# CFP2000 Result

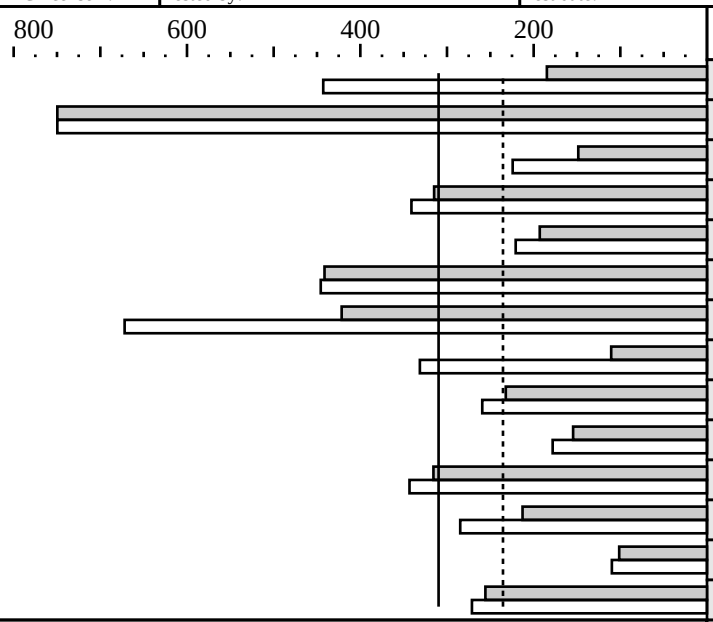
Copyright ©1999-2004, Standard Performance Evaluation Corporation

## Hewlett-Packard Company
## AlphaServer GS1280 7/1300

SPECfp_rate2000 = 310

SPECfp_rate_base2000 = 235

SPEC license #: 2 | Tested by: HP | Test date: Jun-2004 | Hardware Avail: Aug-2004 | Software Avail: Jul-2004




| Benchmark | Base Copies | Base Runtime | Base Ratio | Copies | Runtime | Ratio |
|---|---|---|---|---|---|---|
| 168.wupwise | 16 | 161 | 185 | 16 | 67.1 | 443 |
| 171.swim | 16 | 76.8 | 750 | 16 | 76.8 | 750 |
| 172.mgrid | 16 | 225 | 149 | 16 | 149 | 224 |
| 173.applu | 16 | 124 | 315 | 16 | 114 | 341 |
| 177.mesa | 16 | 135 | 193 | 16 | 118 | 221 |
| 178.galgel | 16 | 122 | 441 | 16 | 121 | 446 |
| 179.art | 16 | 114 | 422 | 16 | 71.8 | 672 |
| 183.equake | 16 | 218 | 111 | 16 | 72.8 | 331 |
| 187.facerec | 16 | 152 | 232 | 16 | 136 | 259 |
| 188.ammp | 16 | 264 | 154 | 16 | 229 | 178 |
| 189.lucas | 16 | 118 | 316 | 16 | 108 | 343 |
| 191.fma3d | 16 | 183 | 213 | 16 | 137 | 285 |
| 200.sixtrack | 16 | 202 | 101 | 16 | 186 | 110 |
| 301.apsi | 16 | 189 | 256 | 16 | 178 | 271 |

### Hardware

| | |
|---|---|
| CPU: | Alpha 21364 |
| CPU MHz: | 1300 |
| FPU: | Integrated |
| CPU(s) enabled: | 16 cores, 16 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<br>Program Analysis Tools V2.0<br>Spike V5.2 (510 USG)<br>HP Fortran V5.5A-3548-48D88<br>HP Fortran 77 V5.5A-3548-48D88<br>KAP Fortran V4.3 000607<br>KAP Fortran 77 V4.1 980926<br>KAP C V4.1 000607 |
| File System: | MFS, 16GB |
| System State: | Multi-user |

## Notes/Tuning Information

```
Baseline   C: cc  -arch ev7 -fast -O4 ONESTEP
     Fortran: f90 -arch ev7 -fast -O5 ONESTEP


Peak:
  All use -g3 -arch ev7 -non_shared ONESTEP
  except these (which use only the tunings shown below):
     173.applu 188.ammp 191.fma3d
  Individual benchmark tuning:
  168.wupwise: kf77 -call_shared -inline all -tune ev67
               -unroll 12 -automatic -align commons -arch ev67
               -fkapargs=' -aggressive=c -fuse
               -fuselevel=1 -so=2 -r=1 -o=1 -interleave
               -ur=6 -ur2=060 ' +PFB
      171.swim: same as base
     172.mgrid: kf90 -call_shared -arch generic -O5 -inline
                manual -nopipeline -transform_loops -unroll 9 -automatic
```

## Notes/Tuning Information (Continued)

```
                  -fkapargs='-aggressive=a -fuse -interleave
                  -ur=2 -ur3=5 -cachesize=128,16000 ' +PFB
      173.applu: kf90  -O5 -transform_loops
                 -fkapargs=' -o=0 -nointerleave -ur=14
                 -ur2=260 -ur3=18' +PFB
       177.mesa: kcc -fast -O4 +CFB +IFB
     178.galgel: f90 -O5 -fast -unroll 5 -automatic
        179.art: kcc  -assume whole_program -ldensemalloc
                 -call_shared -assume restricted_pointers
                 -unroll 16 -inline none -ckapargs='
                 -fuse -fuselevel=1 -ur=3' +PFB
     183.equake: cc -call_shared -arch generic -fast -O4
                 -ldensemalloc -assume restricted_pointers
                 -inline speed -unroll 13 -xtaso_short +PFB
    187.facerec: f90 -O4 -nopipeline -inline all
                 -non_shared -speculate all -unroll 7
                 -automatic -assume accuracy_sensitive
                 -math_library fast +IFB
       188.ammp: cc -arch host -O4 -ifo -assume nomath_errno
                 -assume trusted_short_alignment -fp_reorder
                 -readonly_strings -ldensemalloc -xtaso_short
                 -assume restricted_pointers -unroll 9
                 -inline speed +CFB +IFB +PFB
      189.lucas: kf90 -O5 -fkapargs='-ur=1' +PFB
      191.fma3d: kf90 -O4 -transform_loops -fkapargs='-cachesize=128,16000 ' +PFB
   200.sixtrack: f90 -fast -O5 -assume accuracy_sensitive
                 -notransform_loops +PFB
      301.apsi: kf90 -O5 -inline none -call_shared -speculate all
                 -align commons -fkapargs=' -aggressive=ab
                 -tune=ev5 -fuse -ur=1 -ur2=60 -ur3=20
                 -cachesize=128,16000'

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}
```

Copyright ©1999-2004, Standard Performance Evaluation Corporation



## Notes/Tuning Information (Continued)

```
+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
        per_proc_address_space = 34359738368
        per_proc_data_size = 34359738368
        per_proc_stack_size = 34359738368


Portability: galgel: -fixed

Information on UNIX V5.1B Patches can be found at
http://ftp1.service.digital.com/public/unix/v5.1b/

Processes were bound to CPUs using "runon".
```