STREAM

Summary Version

1.0

Purpose of Benchmark

The STREAM benchmark is a simple, synthetic benchmark designed to measure the sustainable memory bandwidth (in MB/s) and a corresponding computation rate for four simple vector kernels.

Characteristics of Benchmark

STREAM runs on each core of a mode using the OpenMP programming paradigm. Four synthetic computational kernels are run, and the best rates observed are reported for each choice of thread count.

Mechanics of Building Benchmark

The benchmark file is called stream_omp.c. Users should compile this program with the highest optimization possible for their particular platform. As an example, on a representative Linux cluster at LLNL using the Intel C compiler, one would compile as follows:

```
     icc –openmp –O3 –o stream_omp stream_omp.c
```

Mechanics of Running Benchmark

Execution of the resulting executable is very simple. The number of OpenMP threads to use should be set using the OMP_NUM_THREADS environment variable. Users should run from one thread on the core up to “N” threads, where “N: is the number of cores on a node.

For each vector kernel (Copy, Scale, Add, Triad) a memory bandwidth rate, average time, minimum time, and maximum time are reported.

Verification of Results

The STREAM benchmark reports a variety of data for the benchmark, including the number of threads, array size, etc. For each run, all of the data generated should be included in the benchmark submission as well as the compilation details.

CORAL acknowledges Dr. John McCalpin as the developer and maintainer of the STREAM benchmark.

A detail description for the benchmark and sample results can be found at the STREAM benchmark home at http://www.cs.virginia.edu/stream/ref.html.