# REPORT ON THE FUJITSU FUGAKU SYSTEM

Tech Report No. ICL-UT-20-06

Jack Dongarra

University of Tennessee, Knoxville
Oak Ridge National Laboratory

University of Manchester

June 22, 2020

# Overview

The Fugaku compute system was designed and built by Fujitsu and RIKEN. Fugaku 富岳, is another name for Mount Fuji, created by combining the first character of 富士, *Fuji*, and 岳, *mountain*. The system is installed at the RIKEN Center for Computational Science (R-CCS) in Kobe, Japan. RIKEN is a large scientific research institute in Japan with about 3,000 scientists in seven campuses across Japan. Development for Fugaku hardware started in 2014 as the successor to the K computer. The K Computer mainly focused on basic science and simulations and modernized the Japanese supercomputer to be massively parallel. The Fugaku system is designed to have a continuum of applications ranging from basic science to Society 5.0, an initiative to create a new social scheme and economic model by fully incorporating the technological innovations of the fourth industrial revolution. The relation to the Mount Fuji image is to have a broad base of applications and capacity for simulation, data science, and AI—with academic, industry, and cloud startups—along with a high peak performance on large-scale applications.



Figure 1. Fugaku System as installed in RIKEN R-CCS

The Fugaku system is built on the A64FX ARM v8.2-A, which uses Scalable Vector Extension (SVE) instructions and a 512-bit implementation. The Fugaku system adds the following Fujitsu extensions: hardware barrier, sector cache, prefetch, and the 48/52 core CPU. It is optimized for high-performance computing (HPC) with an extremely high bandwidth 3D stacked memory, 4x 8 GB HBM with 1024 GB/s, on-die Tofu-D network BW (~400 Gbps), high SVE FLOP/s (3.072 TFLOP/s), and various AI support (FP16, INT8, etc.). The A64FX processor provides for general purpose Linux, Windows, and other cloud systems. Simply put, Fugaku is the largest and fastest supercomputer built to date. Below is further breakdown of the hardware.

- Caches:
    - L1D/core: 64 KB, 4way, 256 GB/s (load), 128 GB/s (store)
    - L2/CMG: 8 MB, 16 way
    - L2/node: 4 TB/s (load), 2 TB/s (store)
    - L2/core: 128 GB/s (load), 64 GB/s (store)
- 158,976 nodes

- o 384 nodes × 396 racks = 152,064
- o 192 nodes × 36 racks = 6,912
- 4.85 PB of total memory
- 163 PB/s memory bandwidth
- Tofu-D 6D Torus network, 6.49 PB/s injection bandwidth (
  - o (28 Gbps × 2 lanes × 10 ports)
- 15.9 PB of NVMe L1 storage
- PCIe Gen3 ×16
- Many-endpoint 100 Gbps I/O network into Lustre
- The first "exascale" machine (not for 64-bit floating point but in application performance).

Work on the Fugaku system began as the "Post K" computer. The origins date back to 2006 with planning for a follow on to the K Computer system in Kobe.

The deployment of the Fugaku system was done in a pipelined fashion. The first rack was shipped on December 3, 2019, and all racks were on the floor by May 13, 2020. Early users had access to the system in the first quarter of 2020.
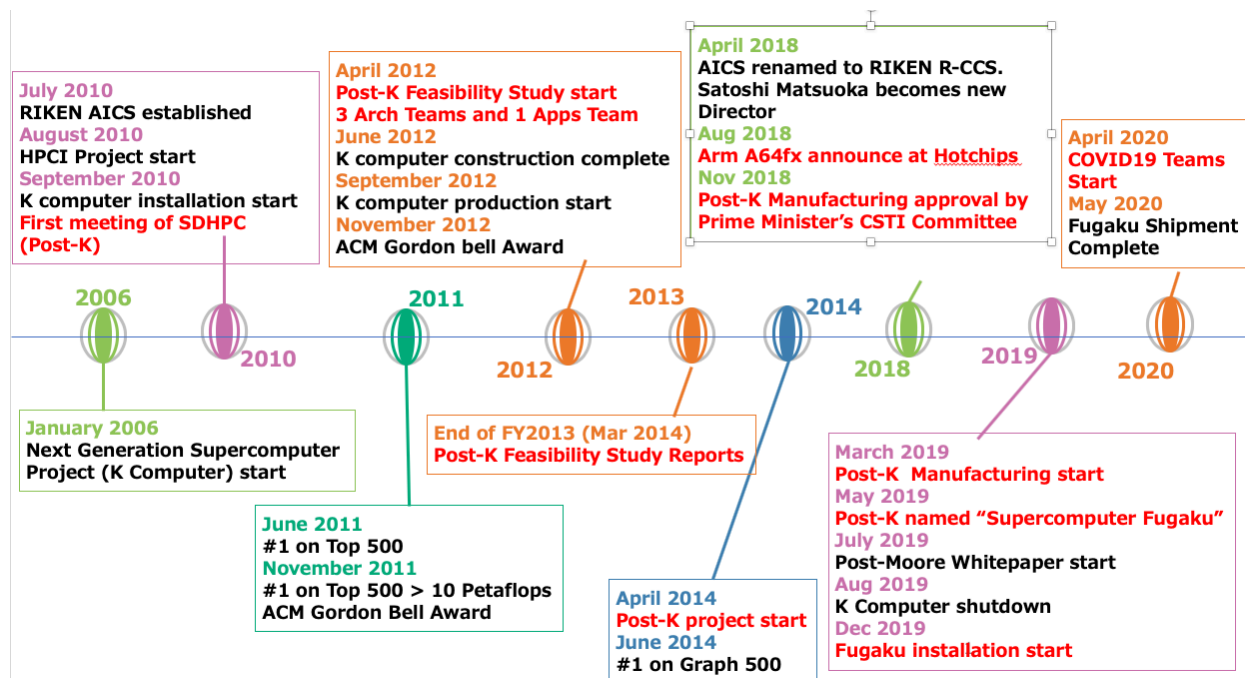


Figure 2. Fugaku Deployment Timetable

Working with Fujitsu to co-design a processor began in 2011 with a goal of a 100× speedup over existing applications on the K Computer.

The CPU is based on ARM architecture version 8.2A and adopts the SVE instructions. Fujitsu is the manufacturer, and the CPU chip is based on a TSMC 7 nm FinFET & CoWoS technologies

using Broadcom SerDes, HBM I/O, and SRAMS. The total number of transistors is 87.86 billion with 594 signal pins.

The A64FX processor is a many-core ARM CPU with 48 compute cores and 2 or 4 assistant cores used by the operating system. It uses a new core design with the ARM V8, 64-bit ARM ecosystem, Tofu-D interconnect and PCIe Gen3 ×16 external connections. While the processor does not have a GPU accelerator, it has the SVE 512-bit × 2 vector extensions that can operate at integers 1, 2, 4, and 8 bytes and floating point 16, 32, and 64 bit levels. The on-package memory is HBM2 with massive memory bandwidth with a bytes-per-DP floating point ratio of 0.4 bytes per FLOP and is capable of streaming memory, strided, and gather-scatter accesses.
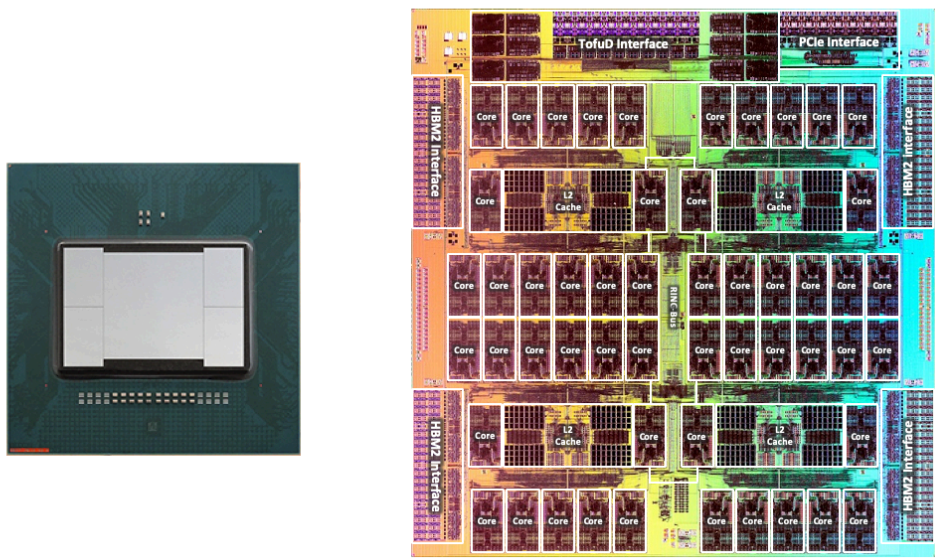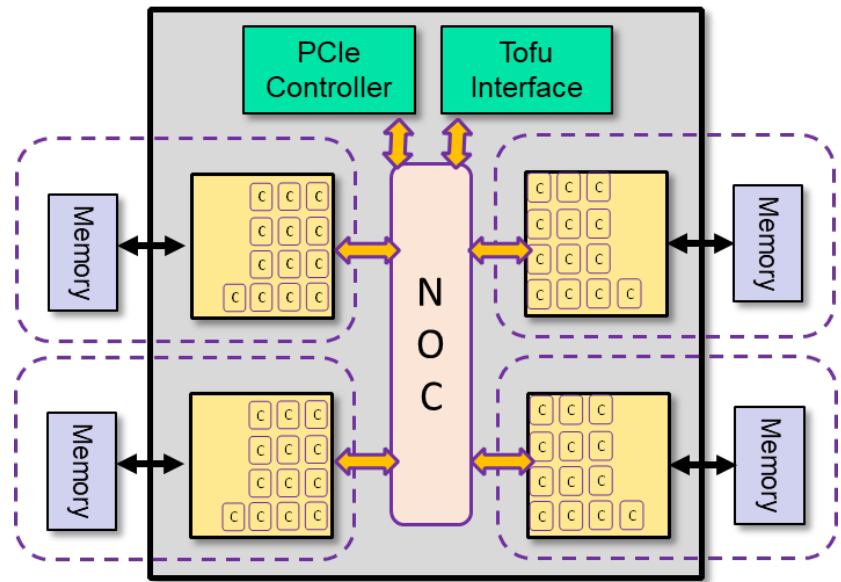




Figure 3. A64FX Processor Layout

# RIKEN Kobe Campus

The Kobe Campus opened in April 2002 as a central research center of the Kobe Biomedical Innovation Cluster promoted by the city of Kobe. The Campus is recognized as an international research center for performing research on developmental biology for multicell organisms, developing a platform for health promotion, and conducting research in areas such as computational science and computer science. As a research center, more than 1,200 scientists and assistants are working to resolve various issues affecting society through collaborating with medical institutions, research institutes, and corporations. The R-CCS is the leading research center in HPC and computational science in Japan, operated the K computer from 2012–2019, and now is running the Fugaku computer. Their objectives are to investigate "the science of (high performance) computing, by computing, and for computing," to promote the core competence of the research in a concrete fashion as open-source software, and to collaborate with many other leadership centers around the globe.

## Fugaku System Configuration

The Fugaku machine includes a total of 158,976 nodes, with 384 nodes per rack × 396 (full) racks = 152,064 nodes and 192 nodes/rack × 36 (half) racks = 6,912 nodes. By comparison, the K Computer had 88,128 nodes.
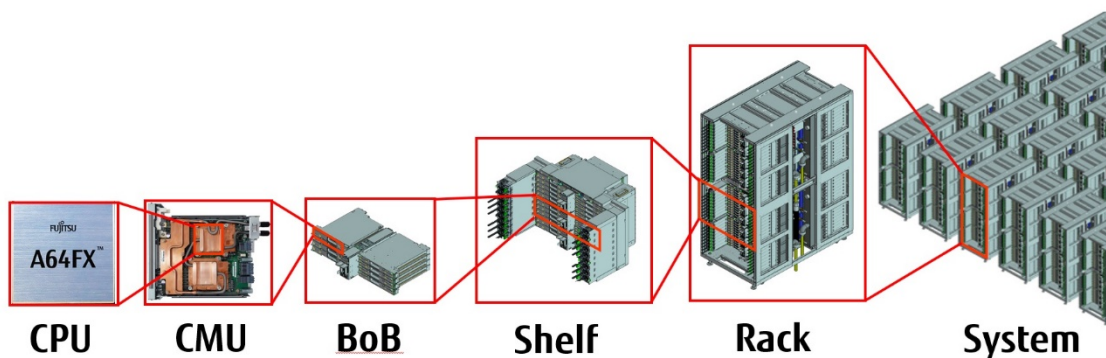


Figure 4. System Configuration

| Unit | # of nodes | Description |
|---|---|---|
| CPU | 1 | Single-socket node with HBM2 and Tofu interconnect D |
| CMU | 2 | CPU Memory Unit: 2× CPU |
| BoB | 16 | Bunch of Blades: 8× CMU |
| Shelf | 48 | 3× BoB |
| Rack | 384 | 8× Shelf |
| System | 152,064 | As a Fugaku system |

Figure 5. System Characteristics

In terms of theoretical peak performance, the following characteristics are given:

- Normal Mode (CPU Frequency of 2 GHz)
    - 64-bit, double-precision FP: 488 PFLOP/s
    - 32-bit, single-precision FP: 977 PFLOP/s
    - 16-bit, half-precision FP (AI training): 1.95 EFLOP/s
    - 8-bit integer (AI Inference): 3.90 Exaops
- Boost Mode (CPU Frequency of 2.2 GHz)
    - 64-bit, double-precision FP: 537 PFLOP/s
    - 32-bit, single-precision FP: 1.07 EFLOP/s
    - 16-bit, half-precision FP (IEEE Float, AI training): 2.15 EFLOP/s
    - 8-bit integer (AI Inference): 4.30 Exaops
- Theoretical Peak Memory Bandwidth: 163 Petabytes/s

The basic node level performance shows:

- Stream triad: 830+ GB/s
- DGEMM: 2.5+ TFLOP/s

When compared to the K Computer, we see the following improvements:

- 64-bit, double-precision FP: 48× speedup
- 32-bit, single-precision: 95× speedup
- 16-bit, half-precision (AI training): 190× speedup
- K Computer theoretical peak: 11.28 PFLOP/s (all precisions)
- 8-bit integer (AI Inference): >1,500× speedup
- K Computer theoretical peak: 2.82 PFLOP/s (64-bit FP)

- Theoretical peak memory bandwidth: 29× speedup
- K Computer theoretical peak: 5.64 PB/s

## Storage System

The storage system consists of three primary layers:

- 1st Layer
    - Cache for global file system
    - Temporary file systems
        - Local file system for compute node
        - Shared file system for a job
- 2nd Layer
    - Lustre-based global file system
- 3rd Layer
    - Cloud storage services (in preparation)

# Power Efficiency and Cooling

The peak power consumption under load (running the HPL benchmark) is 28.33 MW or 14.7 GFLOP/s per watt. This includes all components, such as PSU, FANs in the rack. The cooling solution uses a closed-coupled chilled configuration with a custom water-cooling unit. The CPU Memory Unit (CMU) is also water cooled.
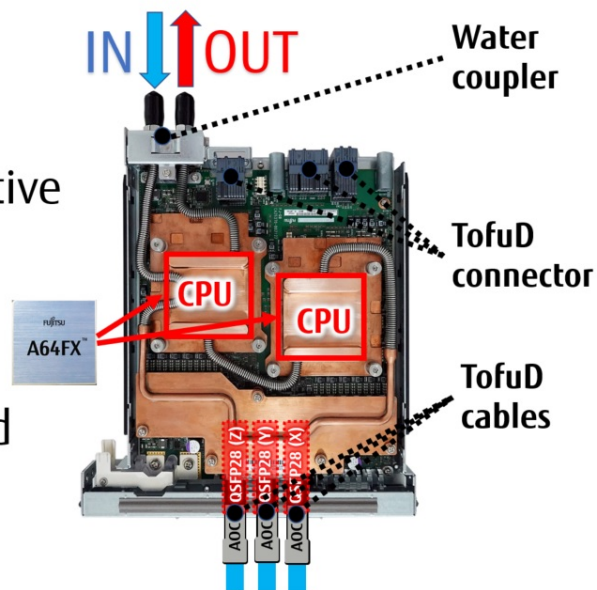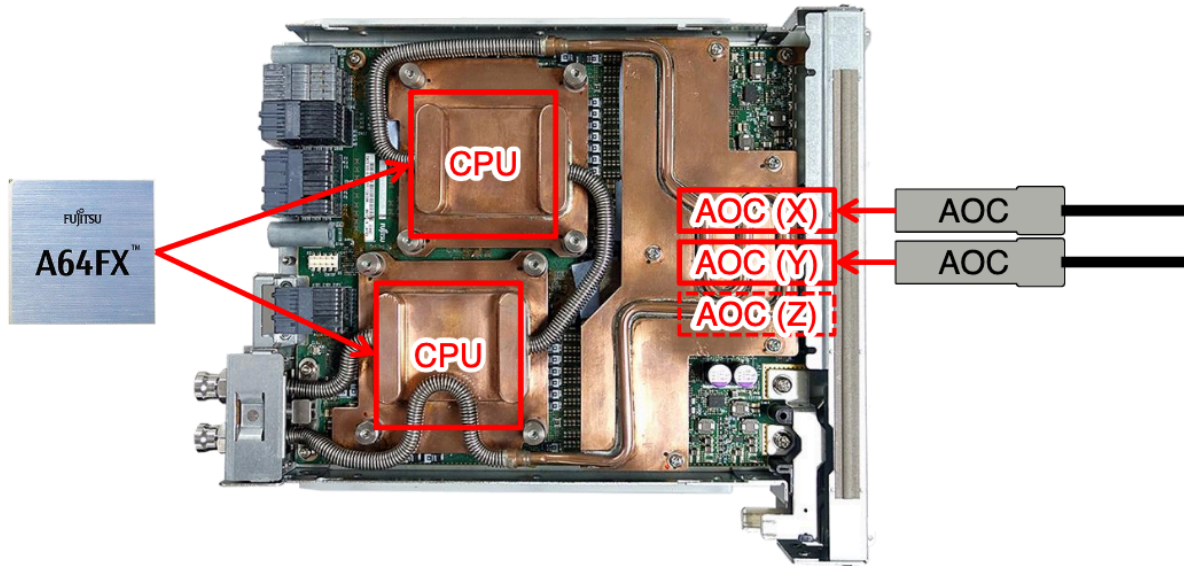
Figure 6. Node board

## Tofu Interconnect Network

Tofu stands for "torus fusion" and represents the designed combination of dimensions with an independent configuration and a routing algorithm. The letter D represents high "density" node and "dynamic" packet slicing for "dual-rail" transfer. The 6-D mesh/torus network of Tofu achieves high scalability for the compute nodes, and the virtual 3-D torus rank mapping scheme provides both high availability and topology-aware programmability.

A node address in the physical 6-D network is represented by six-dimensional coordinates X, Y, Z, A, B, and C. The A and C coordinates can be 0 or 1, and the B coordinate can be 0, 1, or 2. The range of the X, Y, and Z coordinates depends on the system size.

Two nodes the coordinates of which are different by 1 in one axis and identical in the other five axes are "adjacent" and are connected to each other. When a certain axis is configured as a torus, the node with coordinate 0 in the axis, and the node with the maximum coordinate value are connected to each other. The A- and C-axes are fixed to the mesh configuration, and the B-axis is fixed to the torus configuration. Each node has 10 ports for the 6-D mesh/torus network. Each of the X-, Y-, Z-, and B-axes uses two ports, and each of the A- and C-axes uses one port. Each link provides 5.0 GB/s peak throughput. Each link has 8 lanes of high-speed differential I/O signals at a 6.25-Gbps data rate.

Tofu was implemented as an interconnect controller (ICC) chip with 80 lanes of signals for the network. The table below shows the comparison of node and link configurations within the Tofu family. TofuD uses a high-speed signal of 28-Gbps data rate that is approximately 9% faster than that of Tofu2. However, owing to the reduction of the number of signals, TofuD reduces the link bandwidth to 6.8 GB/s, which is approximately 54% for Tofu2. To compensate for the reduction in the link bandwidth, TofuD increases the number of simultaneous communications from 4 of Tofu2 to 6. The injection rate of TofuD is enhanced to approximately 80% of that of Tofu2. There are six adjacent nodes in the virtual 3-D torus, and therefore topology-aware algorithms

can use six simultaneous communications effectively. Tofu and Tofu2 were the previous interconnect networks used on the K computer and the Fujitsu PRIMEHPC FX100 system.

|  | Tofu | Tofu2 | TofuD |
|---|---|---|---|
| Number of signal lanes per node | 80 | 40 | 20 |
| Data rate (Gbps) | 6.25 | 25.78125 | 28.05 |
| Link bandwidth (GB/s) | 5.0 | 12.5 | 6.8 |
| Number of TNIs per node | 4 | 4 | 6 |
| Injection bandwidth per node (GB/s) | 20 | 50 | 40.8 |

Figure 7. Tofu D Interconnect

## Peak Performance

|  | Fugaku Normal Mode (2.0 GHz) | Fugaku Boost Mode (2.2 GHz) | K Computer |
|---|---|---|---|
| **Peak Double Precision (64 bit)** | 488 PFLOP/s | 537 PFLOP/s | 11.3 PFLOP/s |
| **Peak Single Precision (32 bit)** | 977 PFLOP/s | 1.07 EFLOP/s | 11.3 PFLOP/s |
| **Half Precision (16-bit float, IEEE standard)** | 1.95 EFLOP/s | 2.15 EFLOP/s | -- |
| **Integer (8 bit)** | 3.9 EFLOP/s | 4.3 EFLOP/s | -- |
| **Total Memory** | 4.85 PB | -- |  |
| **Total Memory Bandwidth** | 163 PB/s | -- | 5.18 TB/s |

Figure 8. Peak Performance

# High Performance LINPACK, HPCG, and HPL-AI Benchmarks

The Fugaku system achieved an HPL number of 415 PFLOP/s on 152,064 nodes (potential peak is 514 PFLOP/s). (This is 96% of the full system, the full system has 158,976 nodes.) As a result, they will be the new #1 system on the TOP500. The power efficiency is at 14.7 GFLOP/s per watt. The system also achieved an HPCG number of 13 PFLOP/s, and so Fugaku will be #1 on the HPCG list as well.

Notably, Fugaku's HPL number is 80.9% of the peak performance, and their HPCG is 2.8% of peak. For comparison, the US Department of Energy's (DOE's) Summit system at Oak Ridge National Laboratory (ORNL) achieved an HPL score of 149 PFLOP/s, which is 74% of peak performance at 14.7 GFLOP/s per watt, and the HPCG number stands at 2.93 PFLOP/s, which is 1.3% of peak.

The HPL-AI benchmark seeks to highlight the emerging convergence of high-performance computing (HPC) and artificial intelligence (AI) workloads. While traditional HPC focused on simulation runs for modeling phenomena in physics, chemistry, biology, and so on, the mathematical models that drive these computations require, for the most part, 64-bit accuracy. On the other hand, the machine learning methods that fuel advances in AI achieve desired results at 32-bit and even lower floating-point precision formats. The HPL-AI drops the requirement of 64-bit computation throughout the entire solution process and instead opting for low-precision (likely 16-bit) accuracy for LU, and a sophisticated iteration to recover the accuracy lost in factorization. The iterative method guaranteed to be numerically stable is the generalized minimal residual method (GMRES), which uses application of the L and U factors to serve as a preconditioner. The combination of these algorithms is demonstrably sufficient for high accuracy and may be implemented in a way that takes advantage of the current and upcoming devices for accelerating AI workloads.

For the HPL-AI benchmark Fugaku achieves a stunning 1.42 Exaflops/s and is number 1 in the performance. For comparison the DOE ORNL Summit system achieves .55 Exaflop/s on the benchmark and is number 2.

## The Software Stack

The Spack package manager will be used to manage open-source software on Fugaku. Spack is a package manager for HPC, Linux, and macOS intended to make installing scientific software much easier. Spack is not tied to a particular language; the user can build a software stack in Python or R, link to libraries written in C, C++, or Fortran, and easily swap compilers or target specific microarchitectures.

Fugaku users can easily use pre-installed packages and build packages based on Spack recipes. The following list shows the results of building/compiling packages for A64FX according to the Spack recipes. Note that the results in this list do not guarantee that each package will work properly. On the other hand, Fujitsu will provide packages compiled with the Fujitsu compiler on Fugaku as "external" packages, of which Spack can be made aware.

- OpenJDK11
- Ruby 2.6.5 or later
- Python2 2.7.15
- Python3 3.6.8
- Numpy 1.14.3
- SciPy 1.0.0

- Eclipse IDE 2019-09 R Packages

A number of other software packages are also available:

- DL4Fugaku
- Chainer
- TensorFlow
- BLAS
- LAPACK
- ScaLAPACK
- SSL II
- EigenEXA
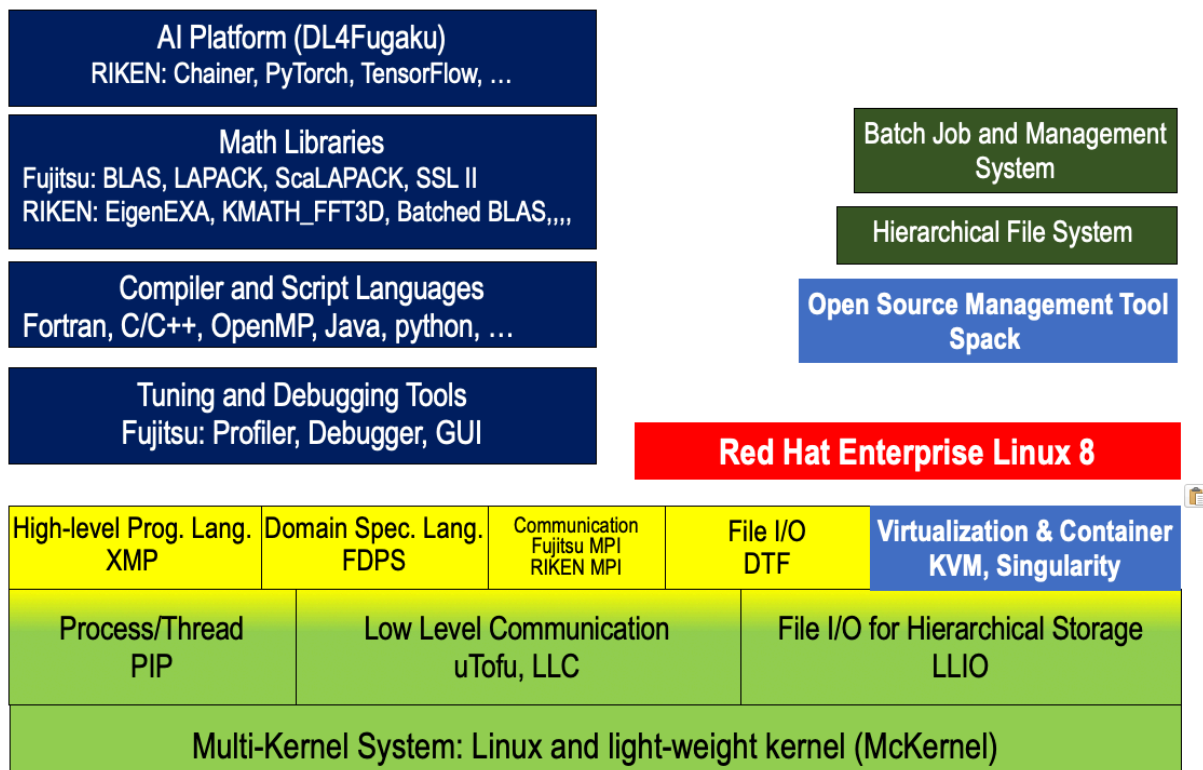- KMATH_FFT3D
- Batched BLAS



Figure 9. Fugaku Software Stack

The operating system is Red Hat Enterprise Linux 8 & McKernel (light-weight multi kernel operating system).

The MPI on the system is Fujitsu MPI (based on OpenMPI) and RIKEN-MPICH (based on MPICH).

# Commercial Applications

Fujitsu works with vendors to make commercial apps, third party, independent software vendor (ISV) packages, available for FX1000, also for FX700 and Fugaku with binary compatibility:
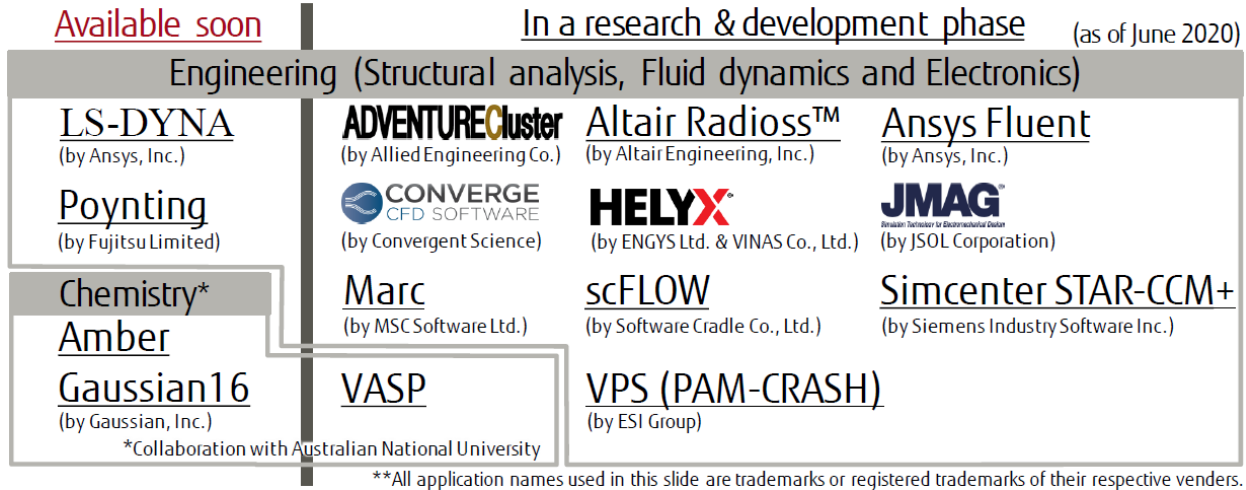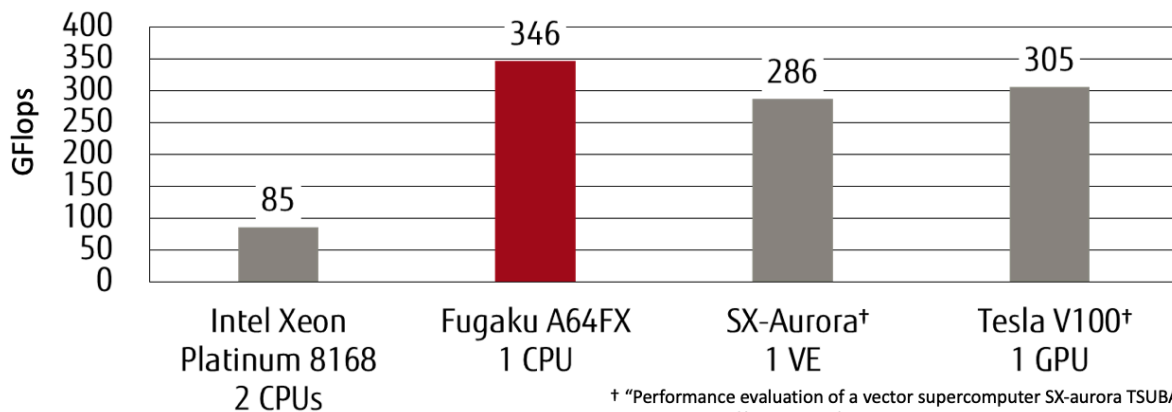


# Application Benchmarks

A number of applications benchmarks have been run and compared with the Fugaku A64FX processor.

■ Himeno Benchmark (Fortran90)

■ Stencil calculation to solve Poisson's equation by Jacobi method

■WRF: Weather Research and Forecasting model

   ■ Vectorizing loops including IF-constructs is key optimization

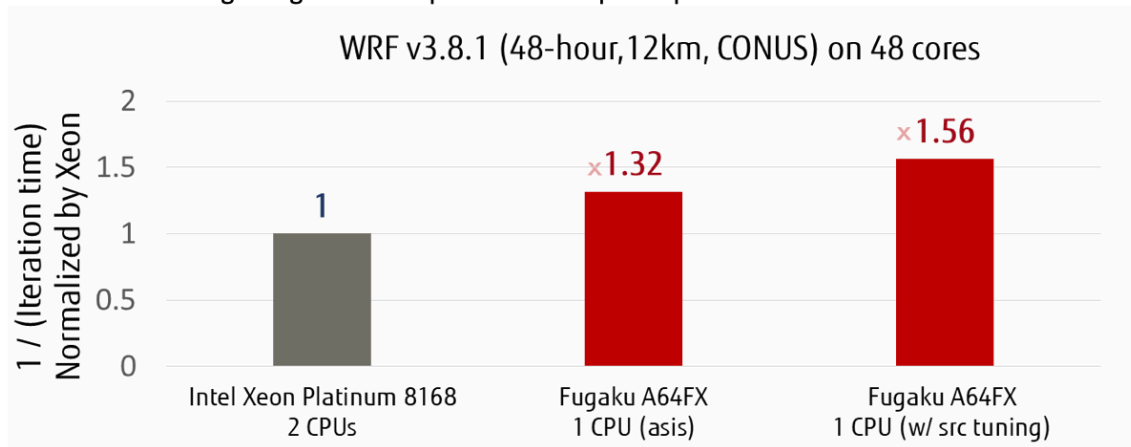   ■ Source code tuning using directives promotes compiler optimizations

Figure 10. Fugaku Benchmark Results

# Early Application and Ongoing Projects

RIKEN, in coordination with the Japanese Ministry of Education, Culture, Sports, Science and Technology (MEXT), has announced that the Fugaku supercomputer will be made available for research projects aimed to combat COVID-19. Below are some of the ongoing projects.

## Exploring New Drug Candidates for COVID-19 by "Fugaku"

Yasushi Okuno, RIKEN / Kyoto University

Currently, clinical trials are underway in Japan and overseas to confirm the effects of existing drugs on COVID-19. Some reports have shown that a given drug could have efficacy through these clinical trials, but the number of cases has been small, and no effective therapeutic drug has yet been identified. Furthermore, due to the small number of drugs being tested, it is possible that none of the drugs have a definite effect. Therefore, in this study, we perform molecular dynamics calculations using "Fugaku" to search and identify therapeutic drug candidates showing high affinity for the target proteins of COVID-19 from approximately 2,000 existing drugs that are not limited to existing antiviral drugs targeted in clinical trials.

## Prediction of Conformational Dynamics of Proteins on the Surface of SARS-Cov-2 using Fugaku

Yuji Sugita, RIKEN

On the surface of the coronavirus, there are many spike proteins that interact with virial receptor ACE2 on the host cell surface. To block the interaction between the spike protein and the receptor is an important research subject to develop a drug for COVID-19. Recently, atomic structures of the spike protein were determined using cryo-electron microscopy (cryo-EM). We perform atomistic molecular dynamics (MD) simulations of the spike protein in solution to

predict experimentally undetectable dynamic structures. We use GENESIS MD software, which allows us about 125 times faster MD simulations on Fugaku compared to the K computer. Furthermore, we enhance motions of a part of the spike protein using a multi-copy-simulation method to predict large-scale conformational dynamics of the spike proteins.

## Simulation Analysis of Pandemic Phenomena

Nobuyasu Ito, RIKEN

Social and economic impact is increasing globally, and Japan is now at critical bifurcation point. And challenges to make its visualization and "big data" mining have started. In this project, making the most of the "Fugaku" and other supercomputers, we make estimations of the possible future of our social and economic activities and policy options to control and resolve the situation. For this purpose, simulations of disease propagation and economic activities and SNS text mining are applied together with the National Institute of Advanced Industrial Science and Technology, Kyoto University, Tokyo Institute of Technology, the University of Hyogo, the University of Ryukyus, and the University of Tsukuba.

## Fragment Molecular Orbital Calculations for COVID-19 Proteins

Yuji Mochizuki (Rikkyo University) conducts the project in close collaboration with Shigenori Tanaka (Kobe University) and Kaori Fukuzawa (Hoshi Univeristy). By using our ABINIT-MP program, a series of fragment molecular orbital (FMO) calculations are carried out on COVID-19 proteins, and detailed interaction analyses are performed. Resulting data are made public as well.

ABINIT-MP has been used in the field of computational drug discovery for the last two decades, and a related consortium activity (FMODD) on the K computer was organized by Fukuzawa. On the present topic, we have performed FMO-based interaction analyses for a complex formed between a COVID-19 main protease and an inhibitor N3, where FX100 at Nagoya University was employed for computations. The analyzed results were published as a paper at ChemRxiv site, a month after the release of the original PDB structure (6LU7). Crucial residues in interacting with the inhibitor were identified by our analyses.

## Prediction and Countermeasure for Virus Droplet Infection under the Indoor Environment

Makoto Tsubokura, RIKEN / Kobe University

Virus droplet infection caused by sneezing, coughing, or talking is strongly influenced by the flow, temperature, and humidity of the air around an infected person and potential victims. Especially in the case of the new corona virus, possibility of aerosol infection by atomized droplets is suggested in addition to the usual droplet infection. Because smaller aerosol particles drift in the air for a longer time, it is imperative to predict the scattering route and to estimate how surrounding airflow affects the infection so that the risk of droplet infection can be properly assessed and effective measures to reduce infection can be proposed. In this project, massively parallel coupling simulation of virus droplet scattering, with airflow and heat transfer under the indoor environment such as inside a commuter train, offices, classrooms, and hospital rooms,

will be conducted. By taking into account the characteristics of the virus, its infection risk of virus droplets is assessed under various conditions. Then countermeasures to reduce the risk are proposed from a viewpoint of controlling the airflow. This project is a collaboration with RIKEN, Kyoto Institute of Technology, Kobe University, Osaka University, Toyohashi University of Technology, and Kajima Corporation.

## Fujitsu and HPE/Cray

HPE/Cray announced the Cray CS500, which is based on the Fujitsu A64FX processor. This product provides a fully enabled Cray programming environment on the system. There have been a number of early adopters, including SUNY Stony Brook, DOE's Los Alamos National Laboratory and ORNL, and University of Bristol. The SUNY Stony Brook system is a $5 million testbed project funded by the National Science Foundation and conducted in collaboration with RIKEN CCS in Japan.

Fujitsu has also announced two A64FX systems called the PRIMEHPC FX1000 and FX700. For customers inside Japan, FX1000 deployments start at a minimum of 48 nodes, and FX700 starts at a minimum of 2 nodes. For customers outside Japan, the entry point for FX1000 is 192 nodes, and the FX700 starts at 128 nodes.

## Summary

The R-CCS Fugaku system is very impressive with over 7 million cores and a peak performance of 514 PFLOP/s in 64-bit floating point for standard scientific computations and 2.15 EFLOP/s in 16-bit floating point (not bfloat16 floating point format) for machine-learning applications. The Fugaku system is almost three times (2.84×) faster than the system it replaces in the number one spot. The HPL benchmark result at 415 PFLOP/s, or 81% of theoretical peak performance, is also impressive with an efficiency of 14.7 GFLOP/s per watt. The HPCG performance at only 2.5% of peak performance shows the strength of the memory architecture performance. The ratio of bytes per floating-point operations from memory on the ARM A64FX HBM2 memory is 0.4 bytes per double-precision FLOP/s, which shows a good balance for floating point operations to data transfer from memory. One would expect good performance on computational science problems and machine-learning applications.

# Appendix A.

**Table A-1. Comparison with top machines on the TOP500**

|  | RIKEN Fugaku | ORNL Summit | Sunway TaihuLight | TianHe-2A |
|---|---|---|---|---|
| **Theoretical Peak** | 514 PFLOP/s | 200 PFLOP/s = (.54*2 CPU + 6*7 Accelerator) | 125.4 PFLOP/s = CPEs + MPEs<br><br>Cores per Node = 256 CPEs + 4 MPEs<br><br>Supernode = 256 Nodes<br><br>System = 160 Supernodes<br><br>Cores = 260 * 256 * 160 = 10.6M | 94.97 PFLOP/s = (7.52 CPU + 87.45 Accelerator) PFLOP/s |
| **HPL Benchmark FLOP/s** | 415 PFLOP/s | 149 PFLOP/s | 93 PFLOP/s | 13.987 PFLOP/s out of a theoretical peak of 21.86 PFLOP/s. |
| **HPL % peak** | 81% | 74% | 74.16% | 63.98% |
| **HPCG benchmark** | 13 PFLOP/s | 2.92 PFLOP/s | .371 PFLOP/s | 4096 nodes<br><br>.0798 PFLOP/s |
| **HPCG % peak** | 2.8% | 1.5% | 0.30% | 0.365% |
| **Compute nodes** | 152,064 (This is on 96% of the full system) | 4608 = 256 cabinets * 18 nodes/cabinet | 40,960 | 17,792 |
| **Node** | 48 cores | 2 IBM POWER9 CPUs 3.07 GHz plus 6 Nvidia V100 (.54 TFLOP/s each) Tesla GPUs / node (7 TFLOP/s each) | 256 CPEs + 4 MPEs | 2 – Intel Ivy Bridge (12 cores, 2.2 GHz) plus 2 Matrix-2000, 1.2 GHz) |
| **Node peak performance** | 3.4 TFLOP/s | 43 TFLOP/s =<br><br>(1.08 CPU + 42 GPU) TFLOP/s | 12 | 5.3376 TFLOP/s = $(2 \times 211.2$ CPU + $2 \times 2.4576$ Accelerator) TFLOP/s |

| Node memory | 32 GB HMB2 | 32 GB CPU + 6 GB GPU | 32 GB per node | 64 GB CPU + 128 GB Accelerator |
|---|---|---|---|---|
| **System memory** | 4.85 PB | 1.76 PB = 4608*600 GB of coherent memory (6×16 = 96 GB HBM2 plus 2×8×32 = 512 GB DDR4 SDRAM) | 1.31 PB (32 GB × 40,960 nodes) | 3.4 PB = 17,792 × (64 GB + 128 GB) |
| **Configuration** | 158,976 nodes | 256 Racks × 18 Nodes | Node peak performance is 3.06 TFLOP/s, or 11.7 GFLOP/s per core.<br><br>260 cores/node<br><br>CPE: 8 FLOPs/core/cycle (1.45 GHz × 8 × 256 = 2.969 TFLOP/s)<br><br>MPE (2 pipelines) 2 × 4 × 8 FLOPs/core/cycle (1.45 GHz × 1= 0.0928TFLOP/s)<br><br>Node peak performance: 3.06 TFLOP/s<br><br>1 thread/core<br><br>Nodes connected using PCI-E<br><br>The topology is Sunway network.<br><br>256 nodes = a supernode (256 × 3.06 TFLOP/s = . 783 PFLOP/s)<br><br>160 supernodes make up the whole system (125.4PFLOP/s)<br><br>The network system consists of three different levels, with | 2 Nodes per blade, 16 blades per frame and 4 frames per cabinet and 139 cabinets in the system. |

| | | | | |
|---|---|---|---|---|
| | | | the central switching network at the top, the super node network in the middle, and the resource-sharing network at the bottom.<br><br>4 SNs = cabinet<br><br>Each cabinet ~3.164 PFLOP/s<br><br>256 nodes per SN<br><br>1,024 nodes (3 TFLOP/s each) per cabinet<br><br>40 cabinets ~125 PFLOP/s | |
| **Total system** | 7,630,848 = 158,976 * 48 cores | 2,397,824 cores | 10,649,600 cores = Node (260) × supernodes(256 nodes) × 160 supernodes | 4,981,760 cores = (17,792 × 2 Ivy Bridge with 12 cores) + (2 × Matrix-2000 × 128) |
| **Power (processors, memory, interconnect)** | 28.33 MW<br><br>(7.33*3.863) | 11 MW | 15.3 MW | 16.9 MW for full system |
| **Footprint** | 1,920 m$^2$ | 520 m$^2$ | 605 m$^2$ | 400 m$^2$ (50 m$^2$/line*8) or 720 m$^2$ total room |