Project #4 Vectorized Array Multiplication/Reduction using SSE

Erick Branner, brannere@oregonstate.edu | 11 May 2022

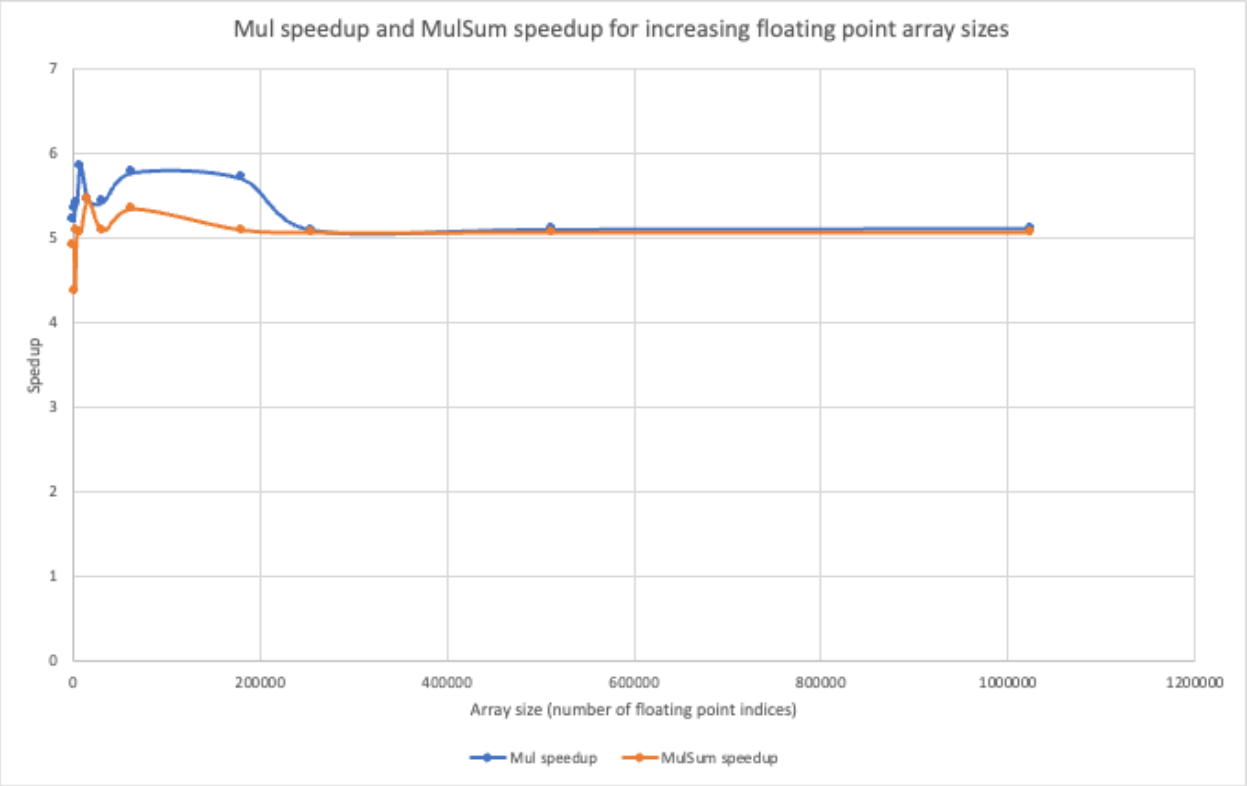
What machine you ran this on

Rabbit

Show the table of performances for each array size and the corresponding speedups

Array size	Mul speedup	MulSum speedup
1000	5.2	4.9
2000	5.35	4.36
4000	5.4	5.07
8000	5.85	5.06
16000	5.45	5.44
32000	5.42	5.08
64000	5.77	5.34
180000	5.7	5.09
256000	5.09	5.06
512000	5.1	5.06
1024000	5.11	5.06

Show the graph of SIMD/non-SIMD speedup versus array size (either one graph with two curves, or two graphs each with one curve)



What patterns are you seeing in the speedups?

Performance becomes about the same for larger array sizes on both. The MulSum speedup is smaller (more instructions to store the sum). Both have a speedup greater than four.

Are they consistent across a variety of array sizes?

They follow a similar pattern, though speedup for MulSum is smaller than Mul speedup. With larger array sizes, both speedups get close to a limit of five.

Why or why not, do you think?

When using SSE, you can do four floating point multiplies in one instruction. We can see the speedup greater than four because the non-SSE implementation will use more than one instruction for each multiply.