## CINT2000 Result

### Hewlett-Packard Company

**AlphaServer GS1280 7/1300**

<table>
<thead>
<tr>
<th>SPEC_license #:</th>
<th>2</th>
<th>Tested by:</th>
<th>HP</th>
<th>Test date:</th>
<th>Jun-2004</th>
<th>Hardware Avail:</th>
<th>Aug-2004</th>
<th>Software Avail:</th>
<th>Jul-2004</th>
</tr>
</thead>
</table>

### SPECint_rate2000 = 366

### SPECint_rate_base2000 = 333

### Hardware

- **CPU:** Alpha 21364
- **CPU MHz:** 1300
- **FPU:** Integrated
- **CPU(s) enabled:** 32 cores, 32 chips, 1 core/chip
- **CPU(s) orderable:** 2 to 64
- **Parallel:** No
- **Primary Cache:** 64KB(I)+64KB(D) on chip
- **Secondary Cache:** 1.75MB on chip per CPU
- **L3 Cache:** None
- **Other Cache:** None
- **Memory:** 2GB per CPU; 256MB RIMMs
- **Disk Subsystem:** AdvFS
- **Other Hardware:** None

### Software

- **Operating System:** Tru64 UNIX V5.1B-1 + PK4
- **Compiler:** Compaq C V6.5-011-48C5K
- **Program Analysis Tools V2.0**
- **Spike V5.2 (510 USG)**
- **Compaq C++ V6.5-041**
- **File System:** MFS, 16GB
- **System State:** Multi-user

### Notes/Tuning Information

#### Baseline C:

- `cc -arch ev7 -fast +CFB ONESTEP`
- `C++: cxx -arch ev7 -O2 ONESTEP`

#### Peak:

- All but 252.eon: `cc -q3 -arch ev7 ONESTEP`
- `164.gzip: -fast -O4 -non_shared +CFB`
- `175.vpr: -fast -O4 -assume restricted_pointers +CFB`
- `176.gcc: -fast -O4 -xtaso_short -all -ldensemalloc -none +CFB +IFB`
- `181.mcf: -fast -xtaso_short +CFB +IFB +PFB`
- `186.crafty: same as base`
- `197.parser: -fast -O4 -xtaso_short -non_shared +CFB`
- `252.eon: cxx -arch ev7 -O2 -all -ldensemalloc -none`
- `253.perlbmk: -fast -non_shared +CFB +IFB +PFB`
- `254.gap: -fast -non_shared +CFB +IFB +PFB`
- `255.vortex: -fast -non_shared +CFB +IFB`
- `256.bzip2: -fast -non_shared +CFB`
- `300.twolf: -fast -O4 -ldensemalloc -non_shared +CFB +IFB`
Notes/Tuning Information (Continued)

Most benchmarks are built using one or more types of profile-driven feedback. The types used are designated by abbreviations in the notes:

+CFB: Code generation is optimized by the compiler, using feedback from a training run. These commands are done before the first compile (in phase "fdo_pre0"):

```
mkdir /tmp/pp
rm -f /tmp/pp/${baseexe}*
```

and these flags are added to the first and second compiles:

```
PASS1_CFLAGS = -prof_gen_noopt -prof_dir /tmp/pp
PASS2_CFLAGS = -prof_use_feedback -prof_dir /tmp/pp
```

(Peak builds use /tmp/pp above; base builds use /tmp/pb.)

+IFB: Icache usage is improved by the post-link-time optimizer Spike, using feedback from a training run. These commands are used (in phase "fdo_postN"):

```
mv ${baseexe} oldexe
spike oldexe -feedback oldexe -o ${baseexe}
```

+PFB: Prefetches are improved by the post-link-time optimizer Spike, using feedback from a training run. These commands are used (in phase "fdo_post_makeN"):

```
rm -f *Counts*
mv ${baseexe} oldexe
pixie -stats dstride oldexe 1>pixie.out 2>pixie.err
mv oldexe.pixie ${baseexe}
```

A training run is carried out (in phase "fdo_runN"), and then this command (in phase "fdo_postN"):

```
spike oldexe -fb oldexe -stride_prefetch -o ${baseexe}
```

When Spike is used for both Icache and Prefetch improvements, only one spike command is actually issued, with the Icache options followed by the Prefetch options.

vm:

```
vm_bigpg_enabled = 1
vm_bigpg_thresh = 6
vm_swap_eager = 0
ubc_maxpercent = 50
```

proc:

```
max_per_proc_address_space = 34359738368
max_per_proc_data_size = 34359738368
max_per_proc_stack_size = 34359738368
max_proc_per_user = 2048
max_threads_per_user = 4096
maxusers = 2048
```
Hewlett-Packard Company
AlphaServer GS1280 7/1300

SPECint_rate2000 = 366
SPECint_rate_base2000 = 333

Notes/Tuning Information (Continued)

per_proc_address_space = 34359738368
per_proc_data_size = 34359738368
per_proc_stack_size = 34359738368

Portability: gcc: -Dalloca=__builtin_alloca; crafty: -DALPHA
perlbmk: -DSPEC_CPU2000_DUNIX; vortex: -DSPEC_CPU2000_LP64
gap: -DSYS_HAS_CALLOC_PROTO -DSYS_IS_BSD -DSYS_HAS_IOCTL_PROTO
     -DSPEC_CPU2000_LP64

Information on UNIX V5.1B Patches can be found at

Processes were bound to CPUs using "runon".