September HPC User Forum 2021

# Supercomputer Fugaku

## Performance Characteristics of A64FX Processor

**Mitsuhisa Sato** Team Leader of Architecture Development Team

Deputy project leader, FLAGSHIP 2020 project

Deputy Director, RIKEN Center for Computational Science (R-CCS)

Professor (Cooperative Graduate School Program), University of Tsukuba

Tetsuya Odajima and Yuetsu Kodama, … and many project members

FLAGSHIP 2020 project, R-CCS

# Outline of my talk

- **Co-design of A64FX processor for “Fugaku” in FLAGSHIP 2020 project**
  - Design target and KPIs, and co-design
  - Overview of A64FX processor
    - A64FX was developed by Fujitsu and RIKEN, and the first processor equipped with Arm SVE.
- **The Performance results of A64FX processor**
  - UK benchmark and LULESH
  - Open-source HPC software
  - SPEC® benchmark
  - Summary of A64FX performance characteristics
- **Performance Tuning and Power Control**
- **Concluding remarks**

# KPIs on Fugaku development in FLAGSHIP 2020 project

## 3 KPIs (key performance indicator) were defined as the design target for Fugaku development

- **1. Extreme Power-Efficient System**
  - Maximum performance under Power consumption of 30 - 40MW (for system)
- **2. Effective performance of target applications**
  - It is expected to exceed 100 times higher than the K computer’s performance in some applications
- **3. Ease-of-use system for wide-range of users**

# Target Application's Performance

- **Performance Targets**
  - 100 times faster than K for some applications (tuning included)
  - 30 to 40 MW power consumption

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

## Predicted Performance of 9 Target Applications

*As of 2019/05/14*

| Area | Priority Issue | Performance Speedup over K | Application | Brief description |
|---|---|---|---|---|
| Health and longevity | 1. Innovative computing infrastructure for drug discovery | **x125+** | **GENESIS** | MD for proteins |
| | 2. Personalized and preventive medicine using big data | **x8+** | **Genomon** | Genome processing (Genome alignment) |
| Disaster prevention and Environment | 3. Integrated simulation systems induced by earthquake and tsunami | **x45+** | **GAMERA** | Earthquake simulator (FEM in unstructured & structured grid) |
| | 4. Meteorological and global environmental prediction using big data | **x120+** | **NICAM+ LETKF** | Weather prediction system using Big data (structured grid stencil & ensemble Kalman filter) |
| Energy issue | 5. New technologies for energy creation, conversion / storage, and use | **x40+** | **NTChem** | Molecular electronic (structure calculation) |
| | 6. Accelerated development of innovative clean energy systems | **x35+** | **Adventure** | Computational Mechanics System for Large Scale Analysis and Design (unstructured grid) |
| Industrial competitiveness enhancement | 7. Creation of new functional devices and high-performance materials | **x30+** | **RSDFT** | Ab-initio program (density functional theory) |
| | 8. Development of innovative design and production processes | **x25+** | **FFB** | Large Eddy Simulation (unstructured grid) |
| Basic science | 9. Elucidation of the fundamental laws and evolution of the universe | **x25+** | **LQCD** | Lattice QCD simulation (structured grid Monte Carlo) |

# Codesign of "Fugaku"

R-CCS

**3 Design Targets:**

- **1. Extreme Power-Efficient System**
  - Maximum performance under Power consumption of 30 - 40MW (for system)
- **2. Effective performance of target applications**
  - It is expected to exceed 100 times higher than the K computer's performance in some applications
- **3. Ease-of-use system for wide-range of users**

Cool (Low-power) technology is important!!

Codesign

Codesign to meet these 3 design targets

### Technologies and Architectural Parameters to be determined

R-CCS

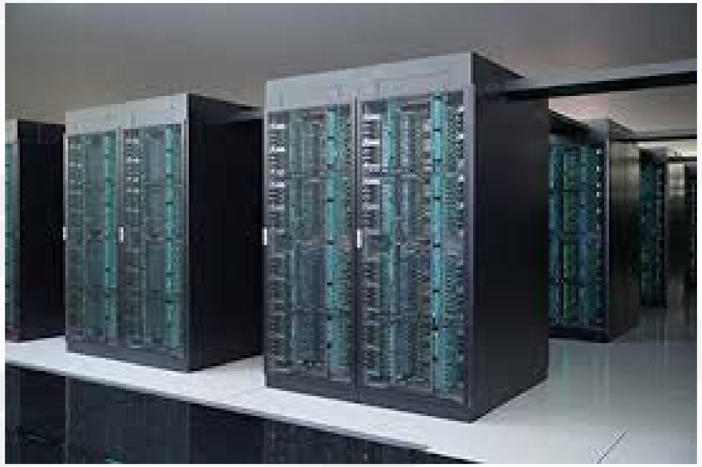
- **Basic Architecture Design (by Feasibility Studies)**
  - Manycore approach, O3 cores, some parameters on chip configuration and SIMD
- **Instruction Set Architecture and SIMD Instructions**
  - Fujitsu collaborated with Arm, contributing to the design of the SVE as a lead partner
- **Chip configuration**
- **Memory technology**
  - DDR, HBM, HMC …
- **Cache structure**
- **Out of order (O3) resources**
- **Enhancement for Target Applications**
- **Interconnect between Nodes**
  - SerDes, topologies "Tofu" or other network?

✓ The number of cores in a CMG
✓ The number of CMGs in a chip
✓ How to connect cores to shared L2 in a CMG
✓ The number of ways, the size, and throughputs of the L1
✓ and L2 caches
✓ The topology of network-on-chip to connect CMGs
✓ The die size of the chip
✓ The number of chips in a node

RIKEN

# Supercomputer “Fugaku” and A64FX processor

- **Ultra-scale “general-purpose” manycore system: 158,976 nodes (1 processor/node, total 7.6 M cores, theoretical peek 537PFLOPS (DP))**
- **Arm-based manycore processor: Fujitsu A64FX (Armv8.2-A SVE 512bit SIMD, #core 48 + 2/4, 3TF@2.0GHz, boost to 2.2GHz)**
  - 12 cores in a cluster of cores called CMG, connected to L2 and HBM memory chips
- **Advanced Memory technology: HBM2 32 GiB, 1024 GB/s bandwidth, packaged in CPU chip**
- **Scalable Interconnect: ToFu-D interconnect**



- Standard programing model is OpenMP-MPI hybrid programming. running each MPI process on a NUMA node (CMG).
- 48 threads OpenMP is also supported.

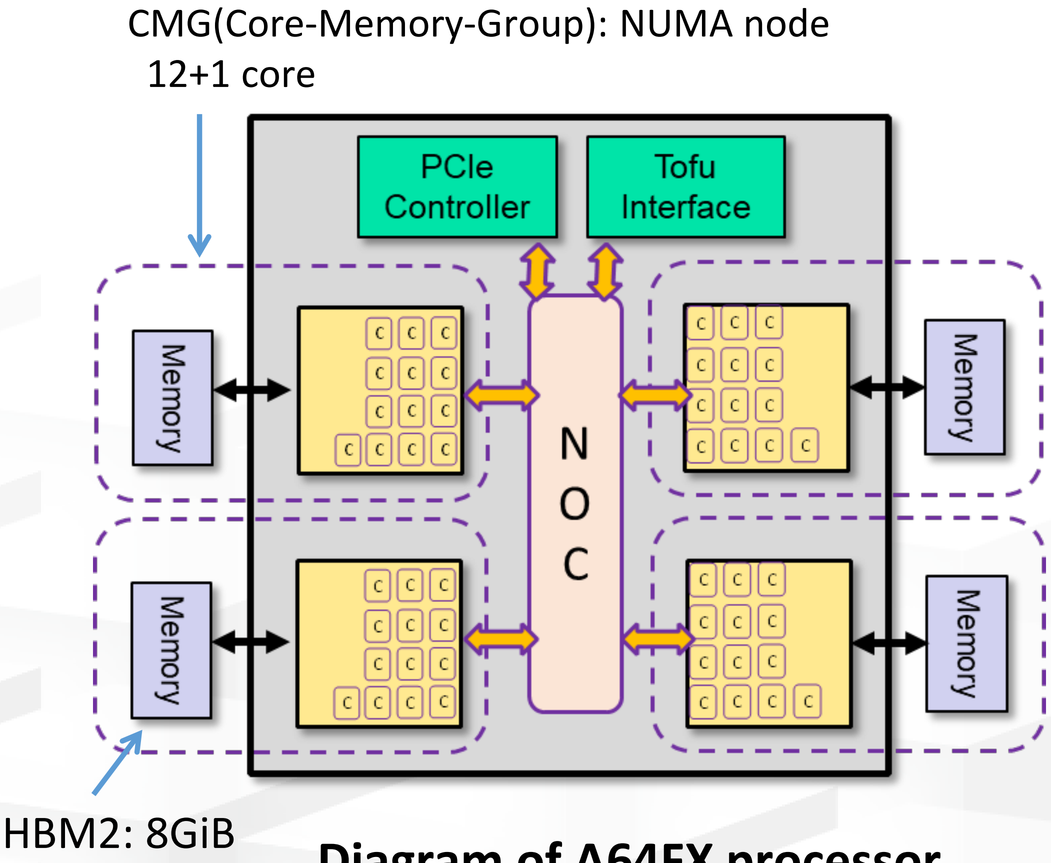
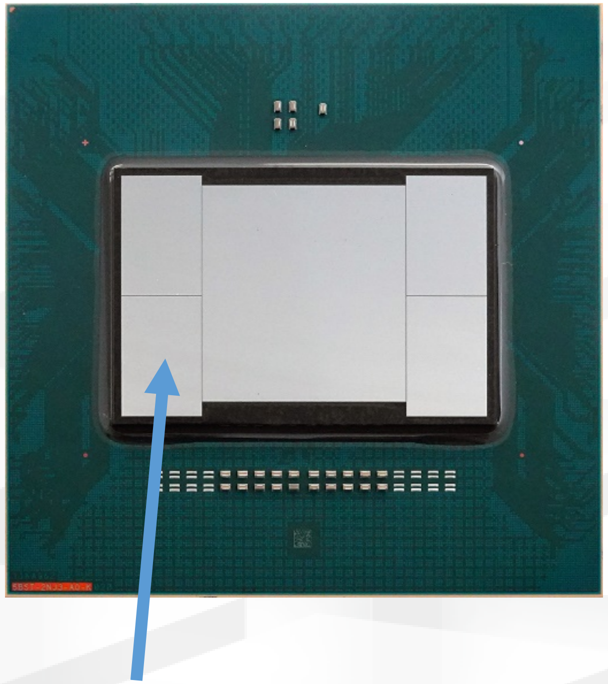
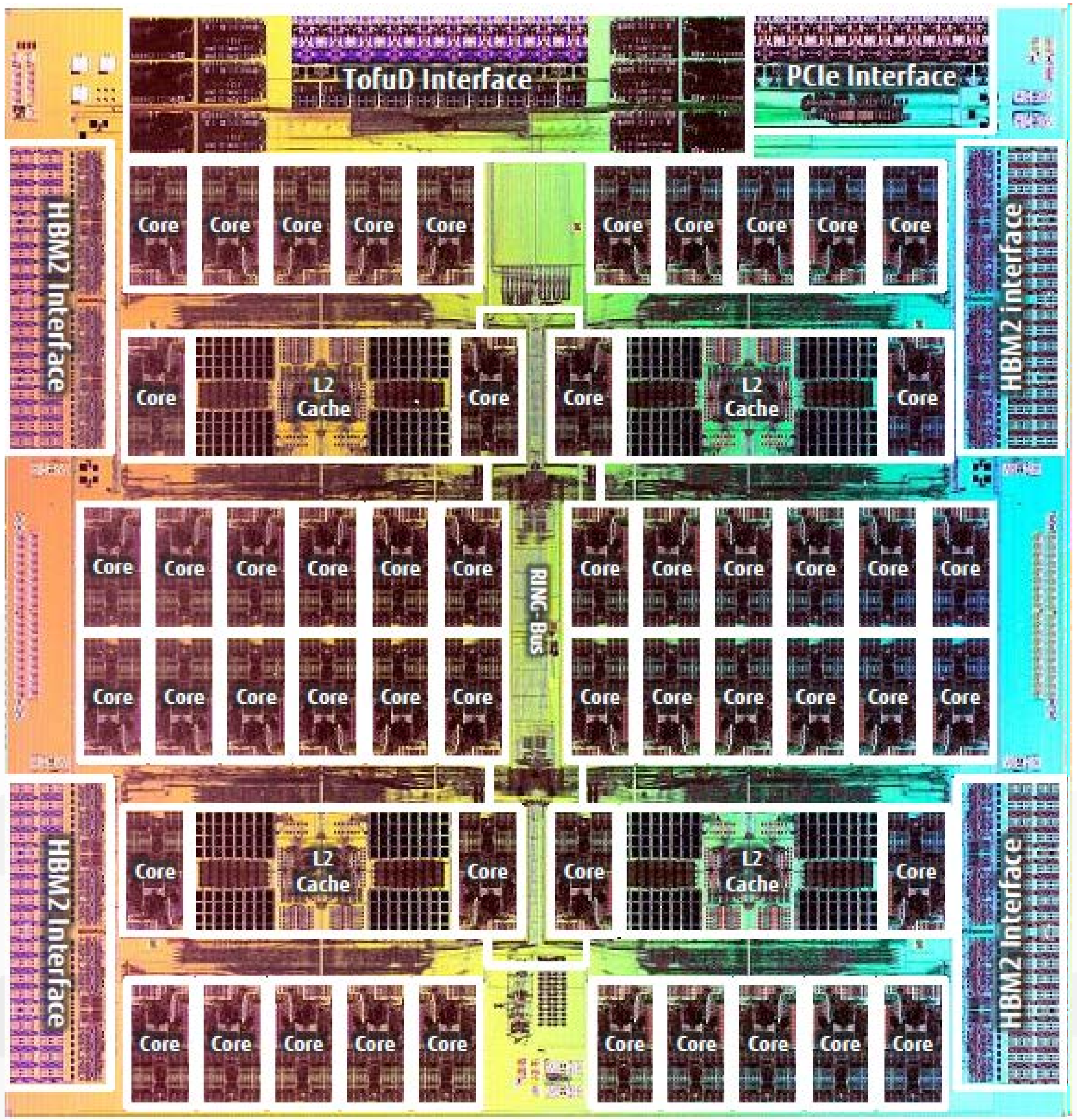



**Diagram of A64FX processor**

# Die Photograph of A64FX processor

- TSMC 7nm FinFET
- 400 mm^2
- HBM2 chips are mounted on Si-interposer connected by TSMC CoWoS technology

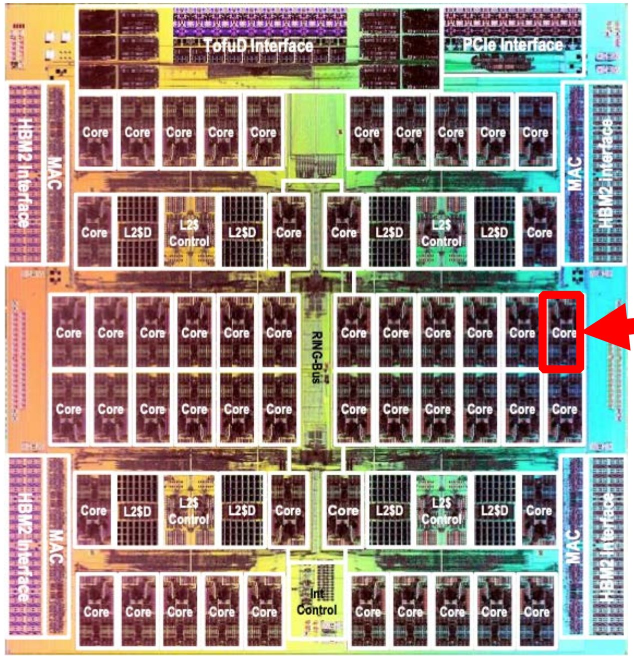


# Comparison of Die-size

- **A64FX: 52 cores (48 cores), 400 mm² die size (8.3 mm²/core), 7nm FinFET process (TSMC)**
- **Xeon Skylake: 20 tiles (5x4), 18 cores, ~485 mm² die size (estimated) (26.9 mm²/core), 14 nm process (Intel)**
- **A64FX core is more than 3 times smaller per core.**

A64FX:
400 mm²
(20 x 20)




Xeon Skylake, High Core Count:
4 x 5 tiles, 18 cores, 2 tiles used for memory interface
485 mm² (22 x 22)

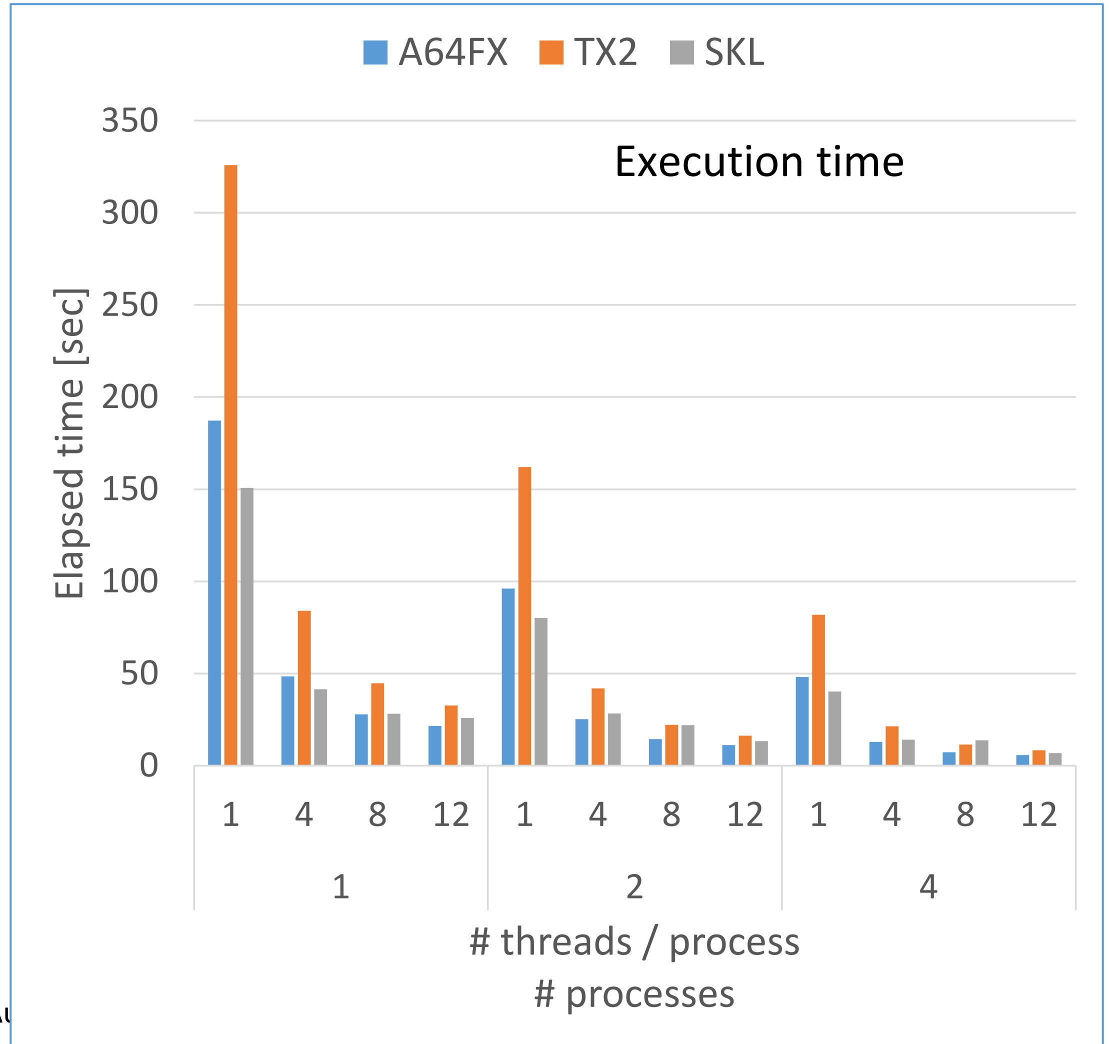
https://www.fujitsu.com/jp/solutions/business-technology/tc/catalog/ff2019-post-k-computer-development.pdf

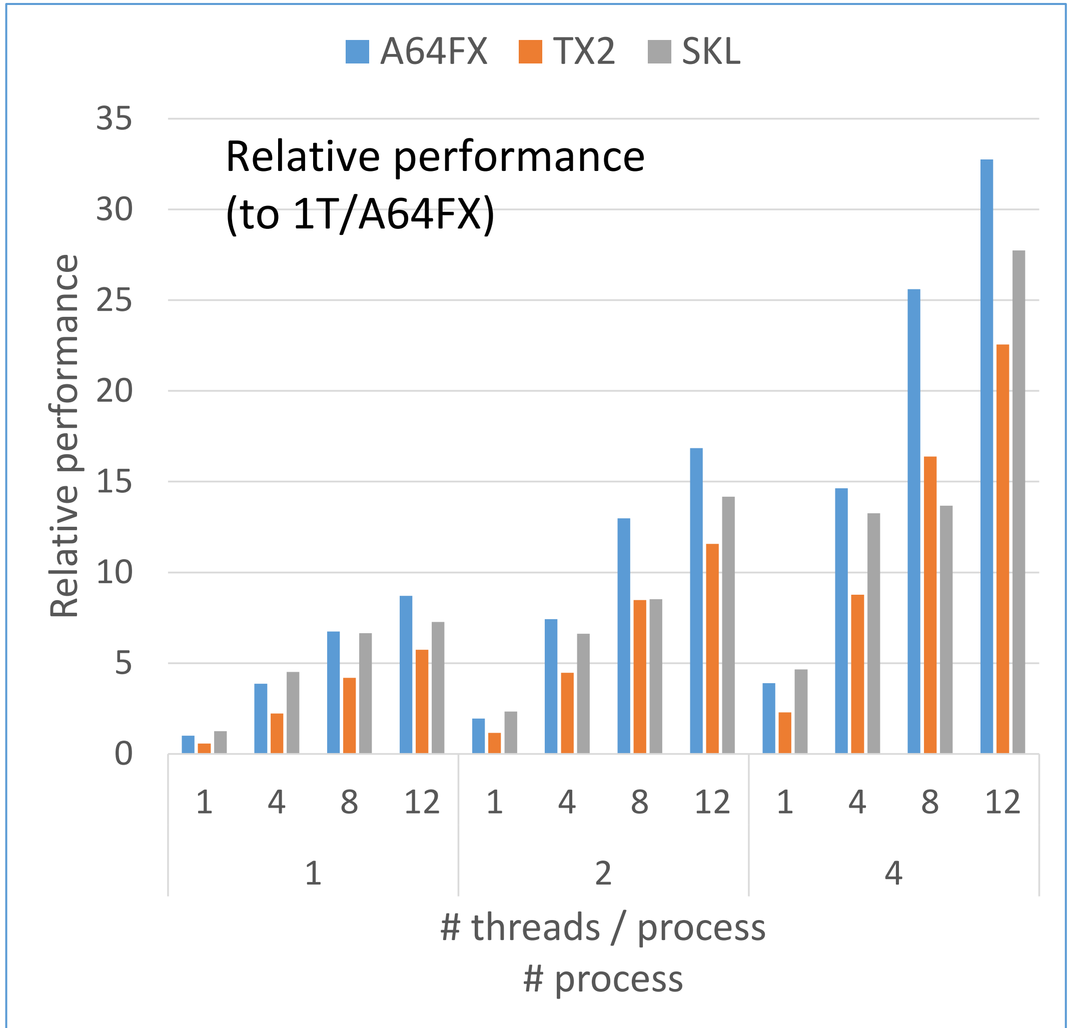
https://en.wikichip.org/wiki/intel/microarchitectures/skylake_(server)

# Benchmark result of CloverLeaf

Taken form UK benchmarks:
A hydrodynamics mini-app to solve the compressible Euler equations in 2D, using an explicit, second-order method

- Comparison with two nodes of TX2 (dual) and Skylake (dual)
- Good scalability by increasing the number of threads within CMG.
- The performance of one A64FX is comparable (better) to that of two nodes (4 sockets) of Skylake

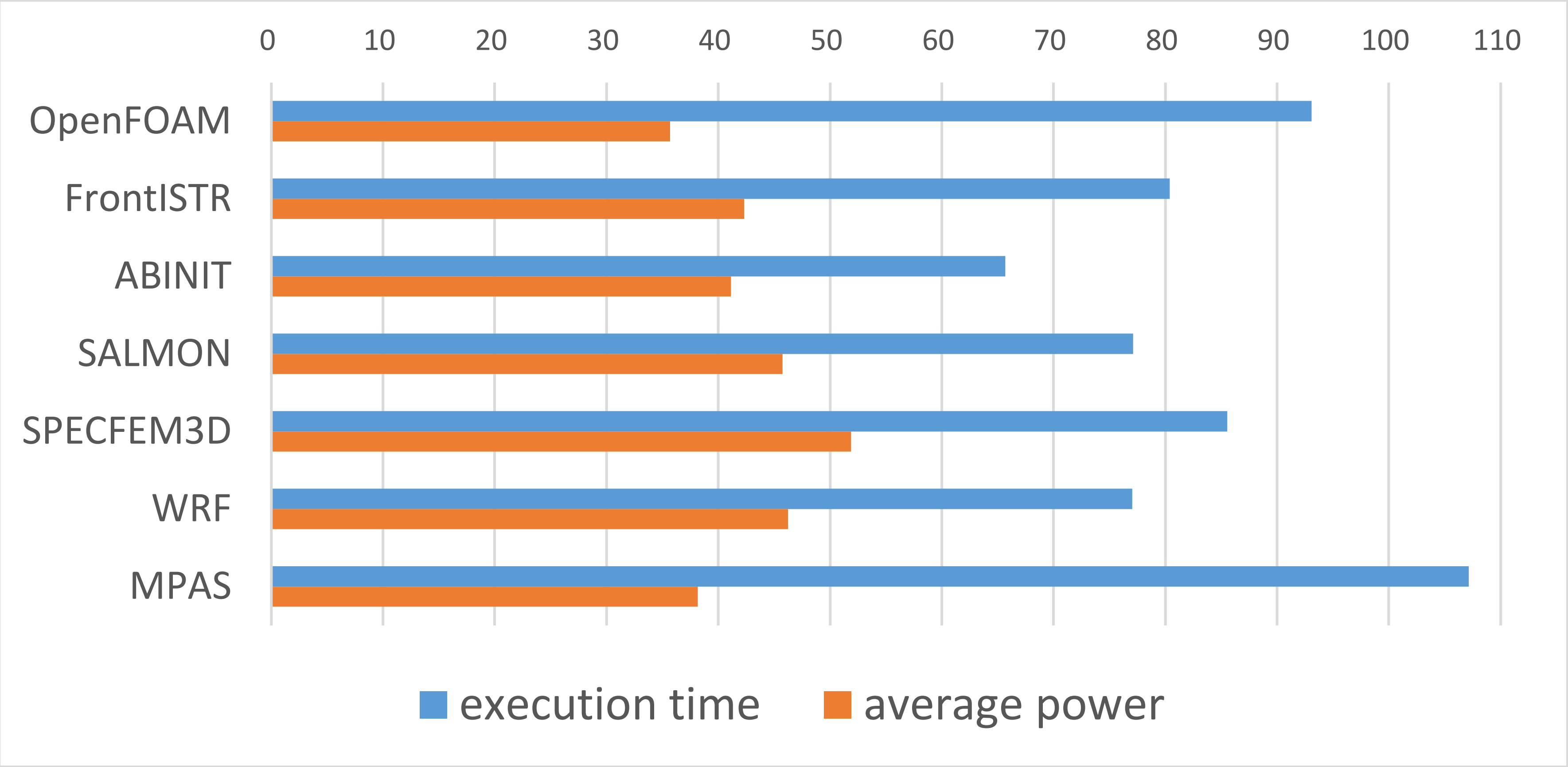


# Performance and Power-efficiency of HPC OSS

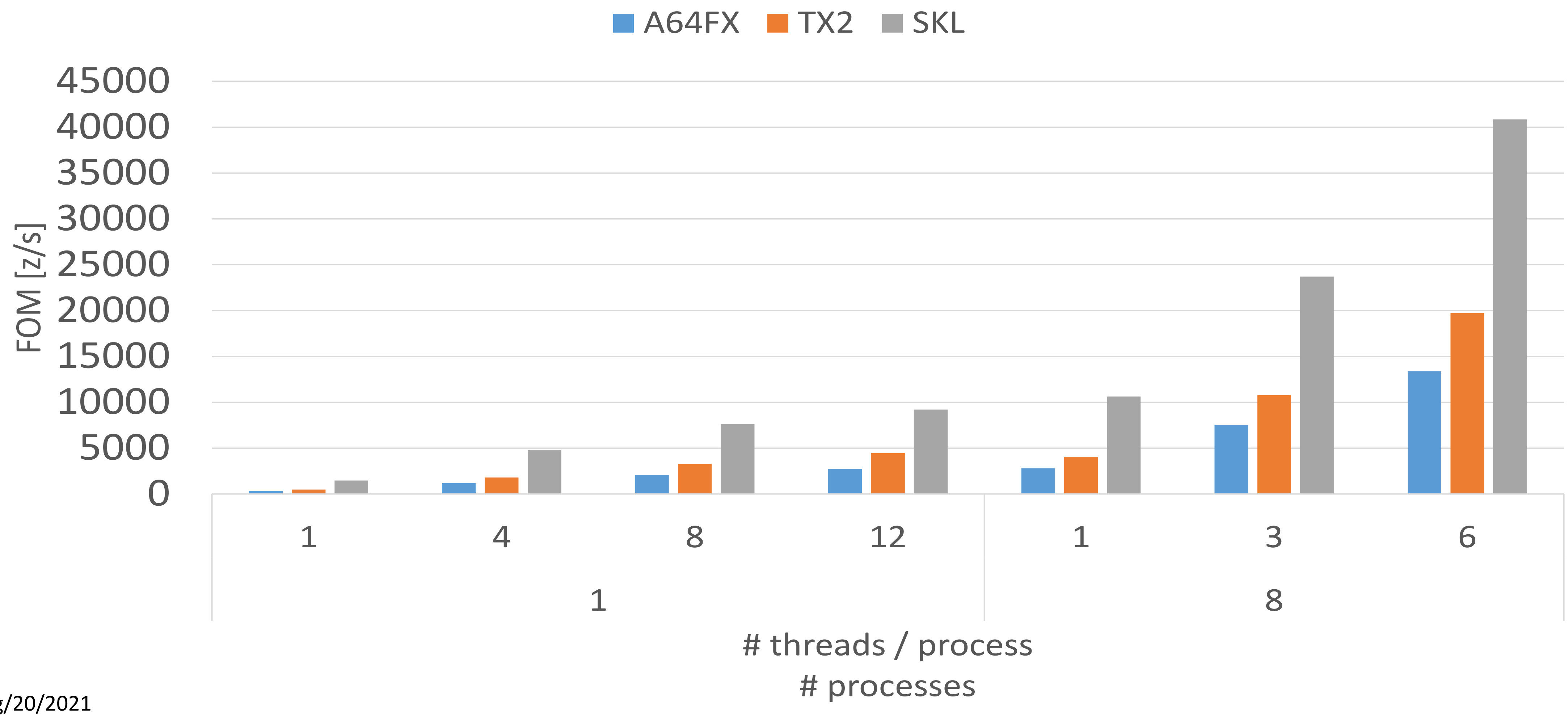
- **Several Open-source software were already ported and evaluated.**
- **Evaluation using one chip A64FX and dual chips of Xeon.**
- **The almost same performance to dual sockets of Xeon with half of power consumption.**

Performance and power efficiency of open-source applications
(results are shown in %, relative to Intel Xeon Platinum 8268 (Cascadelake, 2.90 GHz, 24 cores/socket) (dual sockets))



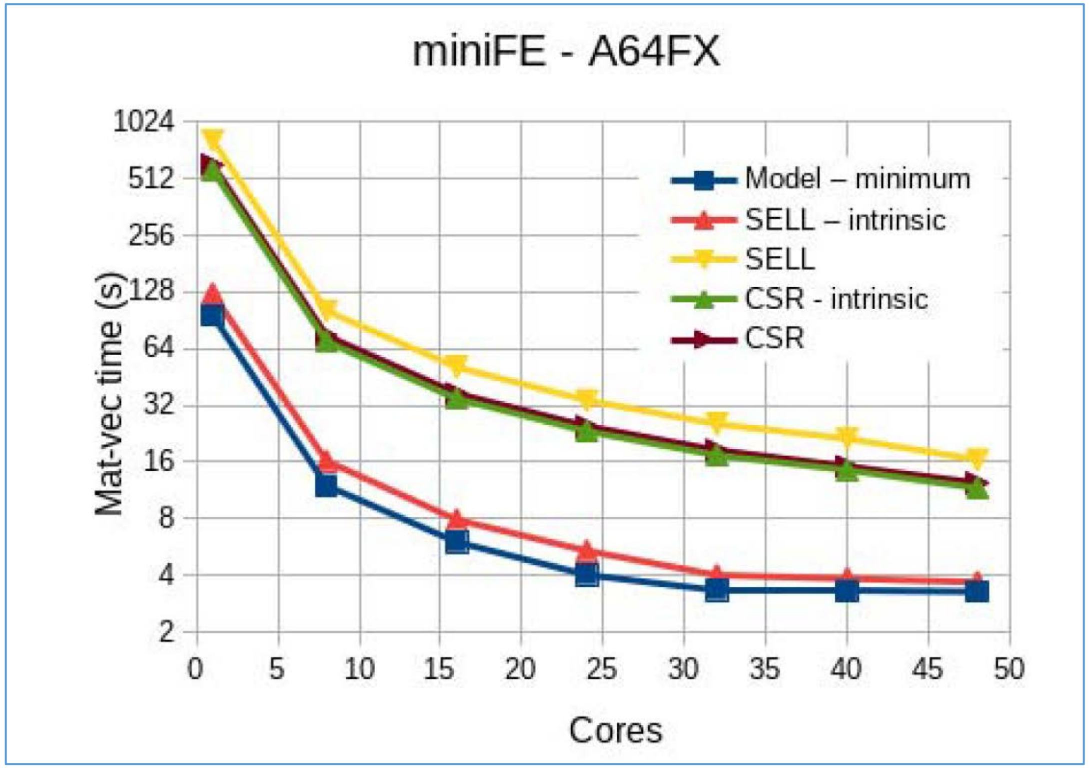
# LULESH

- A64FX performance is less than Thx2 and Intel one
- We found low vectorization (SIMD (SVE) instructions ratio is a few %)
- We need more code tuning for more vectorization using SIMD



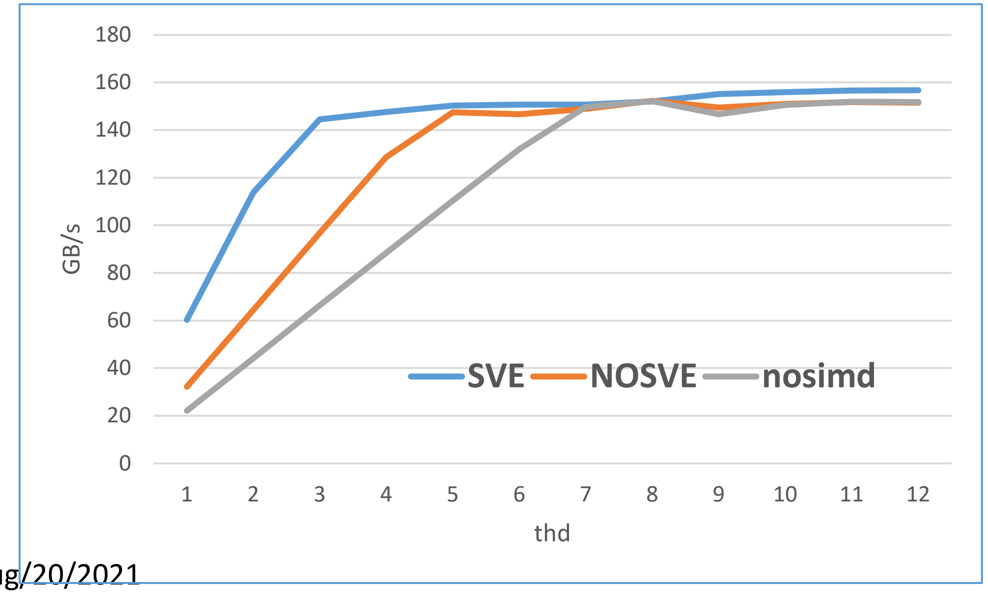
# How to improve the performance of sparse-matrix code

- **Storage format is important:**
  - Sliced ELLPACK format shows significantly better performance than CSR, but only when it is vectorized manually using intrinsics.”
  - CSR is not good even with manual vectorizing.
- **Vectorizing with SVE is important to get memory bandwidth.**




B. Brank, S. Nassyr, F. Pouyan and D. Pleiter, "Porting Applications to Arm-based Processors," EAHPC Workshop, *IEEE CLUSTER* 2020 , Kobe, Japan, 2020, pp. 559-566, doi: 10.1109/CLUSTER49012.2020.00079.

**Memory bandwidth with hardware prefetch**



# SPEC CPU® 2017 integer Speed

- **The performance of A64fX is about ¼ performance of Xeon in single thread.**
  - Fugaku uses normal mode (2.0GHz) with Fujitsu compiler tcsds-1.2.30a. For c and c++, clang mode is used.
  - Xeon is Cisco UCS B200 M5 (Platinum 8168(Skylake), 2.7GHz, 24core x 2 chip, turbo on) with icc 18.0.2.

**https://www.spec.org/cpu2017/results/res2018q2/cpu2017-20180529-06367.txt**

  - Reference machine is UltraSPARC-IV+(2.1GHz, 2cores x 4 chip)

- **The reason for the low single thread integer performance of A64FX is that**
  - the SIMD rate is low in SPEC CPU/int and
  - the frequency and the O3 resource are limited for the throughput-oriented architecture of A64FX.

| | Lang | Threads | A64FX | Xeon |
|---|---|---|---|---|
| 600.perlbench_s | C | 1 | 1.20 | 6.20 |
| 602.gcc_s | C | 1 | 2.63 | 9.57 |
| 605.mcf_s | C | 1 | 3.42 | 11.2 |
| 620.omnetpp_s | C++ | 1 | 1.26 | 7.31 |
| 623.xalancbmk_s | C++ | 1 | 1.61 | 9.46 |
| 625.x264_s | C | 1 | 2.06 | 11.6 |
| 631.deepsjeng_s | C++ | 1 | 1.37 | 5.17 |
| 641.leela_s | C++ | 1 | 1.26 | 4.36 |
| 648.exchange2_s | F90 | 1 | 1.42 | 13.2 |
| 657.xz_s | C/OpenMP | 48 | 8.52 | 23.5 |
| SPECspeed®2017_int_base | | | 1.98 | 9.07 |

COPTIMIZE = -Nclang -Ofast -mcpu=a64fx+sve -ffj-no-fp-relaxed -ffj-eval-concurrent -fsave-optimization-record -fopenmp -Nlst=t -Koptmsg=2
CXXOPTIMIZE = -Nclang -Ofast -mcpu=a64fx+sve -ffj-no-fp-relaxed -ffj-eval-concurrent -fsave-optimization-record -fopenmp -Nlst=t -Koptmsg=2
FOPTIMIZE = -Kfast,openmp -Nlst=t -Koptmsg=2

# SPEC OMP® 2012

- **The performance of A64FX using 48 thread is about 65% performance of Xeon using 56 thread (28 cores).**
  - Fugaku uses normal mode (2.0GHz) with Fujitsu compiler tcsds-1.2.30a. For c and c++, clang mode is used.
  - Xeon is Cisco C240 M5 (Platinum 8280(Cascade Lake), 2.7GHz, 28core x 1chip, hyperthread on (56threads), turbo on) with icc 19.0.1.

https://www.spec.org/omp2012/results/res2019q2/omp2012-20190313-00172.txt

  - Reference machine is Sun Fire X4140 (AMD Opteron 2384, 2.7GHz 4core x 2chips)

- **For some programs (swim and mgrid), A64FX brings extremely good performance due to HBM2.**
- **For 350.md, performance improvement has been confirmed by source code tuning, and we hope that it will be applied by improving the compiler.**

| | Lang | Threads | A64FX | Xeon |
|---|---|---|---|---|
| 350.md | F | 48 | 2.63 | 62.6 |
| 351.bwaves | F | 48 | 15.5 | 11.2 |
| 352.nab | C | 48 | 3.00 | 12.9 |
| 357.bt331 | F | 48 | 5.82 | 16.0 |
| 358.botsalgn | C | 48 | 5.22 | 10.5 |
| 359.botsspar | C | 48 | 3.07 | 6.83 |
| 360.ilbdc | F | 48 | 7.69 | 8.25 |
| 362.fma3d | F | 48 | 4.28 | 11.3 |
| 363.swim | F | 48 | 53.1 | 8.38 |
| 367.imagick | C | 48 | 12.2 | 13.6 |
| 370.mgrid331 | F | 48 | 32.6 | 7.46 |
| 371.applu331 | F | 48 | 8.88 | 14.4 |
| 372.smithwa | C | 48 | 12.8 | 11.8 |
| 376.Kdtree | C++ | 48 | 3.22 | 9.24 |
| SPECompG_base2012 | | | 7.77 | 12.0 |

# Summary of A64FX performance characteristics

- **For core-to-core comparison in intspeed, integer performance is ¼ of Xeon**
- **For chip-to-chip comparison in SPEC OMP, 48 threads performance of one chip is 65% to one chip of recent high-end Xeon (Cascade Lake)**
  - NOTE: Performance of memory-intensive benchmarks is extremely good in A64FX thanks to HBM.
- **For some scientific workload, the almost same performance to dual sockets of Xeon with half of power consumption (UK benchmark and HPC OSS)**
- **High SIMD rate is important to get performance**
  - Need to tune memory access pattern
  - We found many benchmark programs are not well-vecterized.
- **Power efficiency of A64FX is very good (double efficiency than Xeon?)**

# Fugaku System Software Stack

Fugaku AI (DL4Fugaku)
RIKEN: Chainer, PyTorch, TensorFlow, DNNL…

Live Data Analytics
Apache Flink, Kibana, ….

~ 3000 Apps
supported by Spack

Math Libraries
Fujitsu: BLAS, LAPACK, ScaLAPACK, SSL II
RIKEN: EigenEXA, KMATH_FFT3D, Batched BLAS,,,,

Cloud Software Stack
OpenStack, Kubernetis, NEWT...

Open Source
Management Tool
Spack

Compiler and Script Languages
Fortran, C/C++, OpenMP, Java, python, …
(Multiple Compilers supported: Fujitsu, Arm, GNU, LLVM/CLANG, PGI, …)

Batch Job and Management System

Hierarchical File System

ObjectStore
S3 Compatible

Tuning and Debugging Tools
Fujitsu: Profiler, Debugger, GUI

Red Hat Enterprise Linux 8 Libraries

| | | | | |
|---|---|---|---|---|
| High-level Prog. Lang. XMP | Domain Spec. Lang. FDPS | Communication Fujitsu MPI RIKEN MPI | File I/O DTF | Virtualization & Container KVM, Singularity |

| | | |
|---|---|---|
| Process/Thread PIP | Low Level Communication uTofu, LLC | File I/O for Hierarchical Storage Lustre/LLIO |

Red Hat Enterprise Linux Kernel+ optional light-weight kernel (McKernel)

Most applications may work with simple recompile from x86/RHEL environment.
LLNL Spack automates this.

# System software and Programming models & languages for "Fugaku"

- Standard programming model is OpenMP (for NUMA node(CMG)) + MPI
  - Both OpenMPI (by Fujitsu) and MPICH (by Riken) are supported.
  - 4 compilers (Fujitsu, gcc, LLVM/Arm, Cray), OpenMP 4.x is supported.
  - uTofu low-level comm. APIs for Tofu-D interconnect.
- Container and Virtual machine (KVM, Singularity, …)
- DL4Fugaku: AI framework for A64FX and Fugaku, used in Chainer, PyTorch, TensorFlow
- Many Open-source software are already ported using Spack

- System software and Programming tools, Math-Libs developed by RIKEN
  - McKernel: Light-weight Kernel enabling jitter-less environment for large-scale parallel program execution.
  - XcalableMP directive-based PGAS Language
  - FDPS: DLS for Framework for Developing Particle Simulators.
  - EigenExa: Eigen-value math library for large-scale parallel systems.

# Performance Tuning for A64FX processor

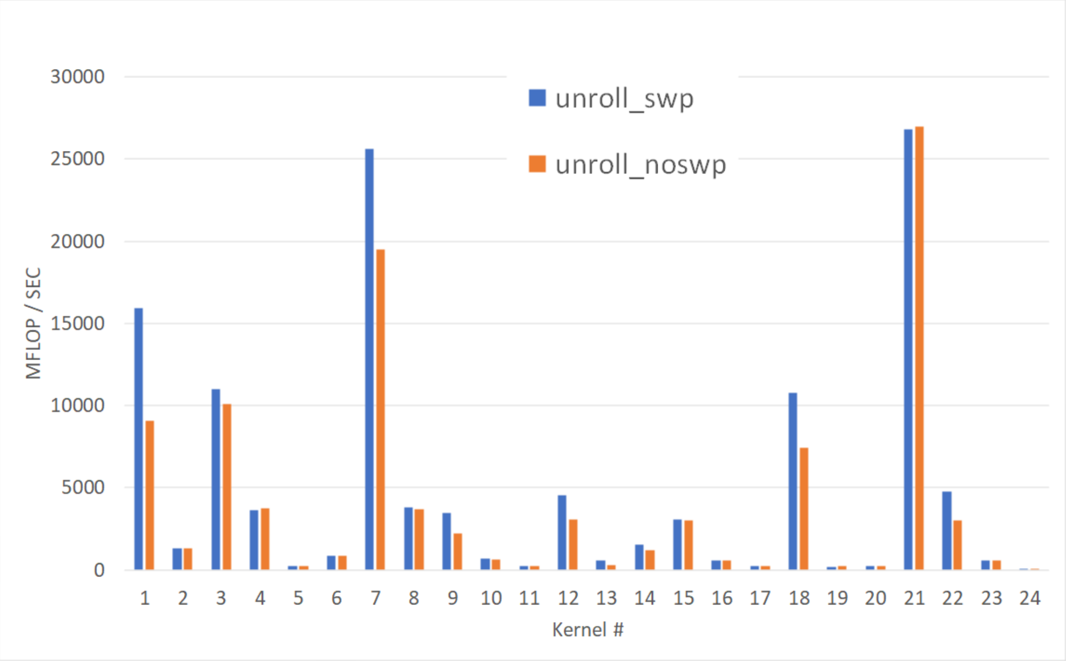
- **HPC-oriented design**
  - Small core ⇒ Less O3 resources
  - (Relatively) Long pipeline
    - 9 cycles for floating point operations
    - Core has only L1 cache
  - High-throughput, but long-latency
  - Pipeline often stalls
    for loops having complex body.

| | A64FX | Skylake |
|---|---|---|
| ReOrder Buffer | 128 entries | 224 entries |
| Reservation Station | 60 (=10x2+20x2) entries | 97 entries |
| Physical Vector Register | 128 (=32 + 96) entries | 168 entries |
| Load Buffer | 40 entries | 72 entries |
| Store Buffer | 24 entries | 56 entries |

A64FX : https://github.com/fujitsu/A64FX
Skylake : https://en.wikichip.org/wiki/intel/microarchitectures/skylake_(server)

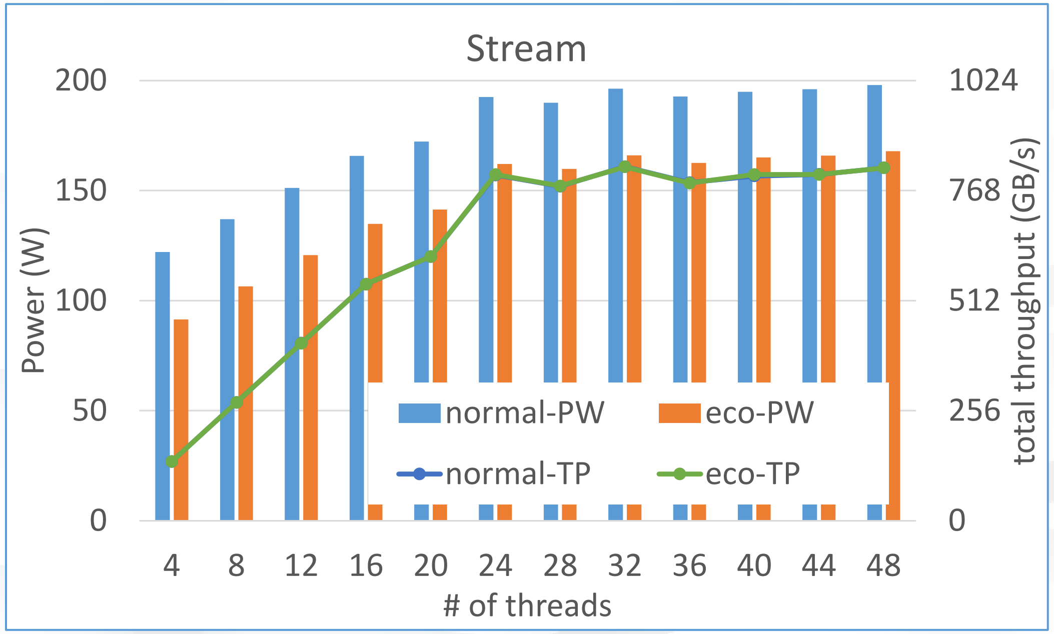
- **Compiler optimization (Fujitsu compiler)**
  - SWP: software pipelining
    - ～ 20% speedup in Livermore Kernels
  - Automatic and Manual loop fissions

**Performance improvement by SWP in Livermore Kernels by Fujitsu compiler**

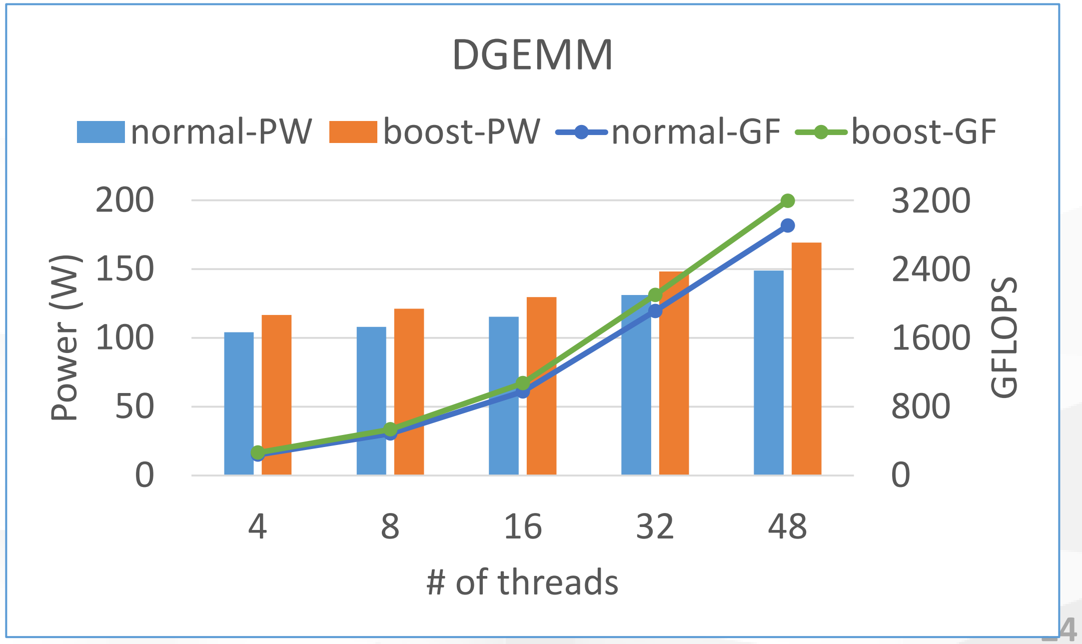


# Evaluation power mode: Boost mode (2.2GHz) & Eco mode (1 SIMD pipeline)

- **Power & Performance of STREAM using Eco mode**
  - The performance is almost the same as that in normal mode (24 threads hits 80% of peak memory bandwidth
  - The power increases upto 24 threads.
  - 15%-25% reduction comparing to that in normal mode.




- **Power & Performance of DGEMM (in Fujitsu Lib) using Boost mode**
  - Reach to 95% out of peak performance
  - The performance is 10% better than that in normal mode.
  - The power increases by 13.7%
  - The power-efficiency decreases by 3.3 %



# Concluding remarks

- **We have confirmed that 3 KPIs were achieved:**
  - Power-efficiency ⇒ Actually, Fugaku is running round at 20MW with 70% utilization
  - Effective Performance of applications. ⇒ Many apps are running more efficient than expected.
  - Ease-of-use ⇒ easy for porting OpenMP+MPI programs without any accelerator programming.

- **A64FX is a manycore processor designed for HPC workload.**
  - Many and small core, and less O3 resource to reduce die-size.
  - For Core-to-core comparison, the integer performance is relatively low.
  - For Chip-to-chip comparison, the floating point performance is comparable to Intel chip in some benchmarks.
    - Performance of memory-intensive benchmarks is extremely good in A64FX thanks to HBM.
  - Very good power-efficiency (double of Intel one)

# Results from Fugaku

- **Performance of Target apps**
  - The target performance has been achieved
- **Power consumption**
  - Lower power consumption than we estimated
    - Almost apps use boost/no-eco
  - In daily operation, system power consumption is around 20MW with 70% utilization

Target applications

Performance relative to the K computer

Power consumption

| アプリケーション | 利用形態 | 問題規模 | ノード数／ジョブ | 性能倍率 | 消費電力 |
|---|---|---|---|---|---|
| GENESIS | 多重 | 92,224原子 | 1 | 131 倍 | 22 MW |
| GENOMON | 多重 | リード長150、14億リード(ペアードエンド) | 96 | 23 倍 | 20 MW |
| GAMERA | 大規模単一 | 1兆自由度 | 147,456 | 63 倍 | 21 MW |
| NICAM+ LETKF | 大規模単一 | 全球3.5kmメッシュ、1024メンバENS同化 | 131,072 | 127 倍 | 22 MW |
| NTChem | 多重 | 720原子、19,680原子軌道 | 17,820 | 70 倍 | 26 MW |
| ADVENTURE | 多重 | 16.5億自由度 | 4,096 | 63倍 | 28 MW |
| RSDFT | 多重 | 110,592原子、221,184バンド | 10,368 | 38 倍 | 30 MW |
| FFB | 大規模単一 | 6,748億要素 | 158,976 | 51 倍 | 29 MW |
| LQCD | 大規模単一 | 192^4格子 | 147,456 | 38 倍 | 20 MW |