|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test** | **Average Time (Over 5 Runs) (Milliseconds)** | | | |
| **Base Multi-Threaded** | **Base SIMD Spheres** | **Stage 1 SIMD Triangles** | **Stage 2 SIMD Lighting** |
| cornell 1024x1024x1 | 232ms | 250ms | 206ms | 1187ms |
| cornell 1024x1024x4 | 3450ms | 3696ms | 2928ms | 21415ms |
| cornell 500x300x1 | 29ms | 32ms | 25ms | 128ms |
| cornell-256lights 512x512x1 | 5625ms | 5982ms | 5912ms | 7031ms |
| allmaterials 1024x1024x1 | 96ms | 73ms | 81ms | 206ms |
| 5000spheres 960x540x1 | 4843ms | 1368ms | 1353ms | 2443ms |
| bunny500.txt 1024x1024x1 | 4118ms | 4229ms | 2643ms | 13981ms |
| bunny10k.txt 256x256x1 | 7862ms | 8029ms | 4946ms | 25115ms |

From the table of above, the speed of stage 1, generally, is quicker than standard time, basic multithreaded and base SIMD spheres, and much faster than stage 2. It is because stage 1 used SIMD function within buffer. The reason of stage 2 so slow is because stages need to shift data between normal and SIMD multiple time. In the end, it causes delay. Of course, it also because programming problem. Stage 2 cannot give a correct output.