

Performance and Power Consumption Analysis of ARM Scalable Vector Extension

Tetsuya Odajima, Yuetsu Kodama, Mitsuhsa Sato
RIKEN Center for Computational Science, Japan

FLAGSHIP2020 Project

- Missions

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

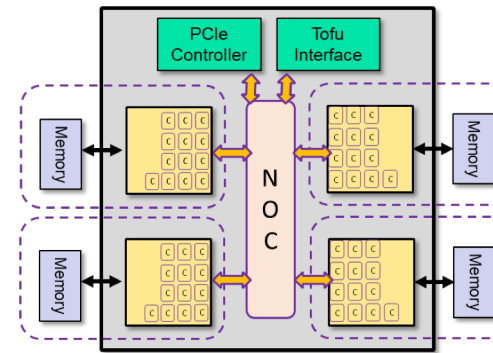
- Overview of post-K architecture

Node: Manycore architecture

- Armv8.2-A + SVE (Scalable Vector Extension)
- SIMD Width: 512 bit
- # of Cores: 48+2 or 4 (for OS)
- Co-design with application developers and high memory bandwidth utilizing **HBM2 (on-package stacked memory)**

Network: TofuD

- Chip-Integrated NIC, 6D mesh/torus Interconnect



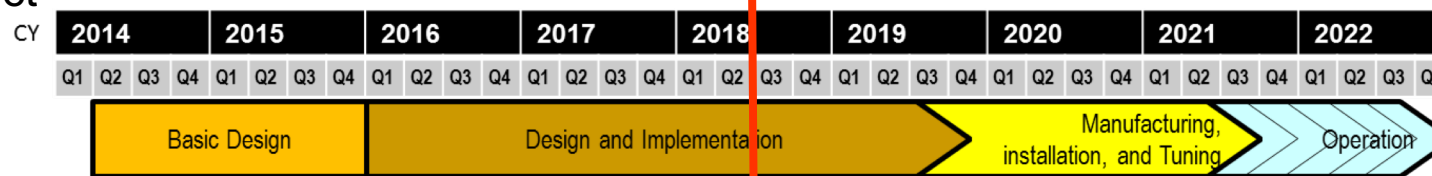
Post-K processor



Prototype board

- Status and Update

- Now 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



Background of Our Research

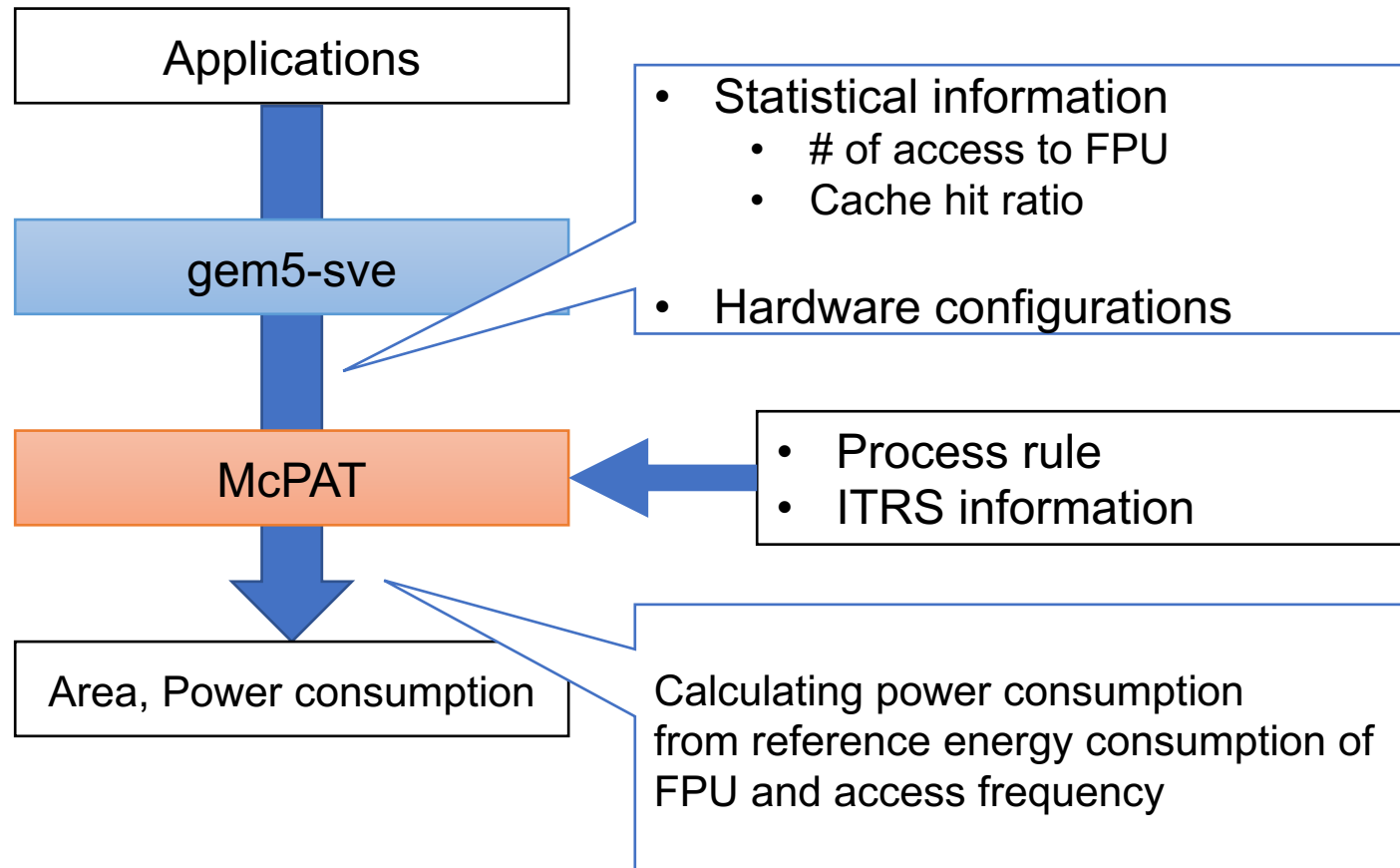
- Performance evaluation and tuning for processor of Post-K by simulator



- Working on research of processor architecture by using simulator
 - Simulator can change its various hardware parameters easier
 - Alternative parameters, not that of Post-K
- Our simulator environment
 - **gem5**: general-purpose processor simulator, supports SVE
 - gem5 got from ARM had only atomic mode, so we implemented O3 mode of gem5-sve
 - **McPAT**: framework for calculating power consumption
 - Estimate the power consumption from the #of access to FPU, registers, caches, etc.

Our Simulator Environment

- Flow of estimating area and power consumption



ARM Scalable Vector Extension (SVE)

- Support 128-bit ~ 2048-bit vector length (VL)
 - 128-bit increments (128 x LEN bit)
- Vector Length Agnostic (VLA)
 - Programming is independent of VL
 - There is no VL information in SVE codes
 - They refer to “LEN” system register in each processor to decide VL
 - (LEN can be changed by kernel call)
 - Same binary runs on different VL environments



Purpose of Our Research

- We have been evaluating the effect of vector length on the performance of benchmarks
 - VLA of SVE is useful for our evaluation
 - It enables to evaluate different vector length with same binary
 - In order to improve the performance of wide SIMD, enough Out-of-Order resources are needed



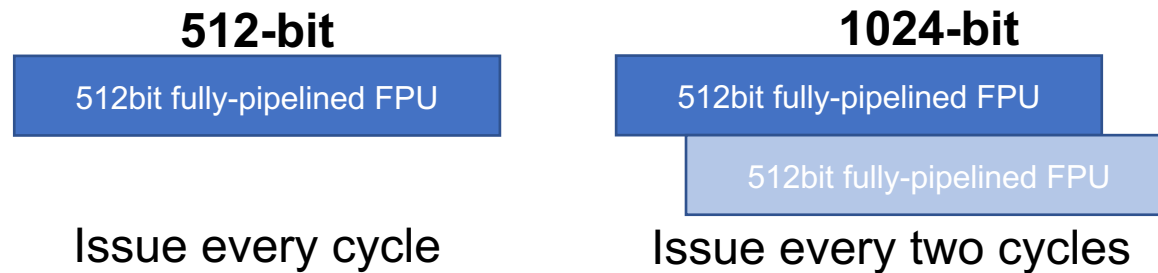
- In our research, we evaluate the effect of vector length in the performance and the energy consumption by using simulator
 - What is a trade-off between performance and hardware resource?

Purpose of Our Research | Detailed

- Comparing the performance and energy consumption among 512-bit and 1024-bit vector length
 - Peak performance of 1024-bit FPU is twice that of 512-bit
 - Area of FPU and register file will be increasing simultaneously
 - Various implementations of FPU
 - For 1024-bit vector
 - One issue with 1024-bit FPU
 - Two issues with 512-bit FPU
 - Area of 512-bit FPU is almost half that of 1024-bit
 - Trade-off between performance and hardware resources
-  Peak performance of FPU is half
- 
- We evaluate the advantage of a longer vector length with multi-cycle FPU

Simulator Environment | Detailed

- gem5 and McPAT are highly flexible tools, but default functions are limited
- **gem5-sve**
 - gem5-sve got from ARM had only “atomic mode”
 - We have been implementing “O3 mode” for gem5-sve
 - Throughput control
 - To keep hardware resource size among different vector length
 - (e.g.) 512-bit: issue every cycle == 1024-bit: issue every two cycles



Peak throughput of FPU and Hardware resources are the same

Simulator Environment | Detailed (Cont'd)

- **McPAT**

- Do not correspond to SIMD instructions (only scalar)
 - We defined information of SVE instructions to the McPAT
- Calculating energy consumption of SIMD instruction
 - { # of access to FPUs } x { # of vector elements } x { reference energy }

Evaluation Environment

- Based on preset parameter in gem5
- Defined instruction latency for SVE referred to NEON

Hardware parameters

Clock Frequency	2.0 GHz	# of cores	1
L1 Dcache, Icache size	32 kB	L2 cache size	2 MB
Integer pipeline	2	Load/Store unit	1 / 1
Floating pipeline	2	Fetch width	3
Process rule	45 nm		

Out-of-Order resource parameters

IQ (Reservation Station)	64 (←32)
ROB (Re-order Buffer)	64 (←48)
LQ (Load Queue)	16
SQ (Store Queue)	16
Physical Vector Register	96

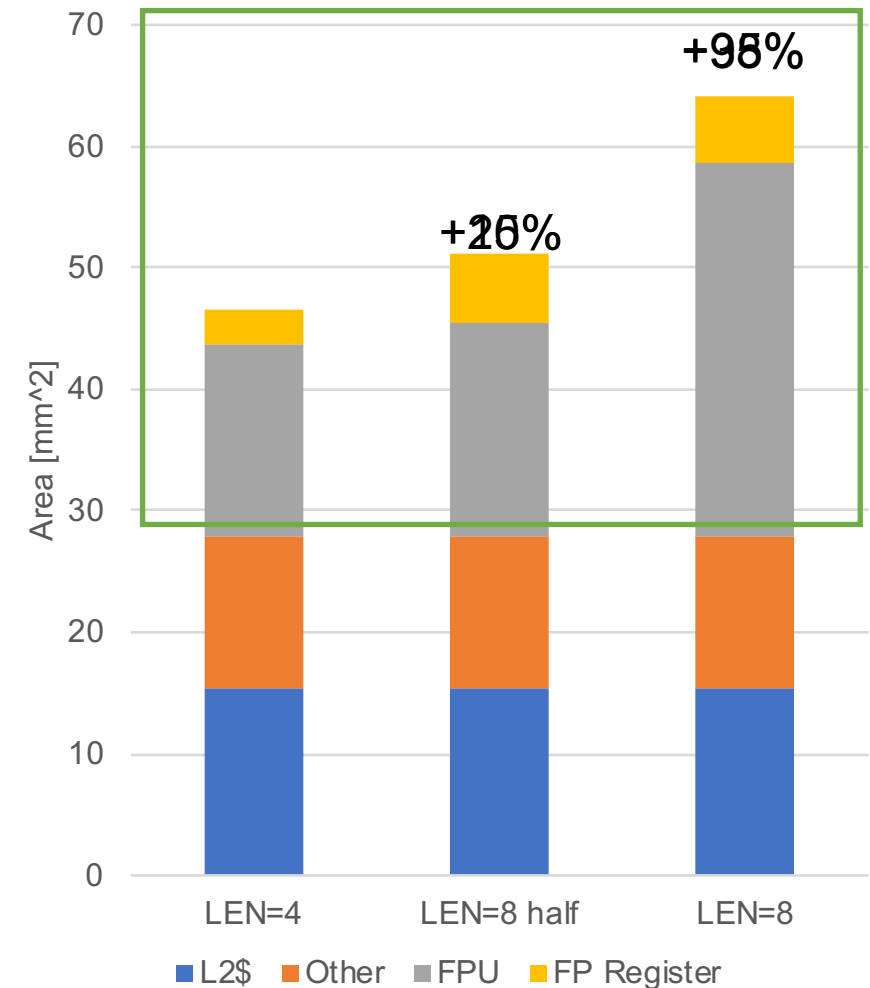
Evaluation Environment (Cont'd)

- In gem5, SIMD width of execution unit and register file will be changed by the VL simultaneously
 - Throughput control; to evaluate different VL with almost the same hardware resources

	LEN=4	LEN=8 half	LEN=8
Vector Length	512-bit	1024-bit	→
FPU throughput	512-bit / cycle	1024-bit / 2cycles	1024-bit / cycle
Peak FPU performance ratio	1	1	2
Number of registers	512-bit x 96	1024-bit x 96	→
L1 throughput	512-bit / cycle	1024-bit / 2cycles	1024-bit / cycle
L2 throughput	256bit / cycle	→	→
FPU & Register area ratio	1	1.25	1.95

Processor Area

- Comparing to LEN=4,
 - area of LEN=8 half is only +10%
 - area of LEN=8 is +38%
 - Only FPUs+registers part: +95%
(FPU = NEON + SVE)
- The impact to multi-core is great



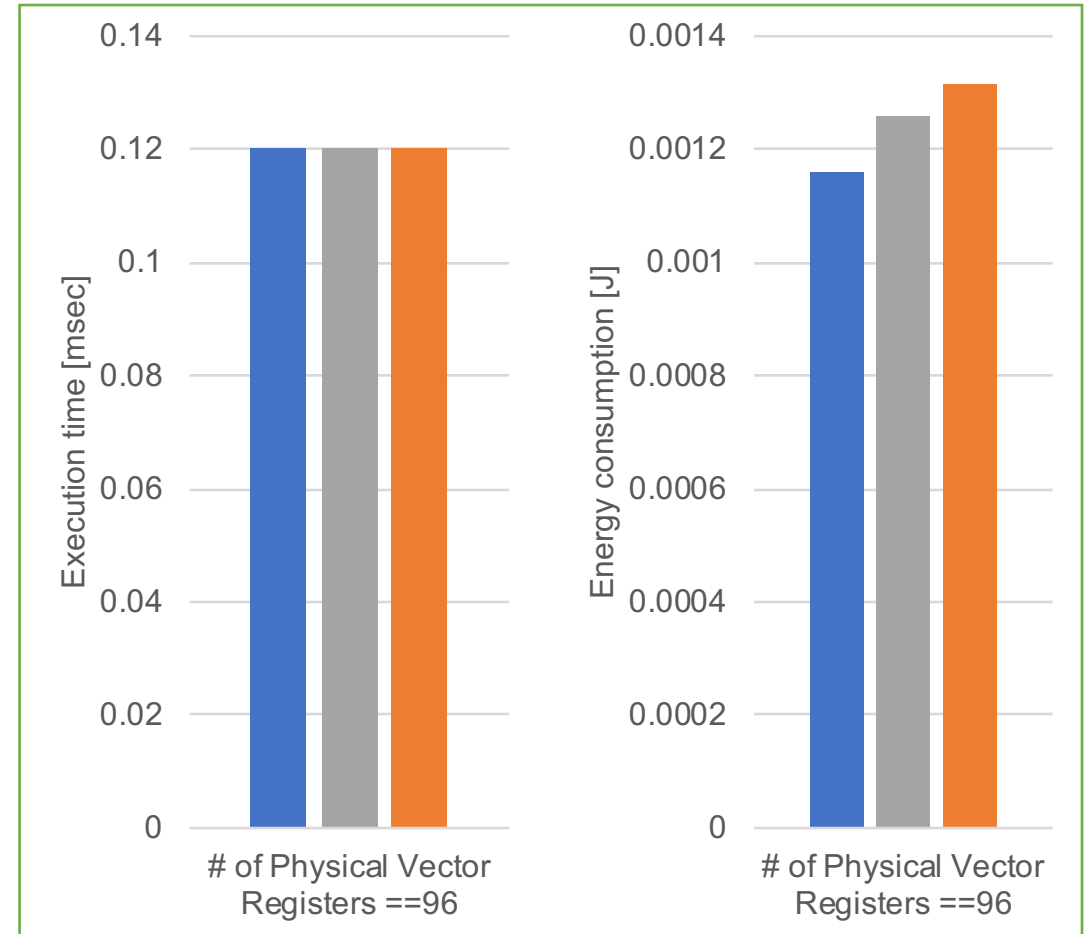
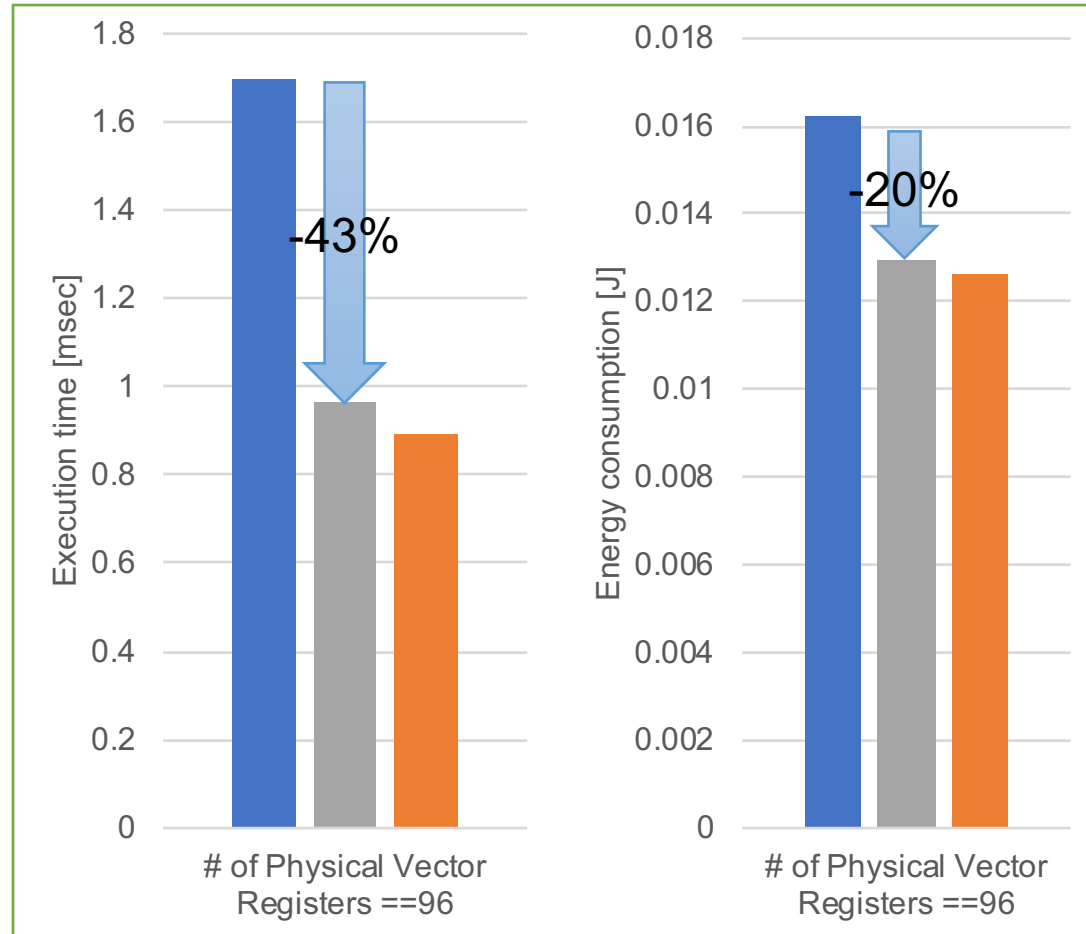
Evaluation Environment (Cont'd)

- Compiler
 - ARM clang version 18.2, -Ofast -march=armv8-a+sve
- Evaluated kernels
 - N-body: latency-bound
 - N=512, TIME_STEP=1
 - Stream Triad: cache/memory-bound
 - N=25600, TIME_STEP=10
- In this evaluation, we don't consider leak current

Evaluation | Time & Energy Consumption

N-body

Stream Triad



LEN=4 LEN=8 half LEN=8

Why is the performance improved with "LEN = 8 half"?

- Maximum utilization of FPU on **LEN=4**
 - N-body: 43.9%
- In this kernels, the utilization with **LEN=4** has room for improvement even if **LEN=8** uses twice cycles on FPU
- However, 44% is the ideal value
 - Utilization of FPU may be down due to lack of O3 resources in fact
- If the utilization of FPU is close to 100%? (e.g.) tuned dgemm
 - Performance improvement can not be obtained, because it limits the throughput of FPU (512-bit)

Conclusion

- By increasing the vector length, performance improvement and low power consumption can be realized in latency-bound kernel
 - N-body: "LEN=8 half" achieves **43% speedup** than LEN=4
20% low energy
- Vector length dose not affect in the performance of memory-bound kernel
- The energy consumption of LEN=8 half is almost the same as that of LEN=8



- A longer vector length with multi-cycle vector units (LEN=8 half) is well balanced between the performance and hardware resource

-
- Backup

Calculating Formula of FPU Area

- Predefined : $8.47 / 2 / 10^6 \times \alpha$



- Our Formula: $\{8.47 + \{8.47 \times 0.8 \times (VL - 2)\} / 2\} / 10^6 \times \alpha$

McPAT
defined
FP/NEON unit

SVE unit part

ITRS info.
defined in McPAT