SAND2018-11376R

EXASCALE COMPUTING PROJECT

ECP-U-2018-XXX

Kokkos interoperability with general SIMD types to force vectorization on ATS-1

WBS STPR 04 Milestone 5

Authors

Christian Trott (1)

Author Affiliations

(1) Sandia National Laboratories

September 19th 2018

U.S. DEPARTMENT OF ENERGY | Office of Science

NNSA National Nuclear Security Administration

EXASCALE COMPUTING PROJECT

EXECUTIVE SUMMARY

This report documents the completion of milestone STPR04-5 Kokkos interoperability with general SIMD types to force vectorization on ATS-1. The Kokkos team worked with application developers to enable the utilization of SIMD intrinsics, which allowed up to 3.7x improvement of the affected kernels on ATS-1 in a proxy application. SIMD types are now deployed in the production code base.

U.S. DEPARTMENT OF ENERGY | Office of Science

NNSA National Nuclear Security Administration

1. MILESTONE OVERVIEW

1.1 DESCRIPTION

Kokkos interoperability with general SIMD types to force vectorization on ATS-1

1.2 EXECUTION PLAN

1) Identify incompatibilities in Kokkos' runtime with using SIMD intrinsic types.
2) Fix incompatibilities in Kokkos' runtime with using SIMD intrinsic types.

1.3 COMPLETION CRITERIA

Demonstrate interoperability of SIMD types in an application.

2. TECHNICAL WORK SCOPE, APPROACH, RESULTS

2.1 CODE ISSUES

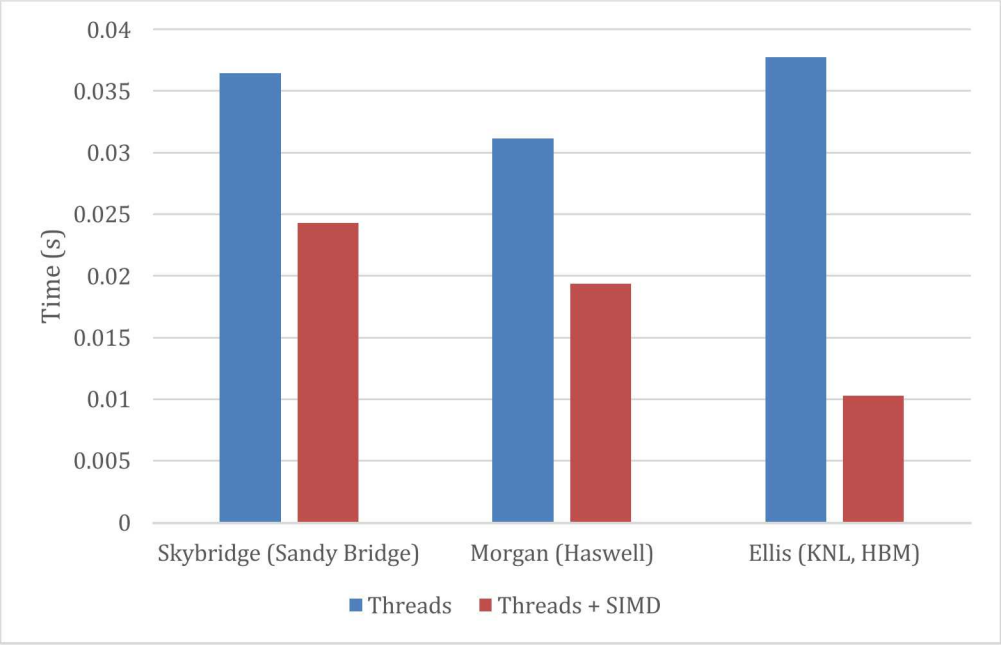
A priori we wouldn't expect any fundamental problems of using SIMD types in Kokkos. Since the vast majority of Kokkos operations and API are agnostic of scalar types. But we encountered a number of them related to alignment. One was caused by a hardcoded alignment value for Kokkos Views. Effectively we have an architecture specific alignment value, but if that was too small for the SIMD type some operations which require alignment to the SIMD length may fail. To resolve this we added a compile time option to override our alignment setting. Another recent issue is that Kokkos Views in Scratch Space were not aligned correctly. This is because the per team scratch in Kokkos' hierarchical parallelism works like a stack. So even if the initial pointer is aligned properly, allocating a view changes the alignment for the next one. To resolve that we increased the scratch size requirements for each view by one scalar size. This way we can align each individual view to its requirement. The overhead of this is additional scratch memory and some extra integer operations at scratch memory view creation. But for most use cases the size overhead is modest. For example for a 16x16 2D per team scratch view of doubles which previously required 2048 bytes, we would only add a single double to end up at 2056 bytes.

All in all there were no major issues with using SIMD intrinsic types in conjunction with Kokkos.

2.2 APPLICATION PERFORMANCE

The primary initial customer asking for SIMD support were Sandia's CFD applications such as NALU which is the application code base for ECP ExaWind. To isolate the benefit of using SIMD types and explore issues in Kokkos regarding the use of SIMD types we developed a miniApp (not publicly released) which implemented the main part of the matrix assembly, i.e. the section where the physics is evaluated. The problem run was a simple heat conduction calculation with a mesh size of 60x60x60. The

results were encouraging providing up to 3.7x performance increase on Intel KNL chips as deployed in ATS 1. NALU adopted those changes later.



2.3 RESOURCE ACCESS

Further evidence and details for the performed work can be found in the relevant

Feature	Location	Access Restriction
Kokkos	https://github.com/kokkos/kokkos/	Open access.

3. RESOURCE REQUIREMENTS

The work performed here required 0.125 FTE.

4. CONCLUSIONS AND FUTURE WORK

There are no major outstanding issues for interoperability of Kokkos with SIMD intrinsic types.

5. ACKNOWLEDGMENTS

This research was supported by the Exascale Computing Project (ECP), Project Number: 17-SC-20-SC, a collaborative effort of two DOE organizations—the Office of Science and the National Nuclear Security

Administration—responsible for the planning and preparation of a capable exascale ecosystem—including software, applications, hardware, advanced system engineering, and early testbed platforms—to support the nation's exascale computing imperative.

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology and Engineering Solutions of Sandia, LLC. a wholly owned subsidiary of Honeywell International, Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA-0003525.