RIKEN Fugaku Processor Simulator Updates on Nov 2019

Yuetsu Kodama
RIKEN R-CCS
yuetsu.kodama@riken.jp

Disclaimer

- The software used for the evaluation, such as the compiler, is still under development and its performance may be different when the supercomputer Fugaku starts its operation.
- The result of the A64FX test chip does not guarantee the performance of the supercomputer Fugaku at the start of its operation.
- The result of the RIKEN post-K processor simulator is just an estimated value, and it does not guarantee the performance of the supercomputer Fugaku at the start of its operation.

Outline

- **Fugaku and A64FX**
- **RIKEN Fugaku processor simulator**
  - Gem5
  - Extensions
    - Software/Hardware prefetch
- **Evaluation results**
  - Stream Benchmark on L1/Memory with prefetch
- **Conclusion**

Fugaku

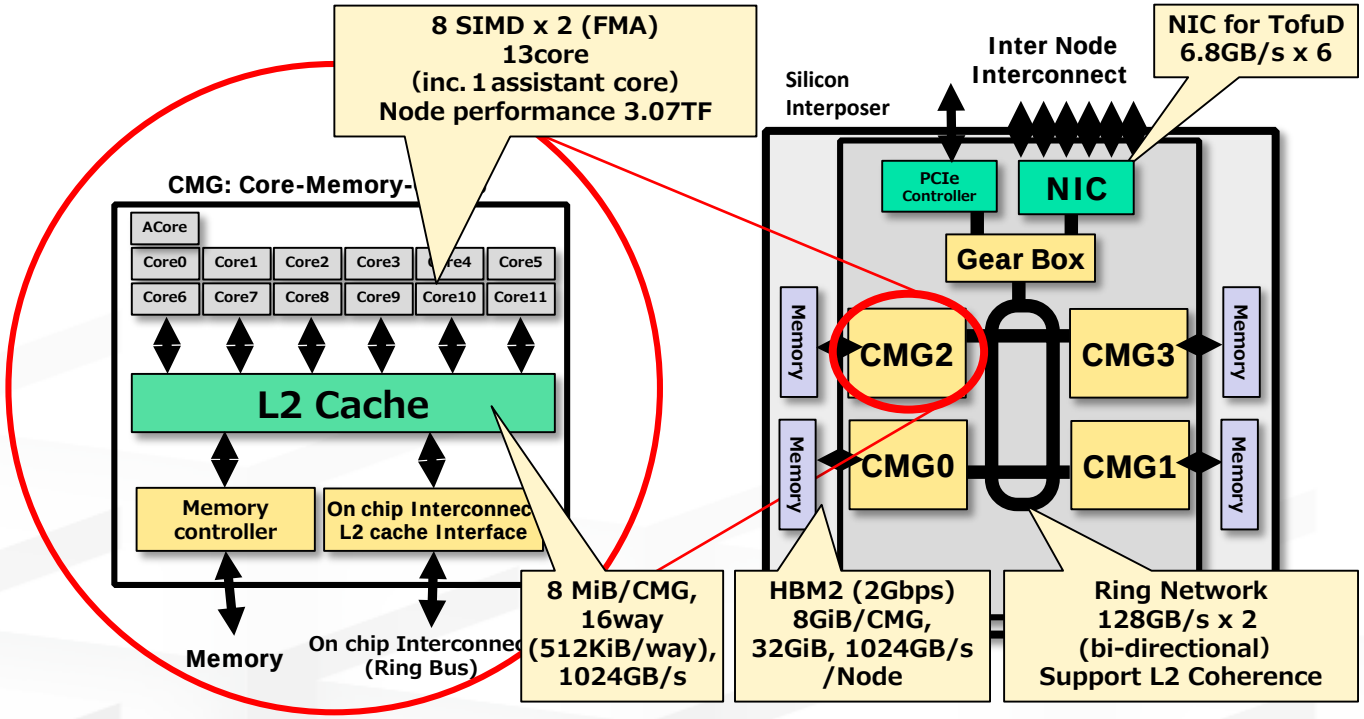
- A Fugaku will start service around 2021, and prototype system has been built in the summer of 2018 and is being tested and evaluated.
- 10 racks of Fugaku can achieve almost the same performance as all nodes (864 racks) of the K computer.
- 400+ racks, 150k+ nodes, 7M+ cores

https://postk-web.r-ccs.riken.jp/

X 10 =

		Post-K	K
CPU Architecture		A64FX (Armv8.2-A SVE +Fujitsu Extension)	SPARC64 VIIIfx
Node	Cores	48	8
	Peak DP performance	3.37 TF@2.2GHz, 3.07TF@2.0GHz	0.128 TF
	Main Memory	32 GiB	16 GiB
	Peak Memory Bandwidth	1024 GB/s	64 GB/s
	Peak Network Performance	40.8 GB/s	20 GB/s
Rack	Nodes	384	102
	Peak DP performance	1.29 PF@2.2GHz, 1.17PF@2.0GHz	< 0.013PF
Process Technology		7 nm FinFET (TSMC N7)	45 nm

A64FX



Each Values@2.0GHz

For more detailed information, please refer to the following URL.
http://www.fujitsu.com/global/solutions/business-technology/tc/catalog/

RIKEN Simulator (Overview)

- **It is a processor simulator of A64FX.**
  - ✓ It enables simulation of 1CMG, where 12 cores with OpenMP execution is available.
- **It is developed based on open source general purpose processor simulator, gem5.**
  - ✓ Initially RIKEN developed O3 (out-of-order) mode for SVE, but currently moved to the version developed by Arm.
  - ✓ RIKEN continues to extend cache and memory system for HPC.
- **It can simulate binaries generated by an Arm SVE compliant compiler (such as Fujitsu prototype compiler or Arm compiler) with an out-of-order execution pipeline.**
  - ✓ You should generate single static binary including the library.
  - ✓ Currently, it supports only Fujitsu and Gcc OpenMP library, not ARM (LLVM) library and any MPI program.
- **It is aimed to estimate the execution time with accuracy that can be used for performance evaluation and tuning.**

gem5

- **Open source general purpose processor simulator**
  - ✓ Refer http://gem5.org for details
  - ✓ Tutorial: ISCA2011, ASPLOS2017, …
- **Multiple ISAs**
  - ✓ Alpha, Arm, SPARC, MIPS, Power, x86, GPU, RISC-V, …
- **Multiple System mode**
  - ✓ FS (Full System) mode, SE (System Emulation) mode
- **Several CPU execution model**
  - ✓ Atomic, in-order, out-of-order(o3), …
  - ✓ The O3 pipeline architecture is based on Alpha 21264
  - ✓ Detailed parameters such as out-of-order resource size, the number of function unit and so on can be set in parameter file.
- **License**
  - ✓ Berkeley-style, free to use as long you as leave our copyright on it.

RIKEN Simulator（Details 1）

- **Tuning detailed parameters such as o3 resource size according to A64FX**
  - ✓ The size of Reservation station, reorder buffer, rename register.
  - ✓ Latency of each pipeline stage, number of simultaneously issued instructions.
  - ✓ Number, latency and throughput of computing units.
  - ✓ Latency and throughput for each instruction group
  - ✓ L1 / L2 cache size, number of ways, latency, throughput, cache line size.
- **Differences between gem5 and A64FX**
  - ✓ The number of reservation station (Fused or Distributed)
  - ✓ Memory address calculation (In memory unit or Independent unit)
  - ✓ Function unit for Instruction (single units or multiple units)

RIKEN Simulator（Details 2）

- **Expand the new functions**
  - ✓ L1 cache: 1 port SRAM, access across lines without overhead
  - ✓ L2 cache: high throughput by multiple banks with interleave
  - ✓ Memory: support HBM2, interleave and bank schedule for A64FX
  - ✓ Bus: asymmetric throughput at input and output
  - ✓ Software prefetch: store prefetch, target L2 prefetch
  - ✓ Hardware prefetch: K-Computer compliant prefetch, target L2 prefetch, store prefetch
- **Usability improvement**
  - ✓ Get statistical information of simulation execution (execution time, cache miss, etc.) in a specified region
  - ✓ Get statistical information compatible with Fujitsu detailed profiles.
  - ✓ Counting floating operations taking into account predicates.

We will release these extensions separately from the parameters of A64FX.
Binary on Jun 2019 is available from https://hub.docker.com/r/linaro/gem5-riken-open.
Source will be opened in 2019.

Software Prefetch

	PLDL1	PSTL1	PLDL2	PSTL2
gem5	○	○*	×	×
RIKEN simulator	○	○	○	○

* Contribution by RIKEN

PLDL1 (prefetch for load targeting L1)
PSTL1 (prefetch for store targeting L1)
PLDL2 (prefetch for load targeting L2)
PSTL2 (prefetch for store targeting L2)

For PLD, the status of cache becomes shared, and for PST, the status of cache becomes exclusive.

PFM instruction has attribute KEEP/STRM, but Gem5 and RIKEN simulator supports only KEEP.

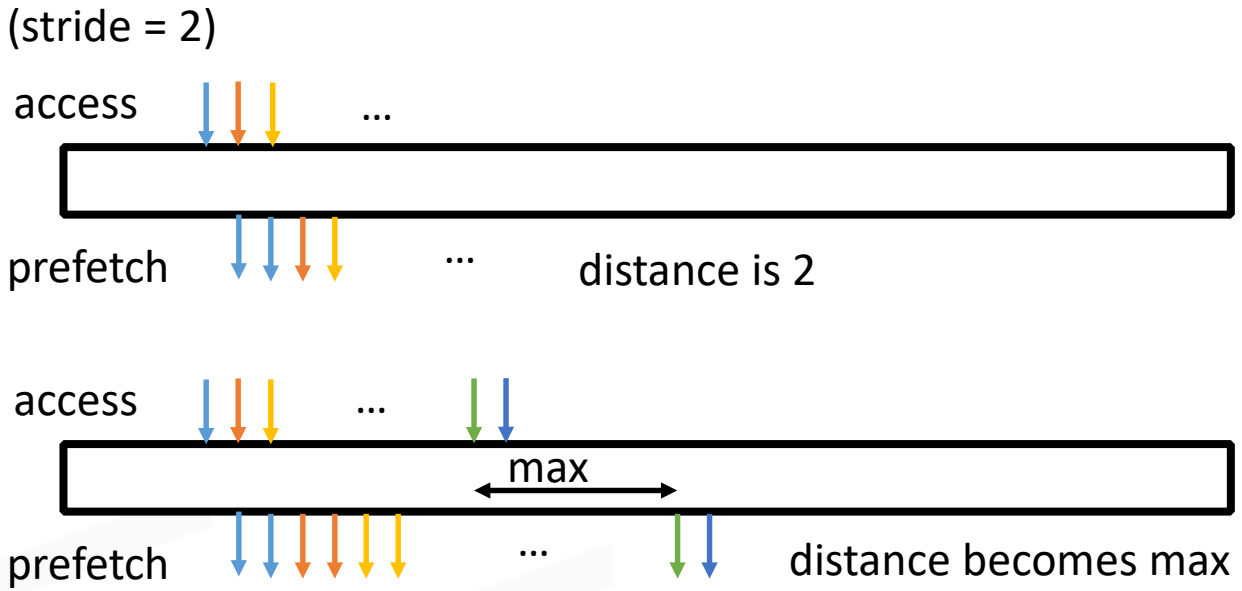
Hardware Prefetch

stride prefetch

Generate N prefetch from next cache line, but prefetch requests has overlap, so next new prefetch is only one.

K prefetch *

First prefetch generates 2 prefetches from next cache line, and second prefetch generates 2 prefetches from next of previous prefetch, so the distance quickly becomes large. When the distance reaches the max value, change to single prefetch.



*: K computer uses it only for L2, but A64FX uses it for L1 and L2

	PLDL1	PSTL1	PLDL2	PSTL2
gem5	○	×	×**	×
RIKEN simulator	○	○	○	○

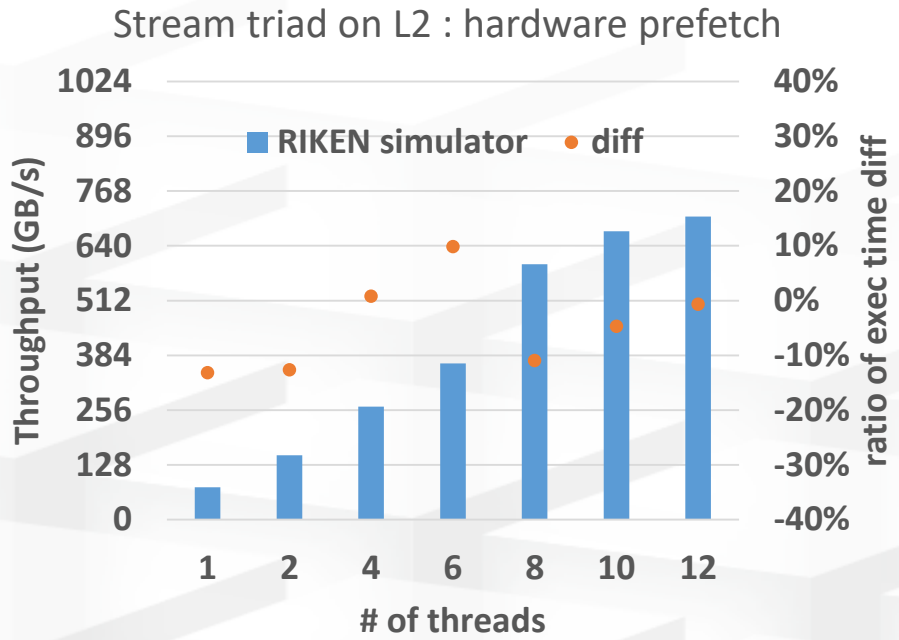
**: not support in banked L2 with interleave

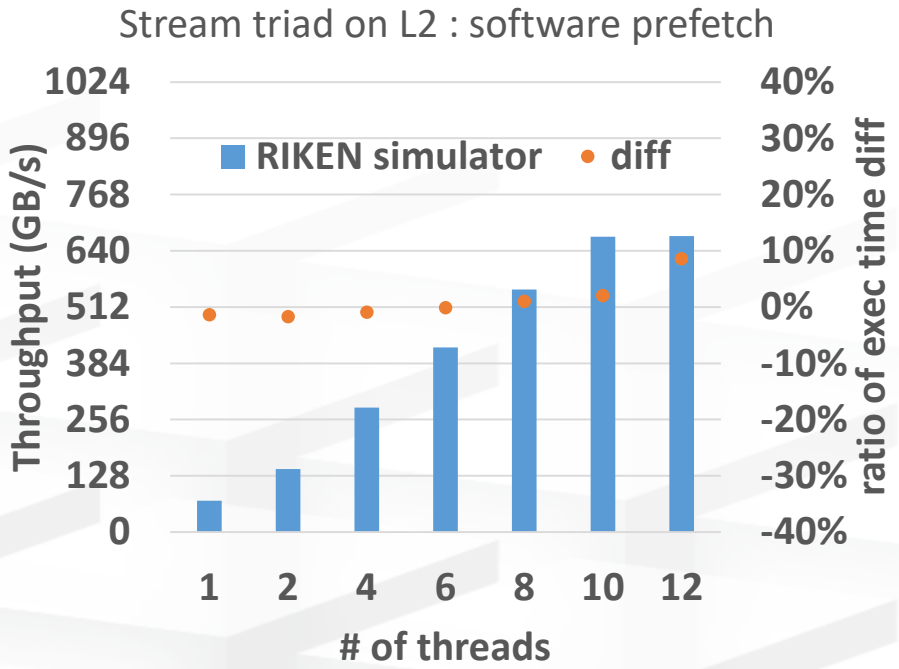
Evaluations

- **It is important to evaluate and confirm the difference between the RIKEN simulator and the actual A64FX.**
- **Target: A64FX test chip**
- **Compiler: Fujitsu compiler prototype in 2019/04**
  - The evaluations using previous version of simulator and compiler is published in http://arxiv.org/abs/1904.06451.
- **Evaluated program:**
  - Arithmetic pipeline: various kernel loops
  - L2 cache throughput: Stream benchmark for L2 size using hardware/software prefetch
  - Memory throughput: Stream benchmark for over L2 size using hardware/software prefetch

Results of L2 Stream benchmark

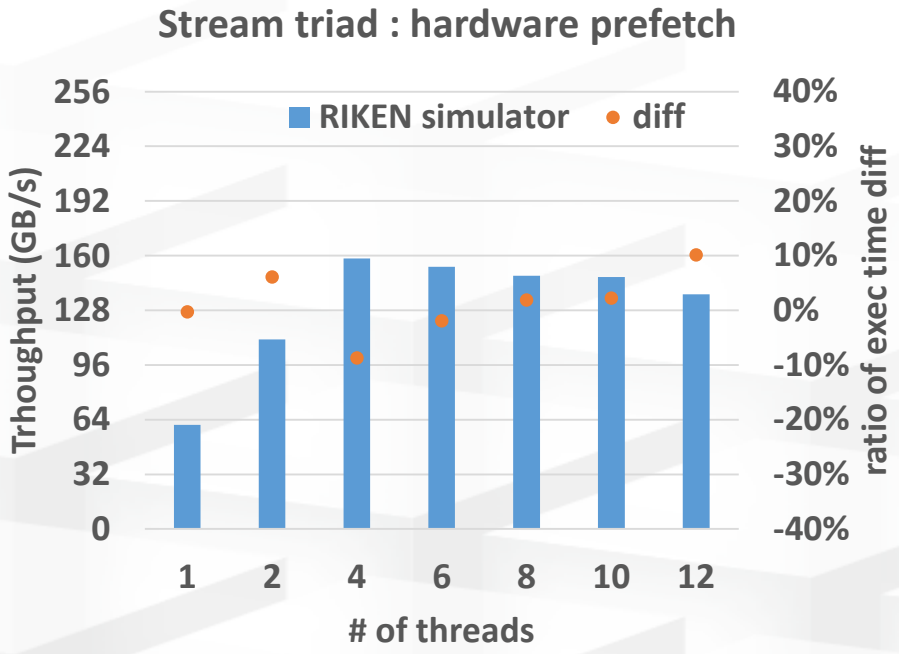
- Measure the total L2 throughput by changing the number of threads from 1 to 12 using hardware / software prefetch.
- Both results are relatively scalable for the number of threads.
- Using hardware prefetch, the variation of difference is something large. This is because the overhead of L2 prefetch, so we continue to reduce the overhead.
- Using software prefetch, the results show very small differences except 12 threads.

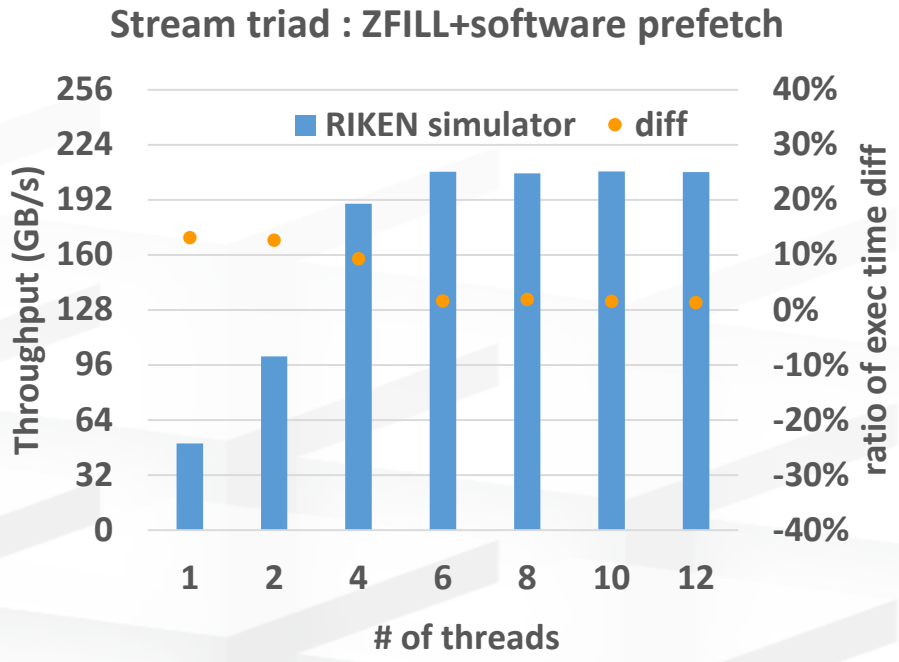




Results of Memory Stream benchmark

- Measure the total memory throughput by changing the number of threads from 1 to 12 using hardware / software prefetch.
- Using hardware prefetch, the memory throughput is saturated in 4 threads, the variation of difference is something large.
- Using ZFILL optimization, the memory throughput is over 200GB/s, and the differences is small except in small threads.





ZFILL Optimization

- **Stream triad**

```
for (i=0; i<N; i++) {
      y[i] = x1[i] + c * x2[i];
}
```

2 read 1 store

- **Memory access**

  Since the store instruction writes to part of the cache line, it is necessary to read the cache line from memory before writing to maintain cache line consistency. Therefore, the actual memory access is 3 read 1 store.

  Therefore, when Stream throughput is 150 GB/s, the memory throughput achieves 200 GB/s.

- **ZFILL Optimization**

  However, when writing to all cache lines as in Stream, it is not necessary to preload from memory. Therefore, the Fujitsu compiler provides the ZFILL option. ZFILL uses a 'DC ZVA' instruction that zero-fills a cache line. As a result, the A64FX test chip achieves 200 GB/s in Stream throughput.

Conclusion

- We developed a RIKEN simulator that can perform cycle-level processor simulation as an evaluation environment until the Fugaku system can be used.
- From the evaluation with the Stream benchmark for L2 and Memory, the results showed that the differences between the RIKEN simulator and the A64FX test chip are about 10%.
- The differences in Stream becomes small from the previous evaluation, because latest simulator supports target L2 and store access in hardware prefetch.
- There were some cases where the difference was large, but we will continue to develop RIKEN simulator.
- We are preparing to open our source code without detailed parameters.