base2006 = 65.2

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

Test date: Mar-2008
Hardware Availability: Sep-2007
Software Availability: Nov-2007

Results Table

Benchmark	Base							Peak						
	Copies	Seconds	Ratio	Seconds	Ratio	Seconds	Ratio	Copies	Seconds	Ratio	Seconds	Ratio	Seconds	Ratio
400.perlbench	4	521	75.0	**520**	**75.2**	517	75.5	4	431	90.7	**431**	**90.6**	437	89.5
401.bzip2	4	660	58.5	667	57.8	**661**	**58.4**	4	620	62.2	623	62.0	**621**	**62.2**
403.gcc	4	**536**	**60.1**	537	60.0	536	60.1	4	**535**	**60.2**	535	60.2	537	59.9
429.mcf	4	637	57.3	639	57.1	**638**	**57.2**	4	608	60.0	**607**	**60.1**	606	60.2
445.gobmk	4	560	75.0	559	75.0	**559**	**75.0**	4	509	82.4	509	82.5	**509**	**82.5**
456.hmmer	4	603	61.9	**603**	**61.9**	602	62.0	4	351	106	**350**	**107**	349	107
458.sjeng	4	717	67.5	720	67.3	**718**	**67.4**	4	645	75.0	**646**	**74.9**	647	74.8
462.libquantum	4	1541	53.8	**1537**	**53.9**	1535	54.0	1	147	141	151	137	**148**	**140**
464.h264ref	4	727	122	**727**	**122**	727	122	4	692	128	695	127	**693**	**128**
471.omnetpp	4	603	41.5	**603**	**41.5**	603	41.5	4	567	44.1	**567**	**44.1**	567	44.1
473.astar	4	542	51.8	**541**	**51.9**	540	52.0	4	498	56.4	501	56.0	**501**	**56.0**
483.xalancbmk	4	**312**	**88.6**	312	88.5	311	88.6	4	**312**	**88.6**	312	88.5	311	88.6

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
```

General Notes

```
All benchmarks compiled in 32-bit mode except 401.bzip2,
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 -inline-calloc -opt-malloc-options=3`

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`
483.xalancbmk: `-DSPEC_CPU_LINUX`

Peak Optimization Flags

C benchmarks:

Continued on next page

Peak Optimization Flags (Continued)

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`

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`

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.20090713.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.20090713.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 17:22:31 2014 by SPEC CPU2006 PS/PDF formatter v6932.
Originally published on 13 May 2008.