The post-K for general-purpose, energy-efficient and sustained application performance

Update of Post-K project

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
FLAGSHIP2020 Project

Missions
- Building the Japanese national flagship supercomputer, post K, and
- Developing wide range of HPC applications, running on post K, in order to solve social and science issues in Japan

Overview of post-K architecture
Node: Manycore architecture
- Armv8-A + SVE (Scalable Vector Extension)
- SIMD Length: 512 bits
- # of Cores: 48 + (2/4 for OS) (> 2.7 TF / 48 core)
- Co-design with application developers and high memory bandwidth utilizing on-package stacked memory (HBM2) 1 TB/s B/W
- Low power : 15GF/W (dgemm)

Network: TofuD
- Chip-Integrated NIC, 6D mesh/torus Interconnect

Status and Update
- Close to end in “Design and Implementation”.
- The prototype CPU powered-on and development is as scheduled
- RIKEN announced the Post-K early access program to begin around Q2/CY2020
- We are working on performance evaluation and tuning by simulators and compilers
Target science: 9 Priority Issues

① Innovative Drug Discovery
RIKEN Quant. Biology Center
Society with health and longevity

② Personalized and Preventive Medicine
Inst. Medical Science, U. Tokyo

③ Hazard and Disaster induced by Earthquake and Tsunami
Earthquake Res. Inst., U. Tokyo
Disaster prevention and global climate

④ Environmental Predictions with Observational Big Data
Center for Earth Info., JAMSTEC

⑤ High-Efficiency Energy Creation, Conversion/Storage and Use
Inst. Molecular Science, NINS

⑥ Innovative Clean Energy Systems
Grad. Sch. Engineering, U. Tokyo
Energy issues

⑦ New Functional Devices and High-Performance
Inst. For Solid State Phys., U. Tokyo
Industrial competitiveness

⑧ Innovative Design and Production Processes for the Manufacturing Industry in the Near Future

⑨ Fundamental Laws and Evolution of the Universe
Cent. for Comp. Science, U. Tsukuba
Basic science
Interactive Models of Socio-Economic Phenomena and their Applications

Frontiers of Basic Science - challenge to extremes -

Formation of exo-planets (second Earth) and Environmental Changes of Solar Planets

Mechanisms of Neural Circuits for Human Thoughts and Artificial Intelligence

Projects (more than 10 teams) were selected in Jun 2016
Co-design from Apps to Architecture

- **Architectural Parameters to be determined**
  - #SIMD, SIMD length, #core, #NUMA node, O3 resources, specialized hardware
  - cache (size and bandwidth), memory technologies
  - Chip die-size, power consumption
  - Interconnect

- **We have selected a set of target applications**

- **Performance estimation tool**
  - Enables performance projection using Fujitsu FX100 execution profile to a set of arch. parameters.

- **Co-design Methodology (at early design phase)**
  1. Setting a set of system parameters
  2. Tuning target applications under the system parameters
  3. Evaluating execution time using prediction tools
  4. Identifying hardware bottlenecks and changing the set of system parameters

---

**Target Application**

<table>
<thead>
<tr>
<th>Program</th>
<th>Brief description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. GENESIS</td>
<td>MD for proteins</td>
</tr>
<tr>
<td>2. Genomon</td>
<td>Genome processing (Genome alignment)</td>
</tr>
<tr>
<td>3. GAMERA</td>
<td>Earthquake simulator (FEM in unstructured &amp; structured grid)</td>
</tr>
<tr>
<td>4. NICAM+LETK</td>
<td>Weather prediction system using Big data (structured grid stencil &amp; ensemble Kalman filter)</td>
</tr>
<tr>
<td>5. NTChem</td>
<td>Molecular electronic (structure calculation)</td>
</tr>
<tr>
<td>6. FFB</td>
<td>Large Eddy Simulation (unstructured grid)</td>
</tr>
<tr>
<td>7. RSDFT</td>
<td>An ab-initio program (density functional theory)</td>
</tr>
<tr>
<td>8. Adventure</td>
<td>Computational Mechanics System for Large Scale Analysis and Design (unstructured grid)</td>
</tr>
<tr>
<td>9. CCS-QCD</td>
<td>Lattice QCD simulation (structured grid Monte Carlo)</td>
</tr>
</tbody>
</table>
KPIs on post-K development in FLAGSHIP 2020 project

3 KPIs (key performance indicator) were defined for post-K development

- **1. Extreme Power-Efficient System**
  - 30-40 MW at system level

- **2. Effective performance of target applications**
  - It is expected to exceed 100 times higher than the K computer’s performance in some applications

- **3. Easy-of-use system for wide-range of users**
**CPU Architecture: A64FX**

- **Armv8.2-A (AArch64 only) + SVE (Scalable Vector Extension)**
  - FP64/FP32/FP16
    - [https://developer.arm.com/products/architecture/a-profile/docs](https://developer.arm.com/products/architecture/a-profile/docs)
- **SVE 512-bit wide SIMD**
- **# of Cores: 48 + (2/4 for OS)**
- Co-design with application developers and high memory bandwidth utilizing on-package stacked memory: **HBM2(32GiB)**
- Leading-edge Si-technology (7nm FinFET), low power logic design (**approx. 15 GF/W (dgemm)**), and power-controlling knobs
- **PCle Gen3 16 lanes**
- **Peak performance**
  - > 2.7 TFLOPS (>90% @ dgemm)
  - Memory B/W 1024GB/s (>80% stream)
  - Byte per Flops: approx. 0.4

◆ “Common” programing model will be to run each MPI process on a NUMA node (CMG) with OpenMP-MPI hybrid programming.
◆ 48 threads OpenMP is also supported.
ARM v8 Scalable Vector Extension (SVE)

- SVE is a complementary extension that does not replace NEON, and was developed specifically for vectorization of HPC scientific workloads.

- The new features and the benefits of SVE comparing to NEON
  - **Scalable vector length (VL)**: Increased parallelism while allowing implementation choice of VL
  - **VL agnostic (VLA) programming**: Supports a programming paradigm of write-once, run-anywhere scalable vector code
  - **Gather-load & Scatter-store**: Enables vectorization of complex data structures with non-linear access patterns
  - **Per-lane predication**: Enables vectorization of complex, nested control code containing side effects and avoidance of loop heads and tails (particularly for VLA)
  - Predicate-driven loop control and management: Reduces vectorization overhead relative to scalar code
  - Vector partitioning and SW managed speculation: Permits vectorization of uncounted loops with data-dependent exits
  - Extended integer and floating-point horizontal reductions: Allows vectorization of more types of reducible loop-carried dependencies
  - Scalarized intra-vector sub-loops: Supports vectorization of loops containing complex loop-carried dependencies
SVE example

**DAXPY (scalar)**
```plaintext
// subroutine daxpy(x,y,a,n)
// real*8 x(n),y(n),a
// do i = 1,n
//     y(i) = a*x(i) + y(i)
// enddo
// x0 = &x[0], x1 = &y[0], x2 = &a, x3 = &n
daxpy_
  ldrsw x3, [x3] // x3=*n
  mov x4, #0 // x4=i=0
  ldr d0, [x2] // d0=*a
  b .latch
.loop:
  ldr d1, [x0,x4,lsl 3] // d1=x[i]
  ldr d2, [x1,x4,lsl 3] // d2=y[i]
  fmaadd d2, d1, d0, d2 // d2+=x[i]*a
  str d2, [x1,x4,lsl 3] // y[i]=d2
  add x4, x4, #1 // i+=1
.latch:
  cmp x4, x3 // i < n
  b.lt .loop // more to do?
ret
```

**DAXPY (SVE)**
```plaintext
// subroutine daxpy(x,y,a,n)
// real*8 x(n),y(n),a
// do i = 1,n
//     y(i) = a*x(i) + y(i)
// enddo
// x0 = &x[0], x1 = &y[0], x2 = &a, x3 = &n
daxpy_
  ldrsw x3, [x3] // x3=*n
  mov x4, #0 // x4=i=0
  whilelt p0.d, x4, x3 // p0=while(i<<n)
  ldlrd z0.d, p0/z, [x2] // p0:z0=bcast(*a)
  .loop:
    lld d1, p0/z, [x0,x4,lsl 3] // p0:z1=x[i]
    lld d2, p0/z, [x1,x4,lsl 3] // p0:z2=y[i]
    fmla z2.d, p0/m, z1.d, z0.d // p0:?z2+=x[i]*a
    stdl z2.d, p0, [x1,x4,lsl 3] // p0:y[i]=z2
    incd x4 // i+=($VL/64)
  .latch:
    whilelt p0.d, x4, x3 // p0=while(i<<n)
    b.first .loop // more to do?
ret
```

- Compact code for SVE as scalar loop
- OpenMP SIMD directive is expected to help the SVE programming

20018/02/27
CMG (Core Memory Group)

- CMG: 13 cores (12+1) and L2 cache (8MiB 16way) and memory controller for HBM2 (8GiB)
- X-bar connection in a CMG maximize efficiency for throughput of L2 (>115 GB/s for R, >57 GB/s for W)
- Assistant core is dedicated to run OS demon, I/O, etc
- 4 CMGs support cache coherency by ccNUMA with on-chip directory ( > 115GB/s x 2 for inter-CMGs)

Figures from the slide presented in Hotchips 30 by Fujitsu
Tofu interconnect D

- Direct network, 6-D Mesh/Torus
- 28Gbps x 2 lanes x 10 ports (6.8GB/s / link)
- Network Interface on Chip
  - 6 TNIs: Increased TNIs (Tofu Network Interface) achieves higher injection BW & flexible comm. Patterns
  - Memory bypassing achieves low latency

<table>
<thead>
<tr>
<th></th>
<th>TofuD spec</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data rate</td>
<td>28.05 Gbps</td>
</tr>
<tr>
<td>Link bandwidth</td>
<td>6.8 GB/s</td>
</tr>
<tr>
<td>Injection bandwidth</td>
<td>40.8 GB/s</td>
</tr>
</tbody>
</table>

Ref) K computer: Link BW=5.0GB/s, #TNI=4

<table>
<thead>
<tr>
<th></th>
<th>Measured</th>
</tr>
</thead>
<tbody>
<tr>
<td>Put throughput</td>
<td>6.35 GB/s</td>
</tr>
<tr>
<td>PingPong latency</td>
<td>0.49~0.54 μs</td>
</tr>
</tbody>
</table>
Preliminary Performance by “real silicon”

- The prototype CPU has been powered-on and preliminary performance evaluation by the prototype CPU has been done.
- Improvement by micro architectural enhancements, 512-bit wide SIMD, HBM2 and process technology
- The results are based on the Fujitsu compiler optimized for our microarchitecture and SVE
- AI apps will be supported by SVE FP16 instructions.

Figures from the slide presented in Hotchips 30 by Fujitsu
Low-power Design & Power Management

● Leading-edge Si-technology (7nm FinFET)
● Low power logic design (15 GF/W @ dgemm)

● A64FX provides power management function called “Power Knob”
  ● FL pipeline usage: FLA only, EX pipeline usage : EXA only, Frequency reduction …
  ● User program can change “Power Knob” for power optimization
  ● “Energy monitor” facility enables chip-level power monitoring and detailed power analysis of applications

● “Eco-mode” : FLA only with lower “stand-by” power for ALUs
  ● Reduce the power-consumption for memory intensive apps.

● Retention mode: power state for de-activation of CPU with keeping network alive
  ● Large reduction of system power-consumption at idle time
KPIs on post-K development in FLAGSHIP 2020 project

3 KPIs (key performance indicator) were defined for post-K development

- **1. Extreme Power-Efficient System**
  - Approx. 15 GF/W (dgemm) confirmed by the prototype CPU
  - Power consumption of 30 - 40MW (for system) is expected to be achieved

- **2. Effective performance of target applications**
  - It is expected to exceed 100 times higher than the K computer’s performance in some applications
  - 106 times faster in GENESIS (MD application), 153 times faster in NICAM+LETKF (climate simulation and data assimilation) were estimated

- **3. Easy-of-use system for wide-range of users**
  - Shared memory system with high-bandwidth on-package memory must make existing OpenMP-MPI program ported easily.
  - No programming effort for accelerators such as GPUs is required.
  - Co-design with application developers
“Fujitsu Completes Post-K Supercomputer CPU Prototype, Begins Functionality Trials”, HPCwire June 21, 2018

“Fujitsu has now completed the prototype CPU chip that will serve as the core of post-K, commencing functionality field trials.”

Shelf: 48 CPUs (24 CMU)
Rack: 8 shelves = 384 CPUs (8x48)
# Advances from K computer

<table>
<thead>
<tr>
<th></th>
<th>K computer</th>
<th>Post-K</th>
<th>ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td># core</td>
<td>8</td>
<td>48</td>
<td></td>
</tr>
<tr>
<td>Si tech. (nm)</td>
<td>45</td>
<td>7</td>
<td></td>
</tr>
<tr>
<td>Core perf. (GFLOPS)</td>
<td>16</td>
<td>56~</td>
<td>3.5</td>
</tr>
<tr>
<td>Chip(node) perf. (TFLOPS)</td>
<td>0.128</td>
<td>2.7~</td>
<td>21</td>
</tr>
<tr>
<td>Memory BW (GB/s)</td>
<td>64</td>
<td>1024</td>
<td></td>
</tr>
<tr>
<td>B/F (Bytes/FLOP)</td>
<td>0.5</td>
<td>0.4</td>
<td></td>
</tr>
<tr>
<td>#node / rack</td>
<td>96</td>
<td>384</td>
<td>4</td>
</tr>
<tr>
<td>Rack perf. (TFLOPS)</td>
<td>12.3</td>
<td>1036.8</td>
<td>84</td>
</tr>
<tr>
<td>#node/system</td>
<td>82,944</td>
<td>???</td>
<td></td>
</tr>
<tr>
<td>System perf.(PFLOPS)</td>
<td>10.6</td>
<td>???</td>
<td></td>
</tr>
</tbody>
</table>

- SVE increases core performance
- Silicon tech. and scalable architecture (CMG) to increase node performance
- HBM enables high bandwidth
RIKEN is constructing “PostK” performance evaluation environment for application programmers to evaluate and estimate the performance of their applications on “PostK” and for performance tuning for “postK”.

The “PostK” performance evaluation environment is available on the servers installed in RIKEN. The environment includes the following tools and servers:

- A small-scale FX100 system and “postK” performance estimation tool:
  The estimation tool gives the performance estimation of multithreaded programs on “postK” from the profile data taken on FX100.

- “PostK” processor simulator based on GEM-5:
  “PostK” processor simulator will give a detail performance results including estimated executing time, cache-miss, the number of instruction executed in O3. The user can understand how the compiled code for SVE is executed on “postK” processor for optimization. (Arm released GEM-5 beta0 of SVE)
  FP16 SVE will be available soon.

- Compilers for “PostK” processor
  - Arm Compiler : LLVM-based compiler to generate code forArmv8-A + SV. C,C++ by Clang, Fortran by Flang

- SVE emulator on Arm server, developed by Arm for fast SVE code execution.

- Arm Severs (HPE Appollo 70, Available from Dec/2018)
# Schedule on Development and Porting Support

<table>
<thead>
<tr>
<th>CY2017</th>
<th>CY2018</th>
<th>CY2019</th>
<th>CY2020</th>
<th>CY2021</th>
</tr>
</thead>
<tbody>
<tr>
<td>Design and Implementation</td>
<td>Manufacturing</td>
<td>Installation, and Tuning</td>
<td>Operation</td>
<td></td>
</tr>
</tbody>
</table>

### Specification
- Armv8-A + SVE
- Overview
- Detailed hardware info.

### Optimization Guidebook
- Publishing Incrementally

### RIKEN Performance Evaluation Environment
- Performance estimation tool using FX100
- RIKEN Simulator

### Early Access Program

- CY2018. Q2, Optimization guidebook is incrementally published
- CY2020. Q2, Early access program starts
- CY2021. Q1/Q2, General operation starts

**Note:** Fujitsu will reveal features of Post-K CPU at Hot Chips 2018.
Post-K CPU New Innovations: Summary

1. **Ultra high bandwidth using on-package memory & matching CPU core**
   - Recent studies show that majority of apps are memory bound, some compute bound but can use lower precision e.g. FP16
   - Comparison w/mainstream CPU: much faster FPU, almost order magnitude faster memory BW, and ultra high performance accordingly
   - Memory controller to sustain massive on package memory (OPM) BW: difficult for coherent memory CPU, first CPU in the world to support OPM

2. **Very Green e.g. extreme power efficiency**
   - Power optimized design, clock gating & power knob, efficient cooling
   - Power efficiency much better than CPUs, comparable to GPU systems

3. **Arm Global Ecosystem & SVE contribution**
   - Annual processor production: x86 3-400mil, ARM 21bil, (2~3 bil high end)
   - Rapid upbringing HPC&IDC Ecosystem (e.g. Cavium, HPE, Sandia, Bristol,…)
   - SVE(Scalable Vector Extension) -> Arm-Fujitsu co-design, future global std.

4. **High Performance on Society5.0 apps including AI**
   - Next gen AI/ML requires massive speedup => high perf chips + HPC massive scalability across chips
   - Post-K processor: support for AI/ML acceleration e.g. Int8/FP16+fast memory for GPU-class convolution, fast interconnect for massive scaling
   - Top performance in AI as well as other Society 5.0 apps