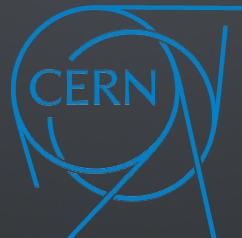




Vectorization of Philox CBRNG and VecGeom via Agner FOG's
Vector Class Lib.

Yigit Demirag

Advisor: Dr. Sandro Wenzel





Outline

1. Personal Background
2. Project Description
3. Brief Timeline





Personal Background

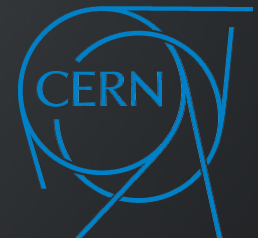
- Senior EEE Student, Bilkent University, Turkey
- Undergrad Researcher @ Computational EM Research Center, Turkey ([threading](#), [vectorization](#))
- Undergrad Researcher @ Nanotechnology Research Center, Turkey ([signal processing](#), [ML](#))
- Openlab 2014 Summer Student at CERN ([vectorization](#))





My GSoC Project (Part-I : Vectorization of Philox CBRNG)

- CBRNGs are widely used at CERN, especially in MC Simulations in GEANT4 and ROOT.
- PRNGs are deterministic algorithms in form of
`uint_64t someRandomNumber = CBRNG(uint64_t key, uint64_t counter)`
- We want to compare AVX2 and SSE4.2 performances using vector library and autovectorization.





My GSoC Project (Part-I : Vectorization of Philox CBRNG)

- Philox is a SP Network.
- S box is a simple Feistel function with 72 rounds
64-bit [XOR, MUL]

$$L' = B_k(R) = \text{mullo}(R, M)$$

$$R' = F_k(R) \oplus L = \text{mulhi}(R, M) \oplus k \oplus L$$

- 16 rounds 64-bit ADD with constant to key.

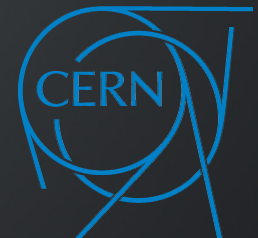




My GSoC Project (Part-I : Vectorization of Philox CBRNG)

Basic steps to vectorize:

1. Convert I/O data structures from AoS to AoS.
2. Make sure arrays are aligned.
3. Check data dependencies.
4. Use `__restrict__` keyword in C99 to tell compiler there is no pointer aliasing.
5. Make sure only calling functions are inlined.
6. Try auto-vectorization with `(-O3 -xAVX2 -vec-report2)`
7. If does not vectorize try `#pragma vector always`, then `#pragma ivdep`.
8. Increase performance by unrolling the loops by `#pragma unroll(N)`.
9. Use encapsulated intrinsics of Agner FOG's lib.





My GSoC Project (Part-I : Vectorization of VecGeom via FOG's Lib)

- VecGeom is a high-performance HEP geometry system that originally developed to be a turn-key replacement for HEP simulation applications e.g. GEANT4, ROOT, USolids.
- It focuses on new hardware architectures and aims to use SIMD vectors whenever possible.





My GSoC Project (Part-I : Vectorization of VecGeom via FOG's Lib)

- Starting from backend (Backend.h and Vector3D.h), all Vc library implementations will be changed with FOG's VCL.
- The same will be done for geometry and physics processes using vectorization steps described before.
- Finally performance tests are going to be held to compare platform independencies and performance of two different libraries.





Thank You!

Yigit Demirag

yigitdemirag@gmail.com

yigitdemirag.com

