Problem 2: Parallel Square Root Computation

Basic System Specs:
Personal Computer
CPU: Intel Core i7-6600U CPU @ 2.60GHz
Cores = 2
Ram: 16GB
ISPC code --target=avx2-i32x8
AVX = 256 bits (8 lanes)

Our observation shows us significant improvement from sequential execution to the SIMD compiled versions. The ISPC version with a single task (1 core) launched versus the sequential execution showed a speed up of x2.30. Running on several different numbers of tasks, on our dual core system, a speed up of x1.79 was reached but this quickly fell off. There still was speed up on an subsequent number of tasks until 8, but the speed up was insignificant compared to the first two (as shown by the graph below). Any speed up is likely due to the utilization of hardware threads during inactivity. The use of more threads tapered off any performance gains.

Our implementation of AVX code was also significantly faster than the single threaded sequential version, and managed to beat the sequential ISPC version as well. The speed up of the AVX from sequential was x3.68 and the speed up of the AVX from the ISPC was x1.60. The significant speedup of AVX could have been due to many factors such as cache locality, memory alignment (ours was aligned), and efficient code execution, but it showed better/similar benefits to using ISPC. The advantage of the ISPC code however is the ease of use compared to the AVX.

Although changing ISPC's target architecture to avx2-i32x16 should have resulted in a large speed up, the computer I was using did not show any (perhaps it was not available and compiled to avx2-i32x8 anyway.

Sample benchmarks during an execution with optimization level set to -O0 (make debug):
Non vectorized computation of square root of 20000000 floating point numbers took 1147605673 ns (1147 ms).
ISPC 1 TASK(S) driven computation of square root of 20000000 floating point numbers took 499542662 ns (499 ms).
ISPC 2 TASK(S) driven computation of square root of 20000000 floating point numbers took 279373115 ns (279 ms).
ISPC 3 TASK(S) driven computation of square root of 20000000 floating point numbers took 244505507 ns (244 ms).
ISPC 4 TASK(S) driven computation of square root of 20000000 floating point numbers took 228583498 ns (228 ms).
ISPC 5 TASK(S) driven computation of square root of 20000000 floating point numbers took 268387584 ns (268 ms).
ISPC 6 TASK(S) driven computation of square root of 20000000 floating point numbers took 269768660 ns (269 ms).
ISPC 7 TASK(S) driven computation of square root of 20000000 floating point numbers took 292493257 ns (292 ms).
ISPC 8 TASK(S) driven computation of square root of 20000000 floating point numbers took 244444109 ns (244 ms).
AVX driven computation of square root of 20000000 floating point numbers took 312534996 ns (312 ms).

Graph is at next page.

Square Root Computation Time