FUJITSU

shaping tomorrow with you

Post-K: Building the Arm HPC Ecosystem

Toshiyuki Shimizu
FUJITSU LIMITED

Nov. 13th, 2017

Copyright 2017 FUJITSU LIMITED

Post-K: Building up Arm HPC Ecosystem

FUJITSU

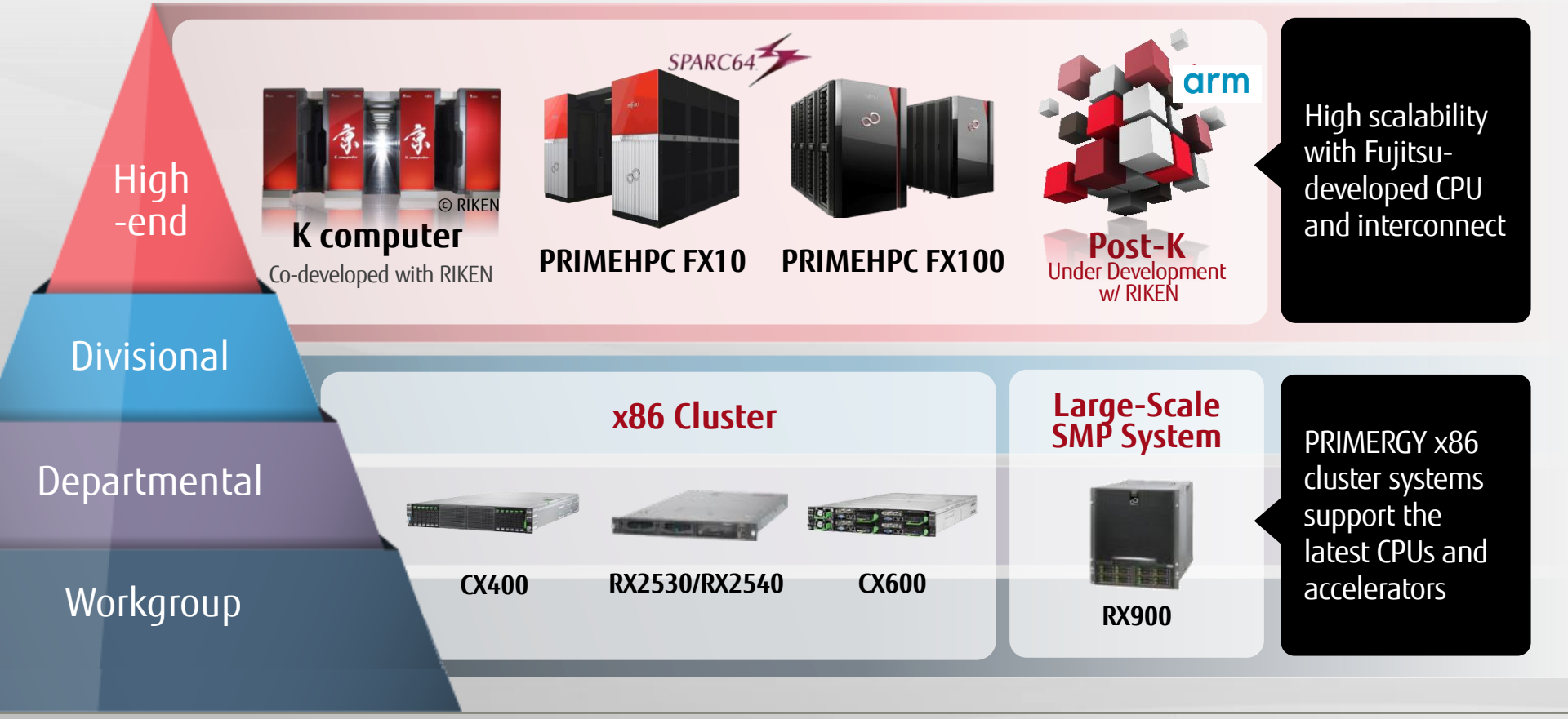
- Fujitsu's approach for HPC
- Approach to make Post-K a resounding success
- The high performance compiler increases software portability
- Summary

Copyright 2017 FUJITSU LIMITED

Fujitsu HPC Solutions to Meet Customer Demands

FUJITSU

- Supercomputers, both Fujitsu-developed CPUs and x86
- Single system image operation w/ Fujitsu system software
- High performance, high availability, and high reliability



Copyright 2017 FUJITSU LIMITED

Fujitsu High-end Supercomputers Development

FUJITSU

2011 2012 2013 2014 2015 2016 2017 2018 2019 2020

K computer

PRIMEHPC FX10 PRIMEHPC FX100

- 1.8x CPU perf. of K
- Easier installation
- 4x(DP) / 8x(SP) CPU per. of K, Tofu2
- High-density pkg & lower energy

Technical Computing Suite (TCS)

- Handles millions of parallel jobs
- FEFS: super scalable file system
- MPI: Ultra scalable collective communication libraries
- OS: Lower OS jitter w/ assistant core

Japan's National Projects

arm

Development

Operation of K computer

HPCI strategic apps program

App. review

FS projects

Post-K computer development

Post-K

K computer and PRIMEHPC FX10/FX100 in operation

The CPU and interconnect of FX10/FX100 inherit the K computer architectural concept, featuring state-of-the-art technologies

System software "TCS" supports Fujitsu supercomputer with originally introduced technologies

Many applications are currently running and being developed for science and various industries

Post-K supercomputer

RIKEN and Fujitsu are working together to provide a successor to K computer with application R&D teams using co-design approach

Copyright 2017 FUJITSU LIMITED

Post-K Features and Status

FUJITSU

- Fujitsu CPU core (w/ Arm SVE) and Tofu maintain the programming models and provide high application performance
- RIKEN & Fujitsu system software enable high performance and low power consumption with flexible operations
- Apps from 9 “priority issues” & many “exploratory challenges” are being optimized for the Post-K




	Functions & architecture	Post-K	FX100	FX10	K
CPU Core	Instruction set architecture	Armv8-A	SPARC V9		
	SIMD width	512bit	256bit	128bit	128bit
	Double precision (64bit)	✔	✔	✔	✔
	Single precision (32bit)	✔	✔	✔	✔
	Half precision (16bit)	✔	-	-	-
Interconnect	Tofu interconnect	Enhanced	Tofu2	Tofu	Tofu

Copyright 2017 FUJITSU LIMITED

Post-K Software Stack

FUJITSU

- Valuable feedbacks through "co-design" from application R&D teams

Post-K Applications

FUJITSU Technical Computing Suite / RIKEN Advanced System Software

Management Software	Hierarchical File I/O Software	Programming Environment
System management for highly available & power saving operation	Application-oriented file I/O middleware	XcalableMP
Job management for higher system utilization & power efficiency	Lustre-based distributed file system FEFS	MPI (Open MPI, MPICH)
		OpenMP, COARRAY, Math Libs
		Compilers (C, C++, Fortran)
		Debugging and tuning tools

Post-K
Under Development
w/ RIKEN

Linux OS / McKernel (Lightweight Kernel)

Post-K System Hardware

Copyright 2017 FUJITSU LIMITED

Post-K to be More Useful?

FUJITSU

- More apps from OSS & ISVs
  - High performance on real applications
  - Lower TCO
    - Low power consumption
    - Water cooling
  - De-facto standards
    - Lowering barriers in developing and porting
  - Ecosystem
    - More Arm platforms
    - More partners
    - More knowledge/experience inside/outside of communities

Copyright 2017 FUJITSU LIMITED

Making the Post-K a Resounding Success

FUJITSU

- Recapping the goal & requirements
  - High performance HW and SW complying open standards
  - Apps in quality & variety
  - Environments – rich, modern, and comprehensive
- Our approach
  - Arm architecture (w/ Fujitsu's proven microarchitecture)
    - SBSA: Server Base System Architecture
    - SBBR: Server Base Boot Requirements
    - VLA: Vector-Length Agnostic

    Assure binary compatibility
  - Fujitsu enhanced/maintained system software
    - Based on Linux & OSSs
    - Single source for x86 & Arm
      - Open MPI, OpenMP, Libraries,
      - Performance analyzer, Debugger

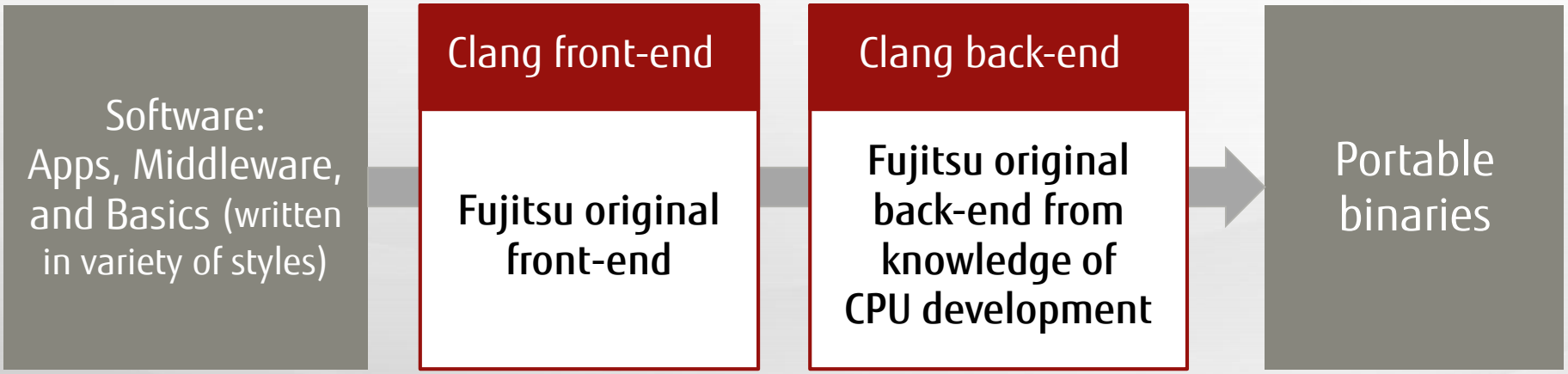
    Lowering barriers for single source development
  - Powerful ***but*** original compilers --- will be aligned to be useful & popular

Copyright 2017 FUJITSU LIMITED

Compilers to Increase Software Portability

FUJITSU

- Transform our original & powerful compilers to be all-around
  - Working and contributing for the **Clang project** to satisfy both high performance and portability



- Fujitsu's back-end advantage
  - Auto-parallelization for many-core architecture
  - Auto-vectorization for Scalable Vector Extension
  - Strong software pipelining with loop fission

Utilize Post-K uArch:

- Rich & wide SIMD
- Sector cache...

Copyright 2017 FUJITSU LIMITED

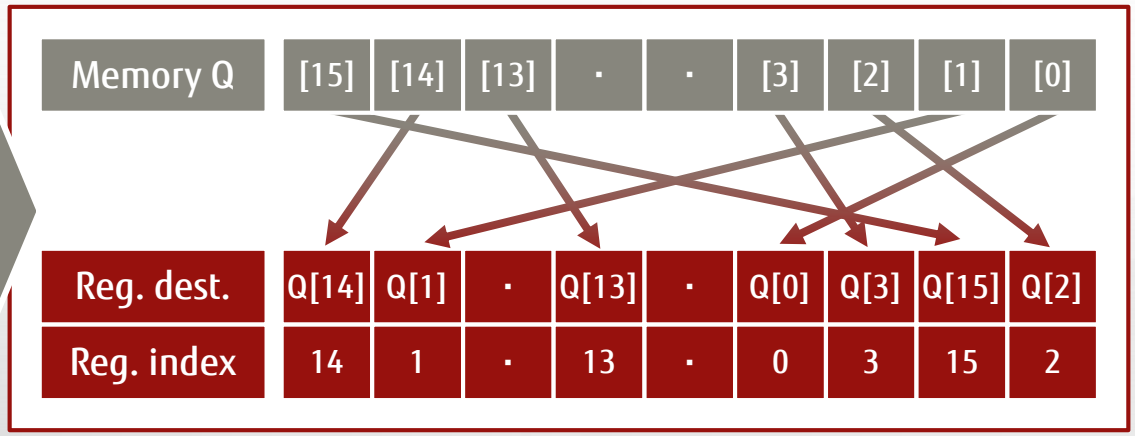
Auto-vectorization for Arm SVE

FUJITSU

4 Byte x 16 SIMD List Memory Access by utilizing 512bit Register

```
1  int index[n]
2  float P[n], Q[n];
3
4  for (i=0; i<n; ++i) {
5    P[i] = Q[index[i]];
6  }
7
```

SVE



Various Types of SIMD Optimization by Utilizing Predicate Registers

Loop including IF clause

```
1  for (int i=0; i<n; ++i) {
2    if (mask[i] !=0) { a[i] = b[i]; }
3  }
```

Small Loop less than SIMD length

```
1  for (int i=0; i<VL/2; ++i) {
2    a[i] = b[i] * c[i];
3  }
```

While Loop with Data Dependency

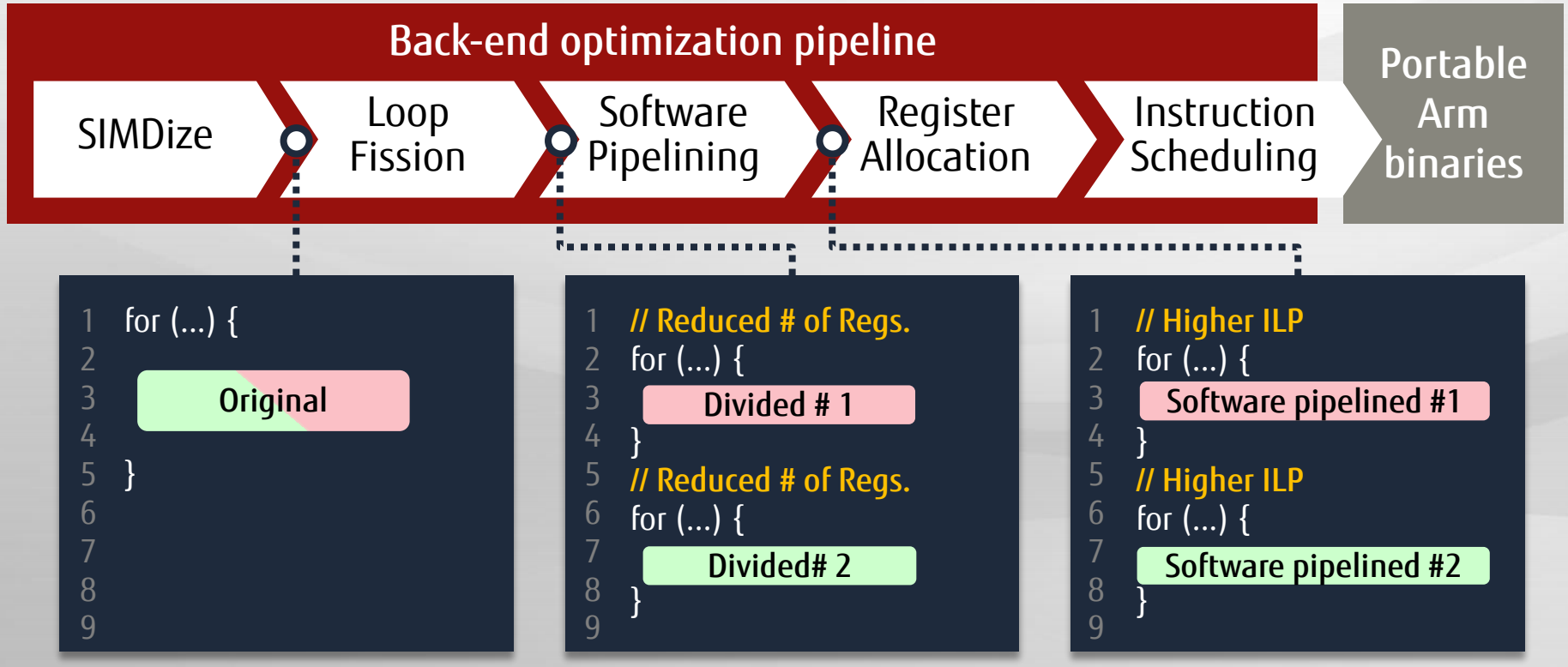
```
1  do {
2    b[i] = a[i];
3  } while(a[i++] != 0);
```


Copyright 2017 FUJITSU LIMITED

Fujitsu Compiler Back-end Optimization Flow

FUJITSU

- Loop Fission reduces required resources, such as registers
- Software Pipelining and Register Allocation
- Best utilization of hardware functions and resources

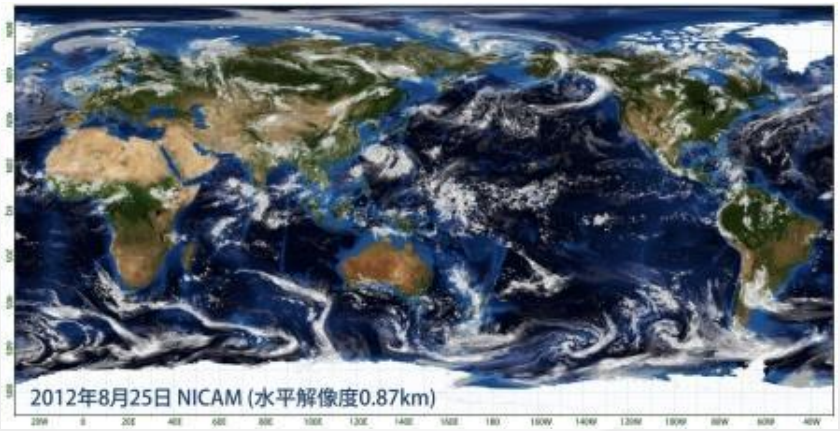


Copyright 2017 FUJITSU LIMITED

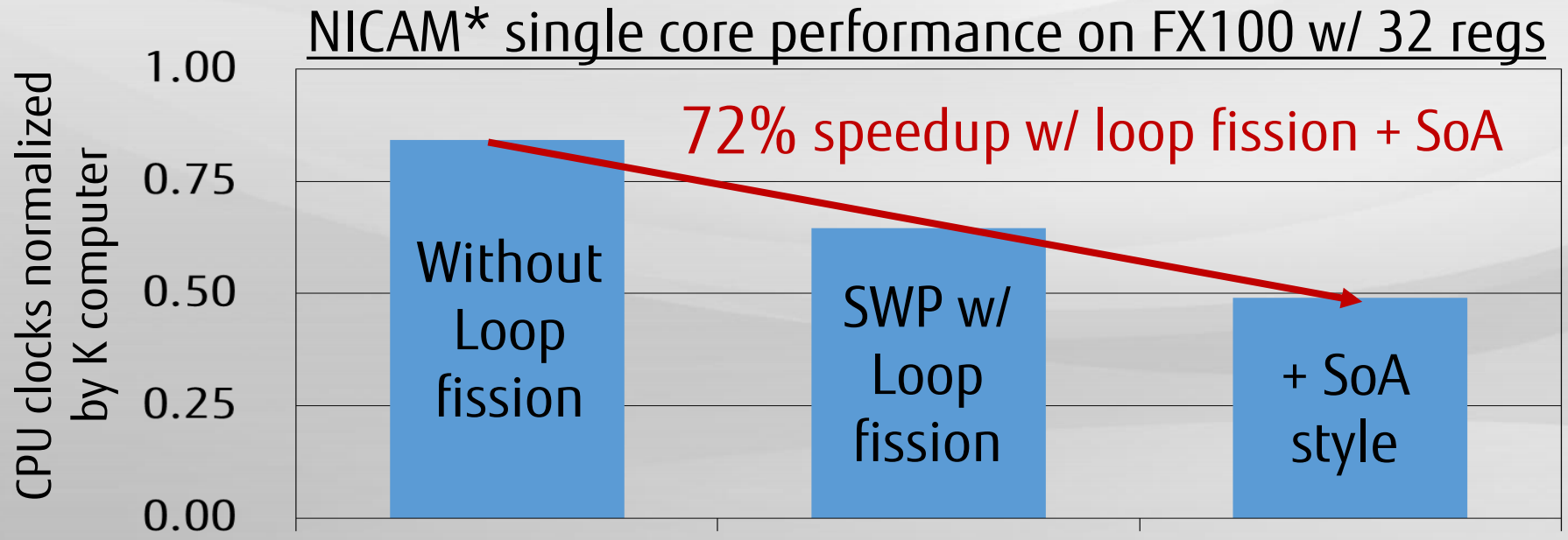
Effectiveness of SWP w/ Loop Fission and SoA

FUJITSU

- Runs on FX100 w/ 32 registers
  - 72% speed-up per core is observed
  - >2x speed-up compared w/ K computer
- Software Pipelining w/ Loop Fission utilizes CPU resources
- SoA-style layout extracts more



(Source: http://www.riken.jp/pr/topics/2013/20130920_1/)



*NICAM-DC-MINI: Climate simulations with fine mesh, https://github.com/fiber-miniapp/nicam-dc-mini

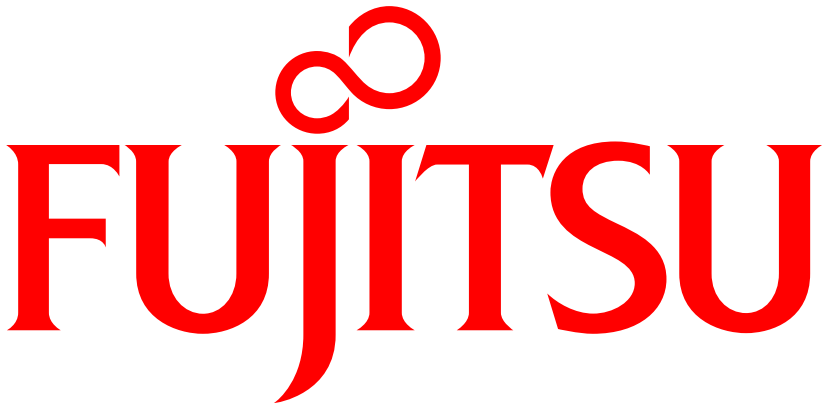
Copyright 2017 FUJITSU LIMITED

Summary

FUJITSU

- Fujitsu's Approach to HPC
  - Supporting high-end supercomputers with original CPU & x86 clusters
  - Developing the Post-K for app performance and low power consumption
  - Expecting more apps from OSS & ISVs through growing ecosystem
- Keys for Post-K Success
  - High performance standard-compliant HW and SW
  - All-around high performance compiler with binary compatibility
  - Many and varied high quality apps with x86 software compatibility
- Open & Highly Optimized Compilers
  - Clang + Fujitsu technologies
  - Tentative evaluation results are encouraging

Copyright 2017 FUJITSU LIMITED



shaping tomorrow with you