

# UMTmk

---

## Summary Version

1.2

## Purpose of Benchmark

UMTmk measures single CPU (serial) performance in the presence of heavy memory access.

## Characteristics of Benchmark

UMTmk is a Microkernel Benchmark based on UMT, which is an LLNL ASC proxy application (mini-app). UMT performs three-dimensional, non-linear, radiation transport calculations using deterministic (Sn) methods.

The microkernel will measure the performance of a set of loops encapsulated in the F90 function `snswp3d`. The function `snswp3d` is UMT's largest consumer of CPU resources. By initializing the F90 objects with a set of realistic data we have attempted to mimic the performance of UMT on bidders' hardware/software platforms.

## Mechanics of Building Benchmark

1. `bunzip2 UMTmk1.2.tar.bz2`
2. `tar oxvf UMTmk1.2.tar`
3. `cd UMTmk1.2`
4. Peruse `README` file
5. Modify `Makefile` to reflect the platform's compilers, compiler options, libraries, MPI wrappers etc. SIMD and/or other vectorization may be turned on and is encouraged.
6. `gmake clean`
7. `gmake`
8. The executable is named `UMTmk` and lives in the top level directory

## Mechanics of Running Benchmark

Execute the binary `UMTmk` - single CPU

Two problem sizes have been provided in two directories, small and large.

- The "small" case has 9336 zones, 64 groups and takes 0.0578 seconds to execute
- The "large" case has 43200 zones, 16 groups and takes 0.1137 seconds to execute

The code will read (realistic) input data from 4 ASCII files: `fort.10`, `fort.11`, `fort.13`, and `fort.14`. The "large" and "small" versions of these 4 files are different and are not interchangeable.

Hence, to run the code:

1. `cd UMTmk1.2/small`
2. `../UMTmk`
3. `cd UMTmk1.2/large`
4. `../UMTmk`

Bidders wishing to reduce the CPU cycles and hence the time for simulation of new hardware on simulators or emulators can change line 10 of `fort.11`, which specifies the number of groups, to something smaller.

## Verification of Results

For the two cases discussed above, "small" and "lsrge", two output files have been provided for comparison:

```
./large/UMTmk1.2_large.out  
./small/UMTmk1.2_small.out
```

Compare your output with the appropriate UMTmk1.2 output file, especially the line with  
`V&V chk =`

The UMTmk code will report the wall time consumed in the function `snswp3d`. The walltime is monitored using the OpenMP function `omp_get_wtime()`.