IEEE-CSC, ESAS and CSSJ SUPERCONDUCTIVITY NEWS FORUM (global edition), Issue No. 62, Jan. 2026. Presentation given at QUEST 2025, Yokohama, Japan, Oct. 2025.

Towards Cryogenic Superconductor Classical/Quantum Computing from a Computer Architecture Perspective

Koji Inoue

Kyushu University

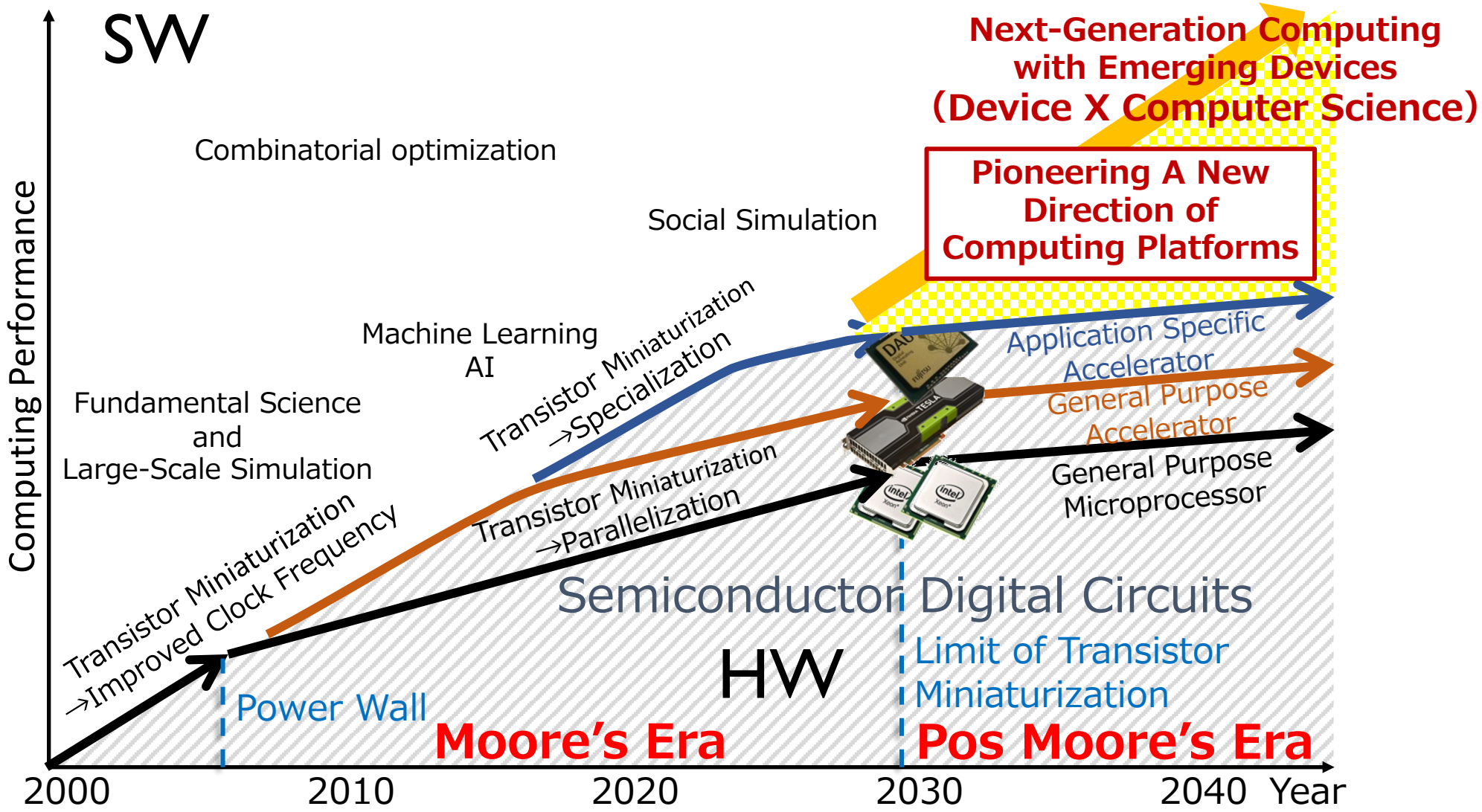
Senior Vice President

Professor: Department of Advanced Information Technology

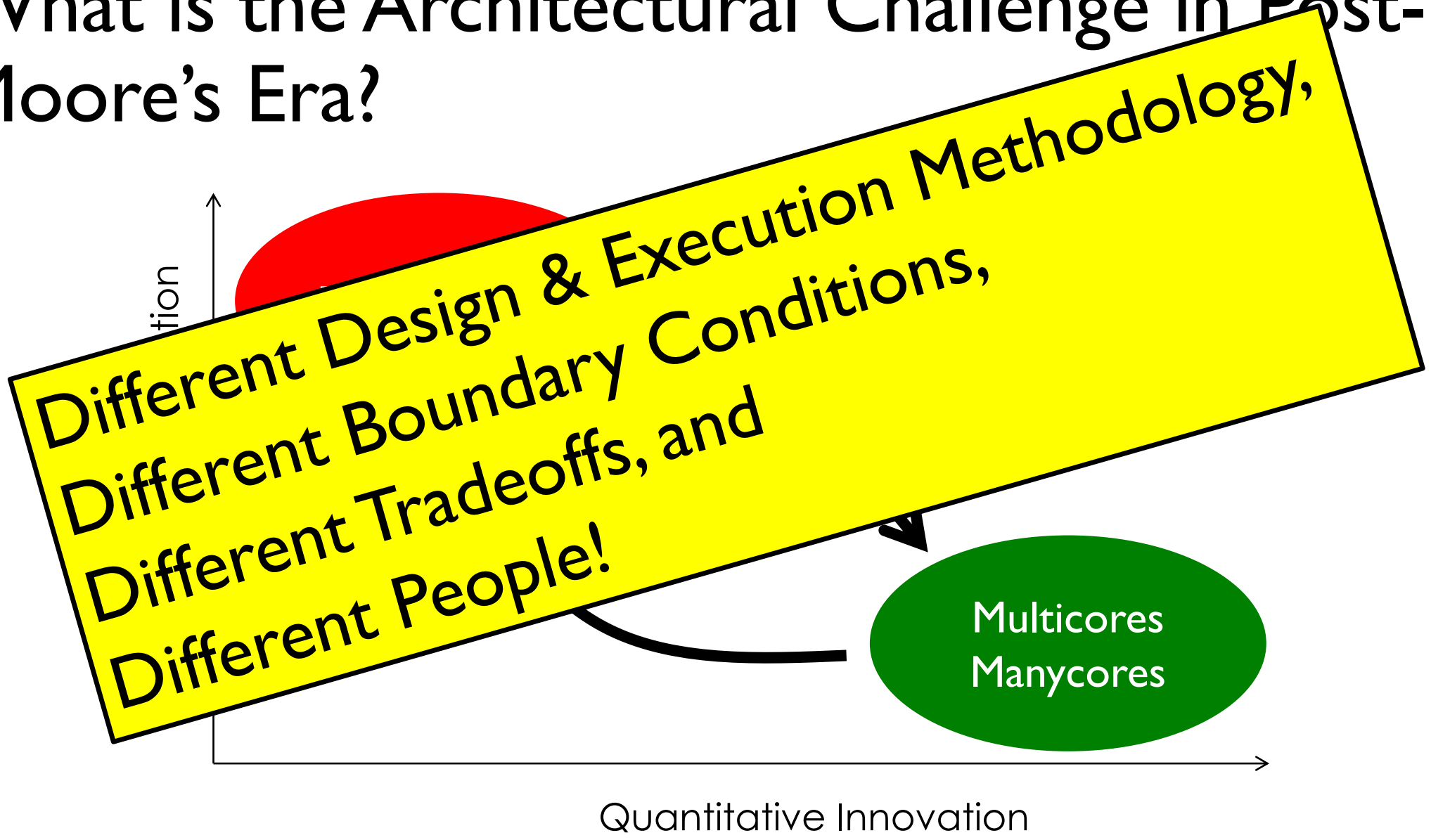
Director: the Quantum Computing System Center (QCSC)

Director: the System LSI Research Center (SLRC)

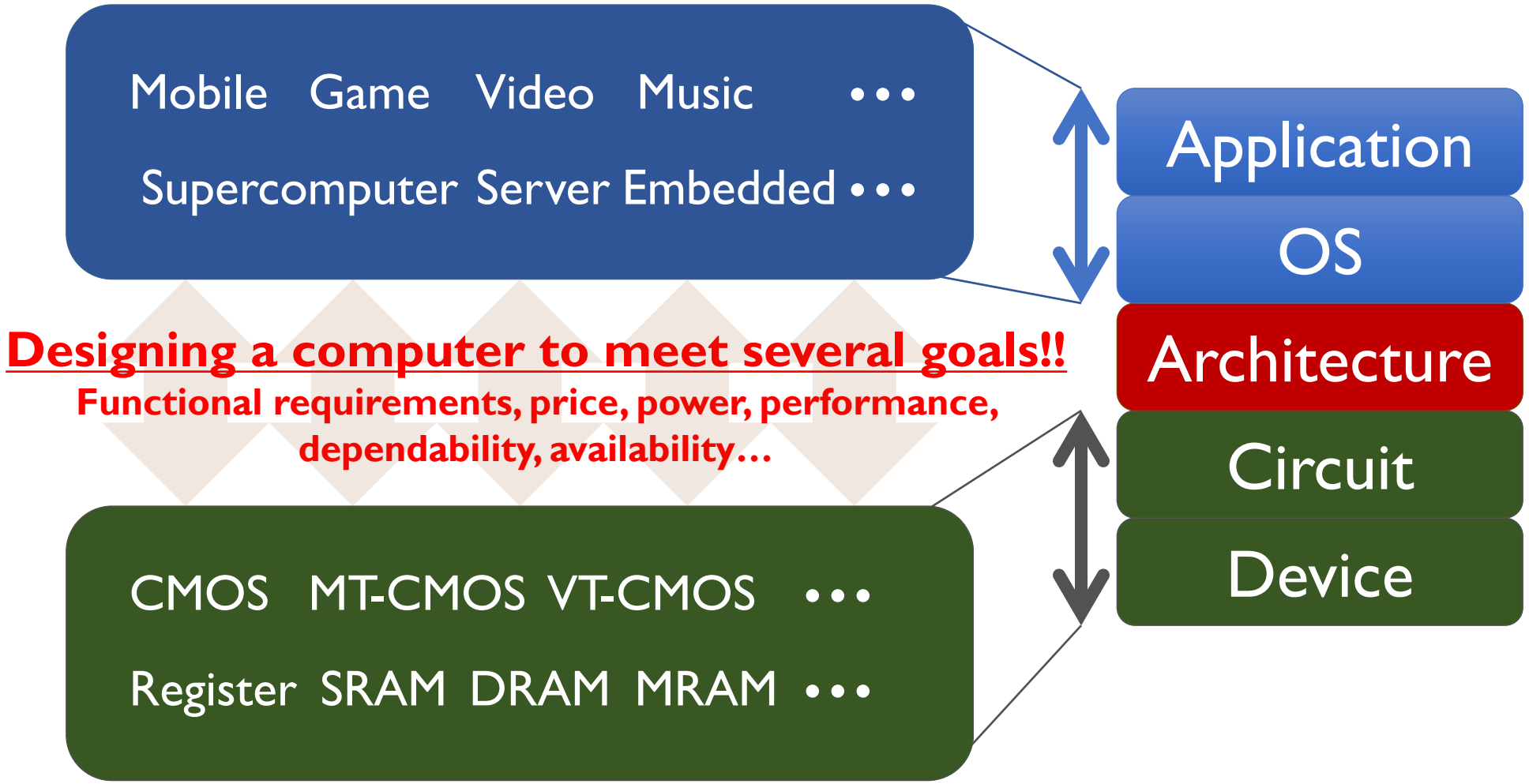
What is the Research Challenge?



What is the Architectural Challenge in Post-Moore's Era?

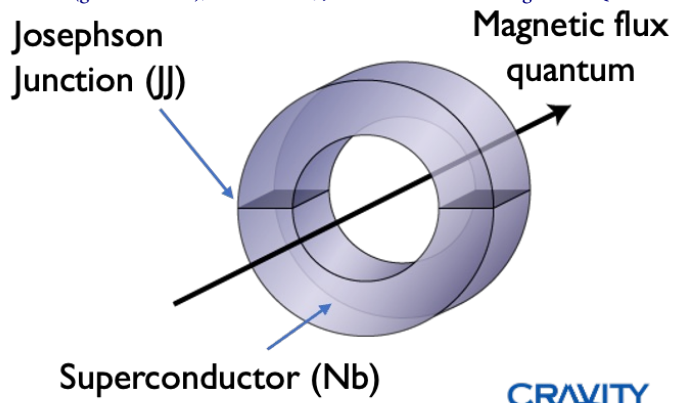


The Role of Computer Architects



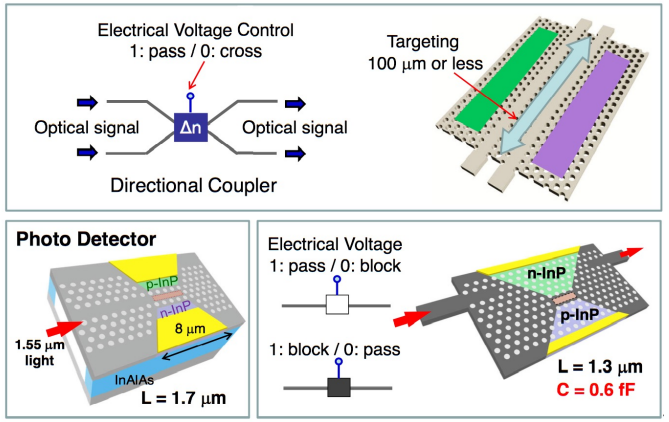
Collaborated Research

- Superconductor Classical Computing
- Quantum Computing
- Photonic Computing
- 3D Nanowire Computing
- VNFPGA-based Intermittent Computing

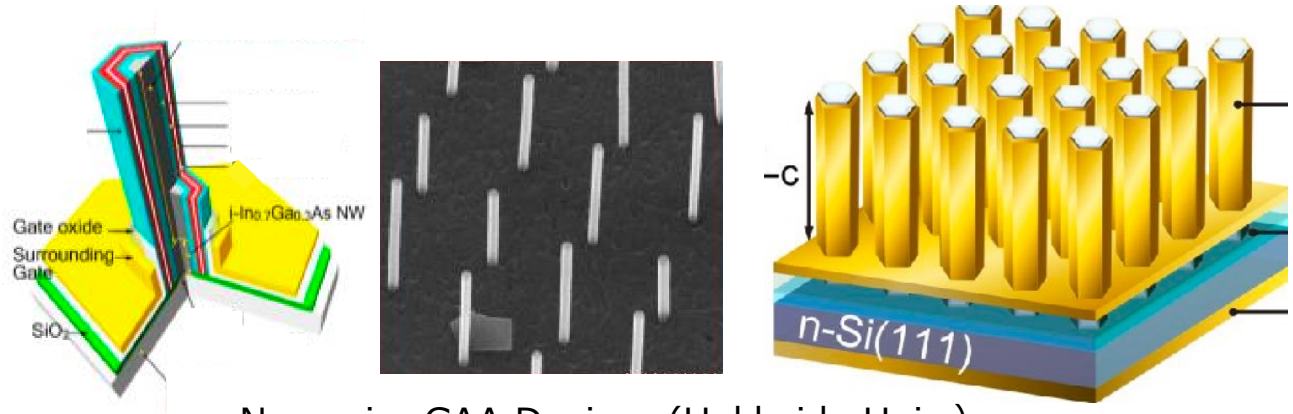


S. Nagasawa et al. *IEICE Trans. Electron.* **E97-C** (2014) 132–140.

Superconductor Devices (Nagoya U. / AIST)



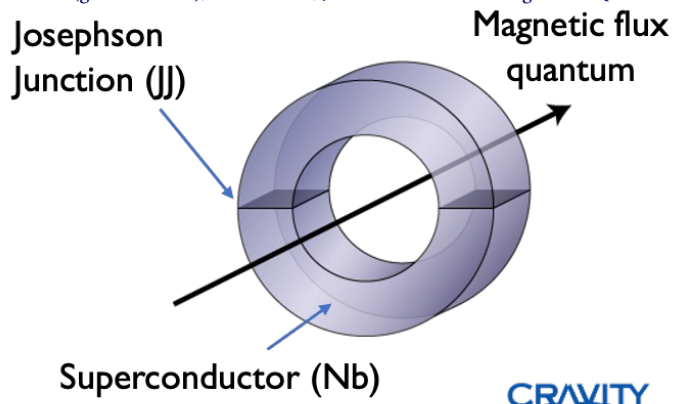
Optical Devices (NTT Lab.)



Nanowire GAA Devices (Hokkaido Univ.)

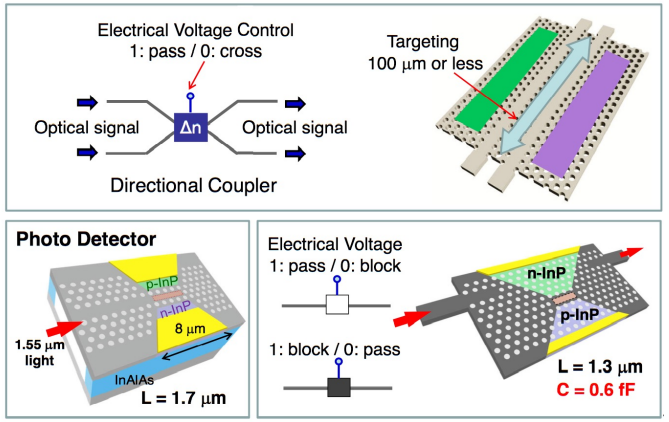
Collaborated Research

- **Superconductor Classical Computing**
- Quantum Computing
- Photonic Computing
- 3D Nanowire Computing
- VNFPGA-based Intermittent Computing

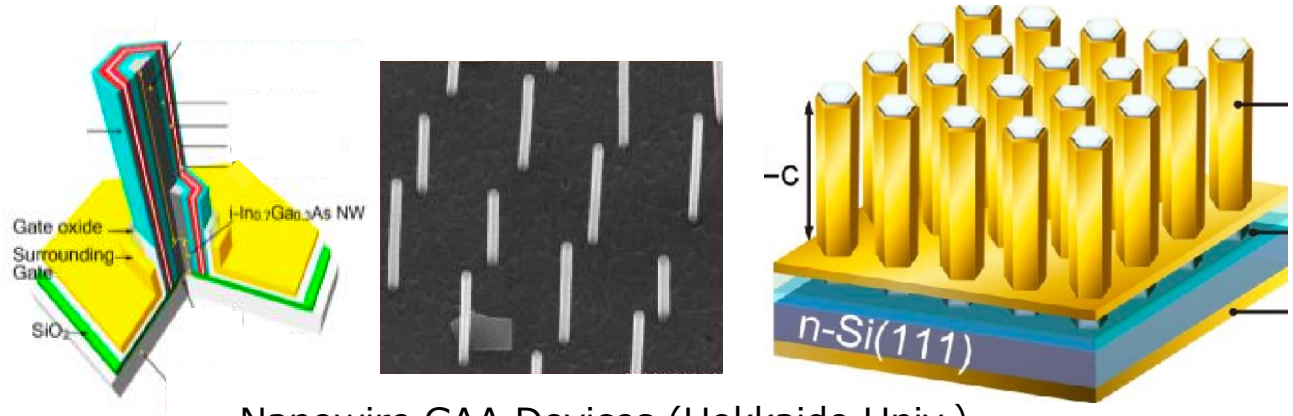


S. Nagasawa et al. *IEICE Trans. Electron.* **E97-C** (2014) 132–140.

Superconductor Devices (Nagoya U. / AIST)

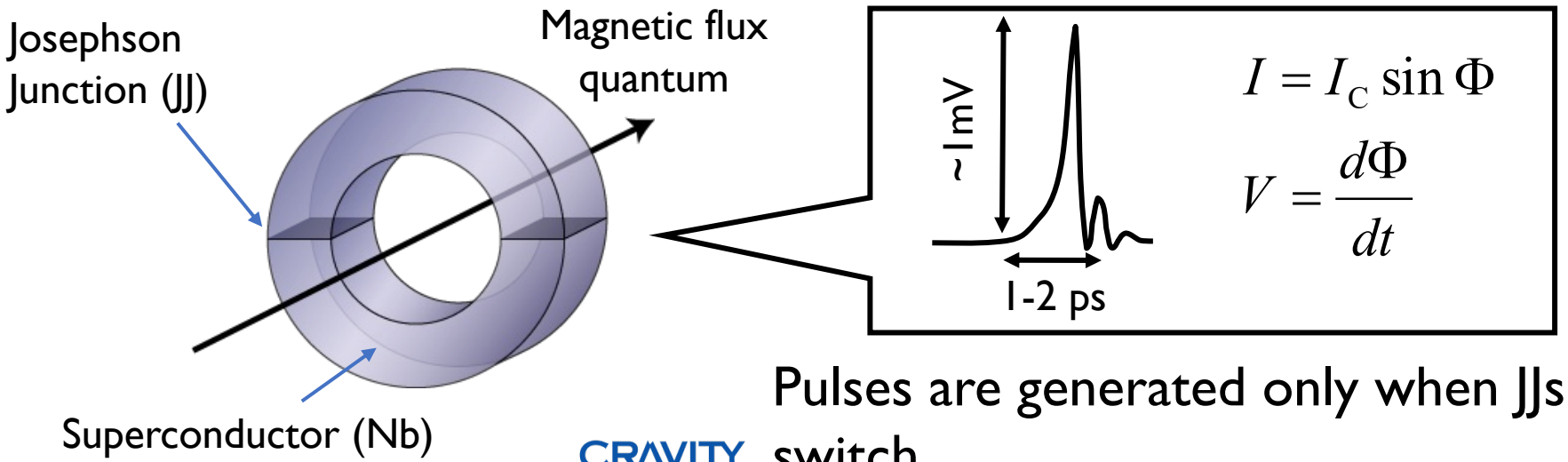


Optical Devices (NTT Lab.)



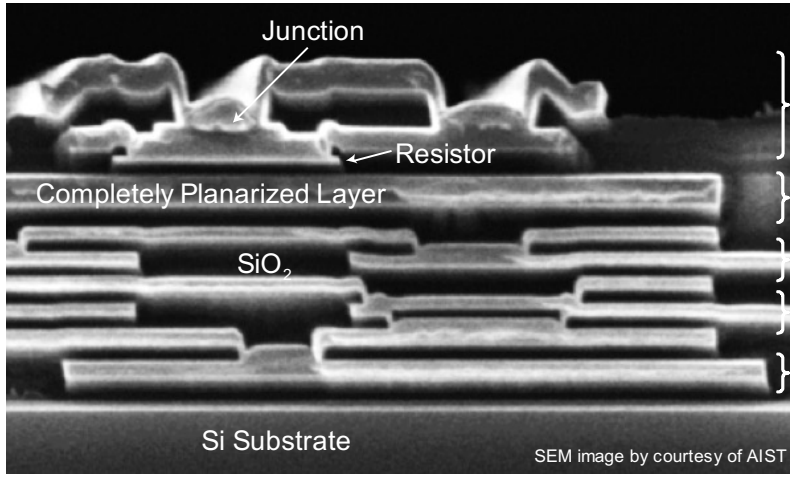
Nanowire GAA Devices (Hokkaido Univ.)

SFQ Device & Circuit



Pulses are generated only when JJs switch.

- **Extremely low power**
  ~1 μW gates operating at 100 GHz
- **High-speed operation**
  >100 GHz demonstrations, etc.
- **Ultrafast interconnects**
  Signal transmission at the speed of light (SFQ has no mass)

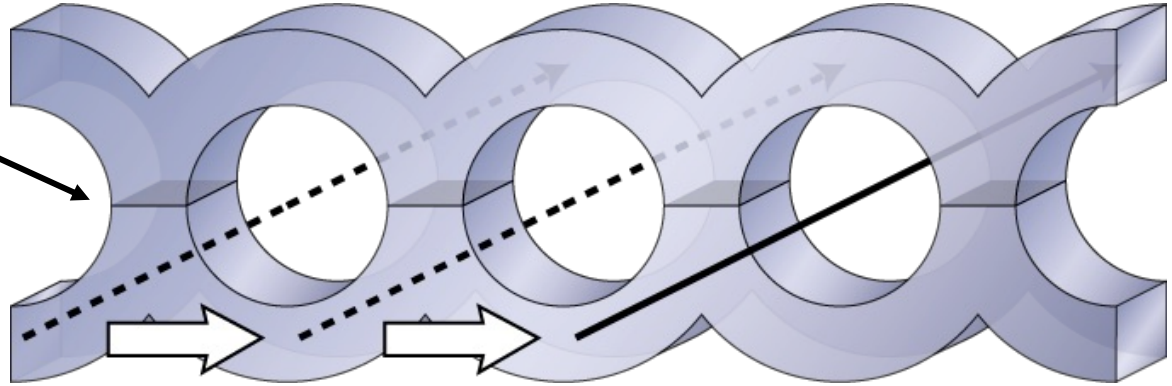


S. Nagasawa et al. *IEICE Trans. Electron.* **E97-C** (2014) 132–140.

Propagation of SFQ Signal

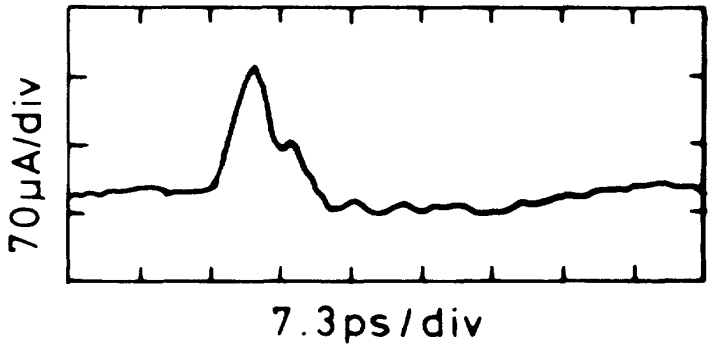
- Josephson transmission line (JTL)

Josephson junction (JJ)

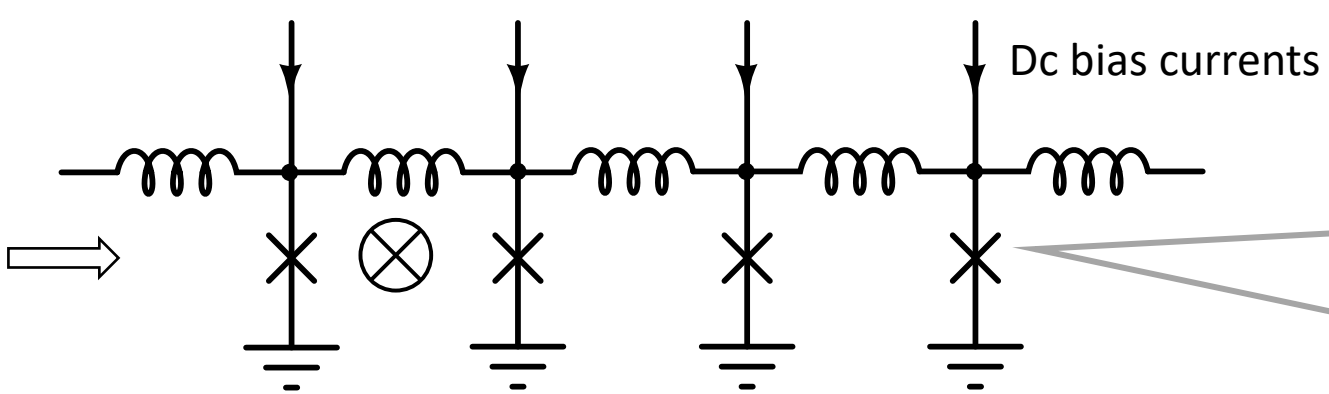


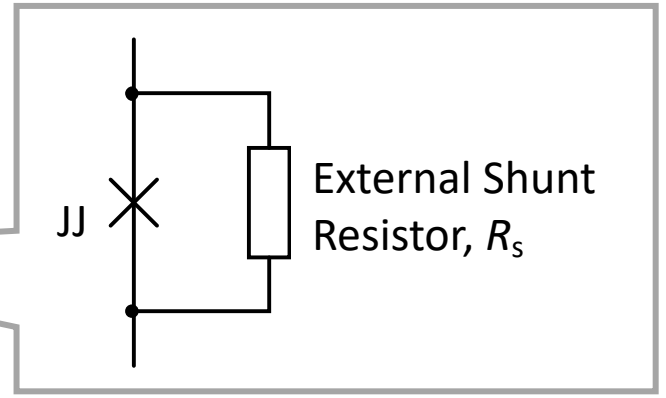
SFQ (single flux quantum)

Impulse-shape voltage only when JJ switches.
→ **Energy consumption**



A. Fujimaki et al., PRL 1987

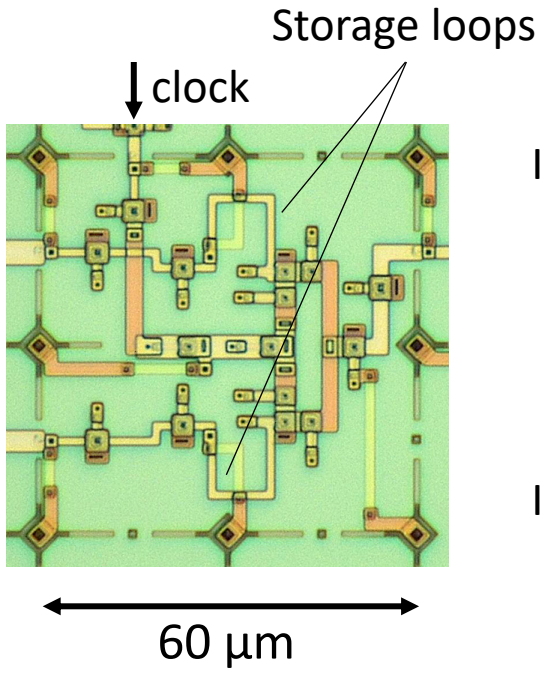


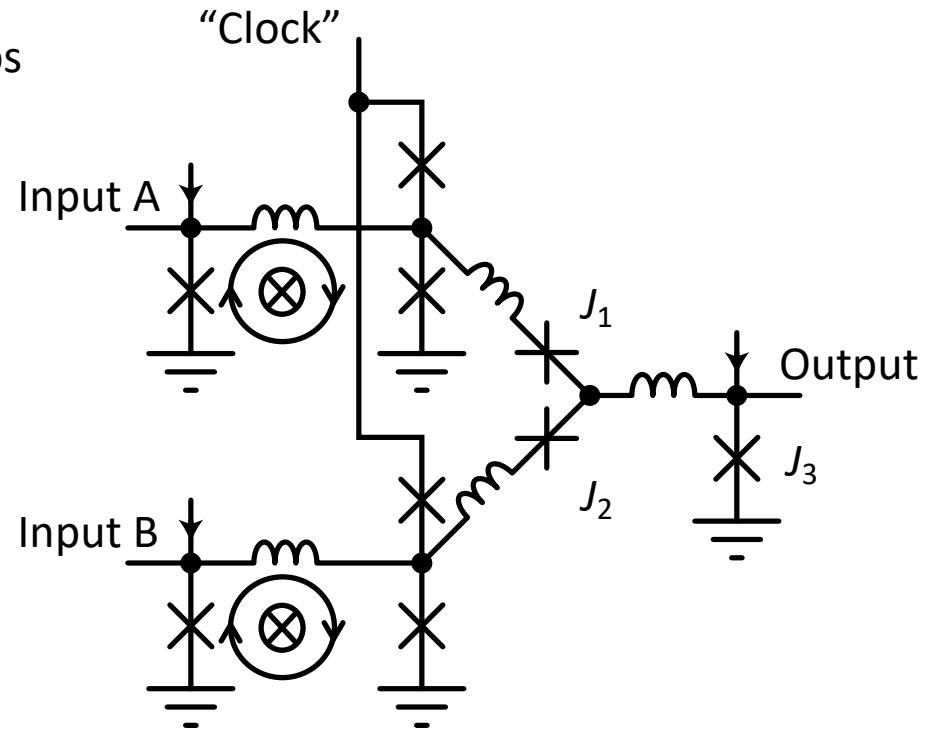


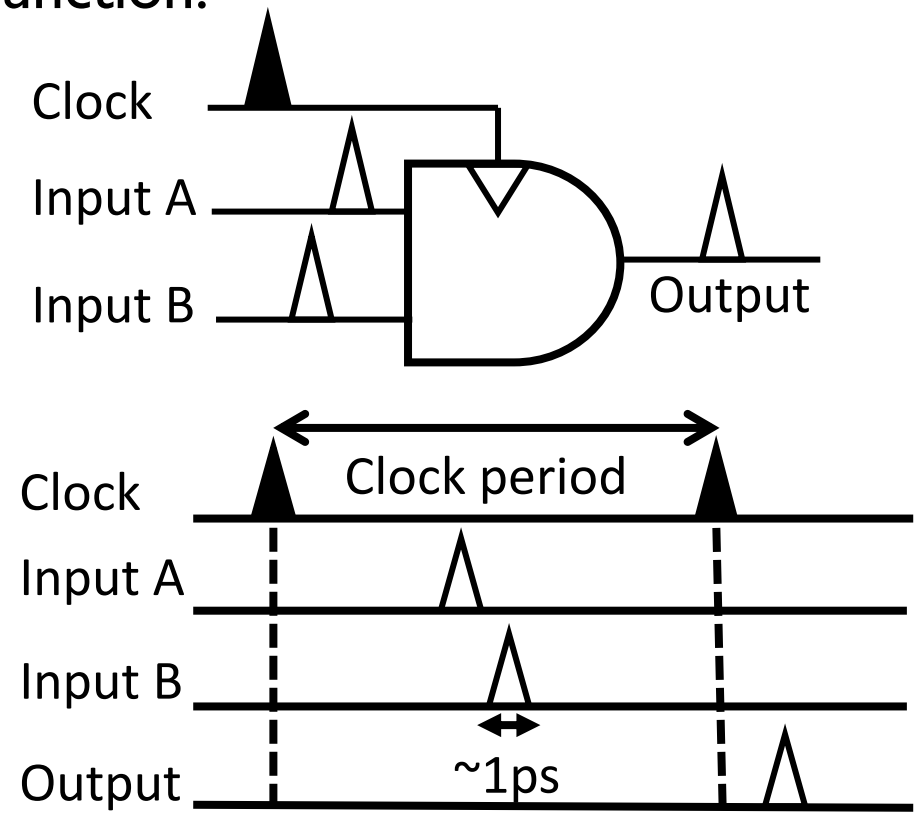
After K. K. Likharev and V. K. Semenov *IEEE Trans. Appl. Supercond.* **1** (1991).

SFQ Logic Gate (AND)

- Use "Clock" as a timing reference for synchronization.
- Every logic gate is clocked gate and has the latch function.





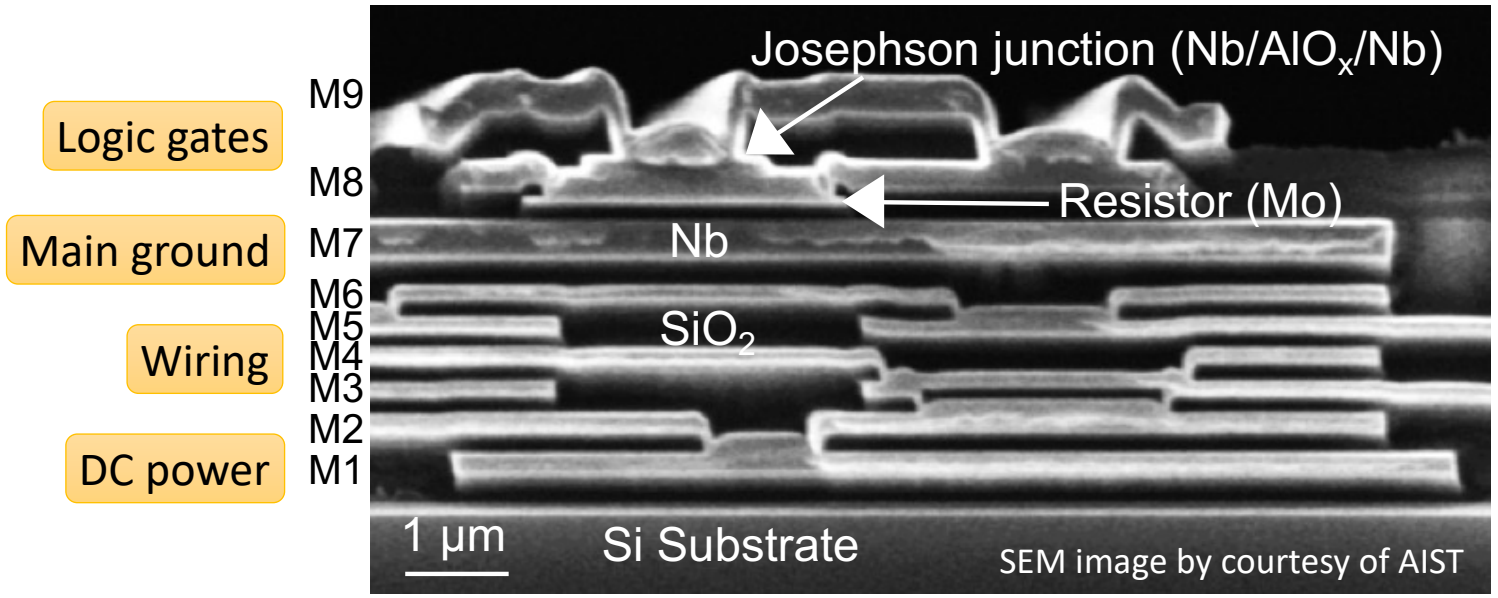


Fabrication Process

- 3–10 layer process is under development in Japan, US, and China.

AIST Advanced Process, Japan

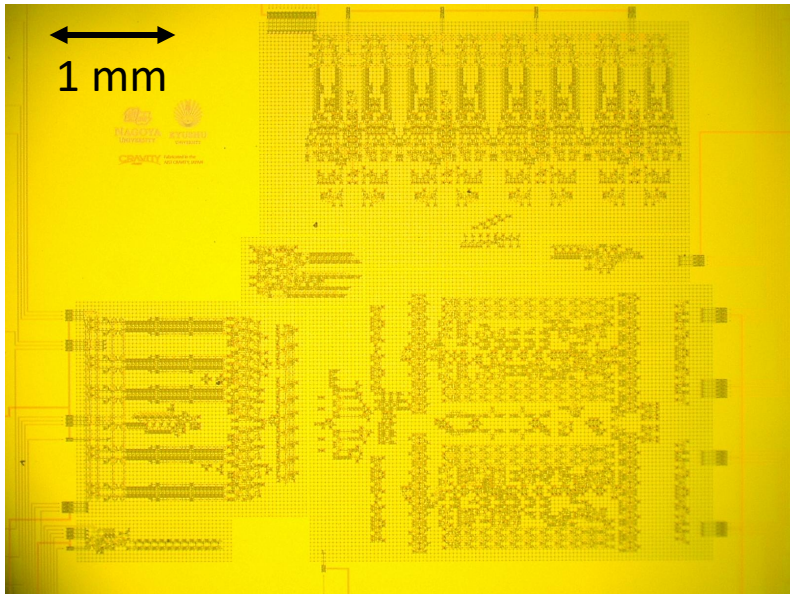
1-μm sq. JJ, Nb 9-layer + Mo



S. Nagasawa et al. *IEICE* **E97-C** (2014) 132-140.

32-GHz, 6.5-mW SFQ MPU

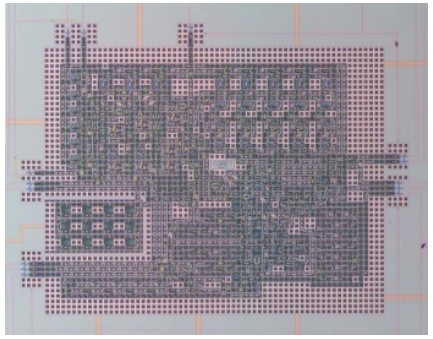
25,403 JJs, 4.1 x 5.3 mm^2



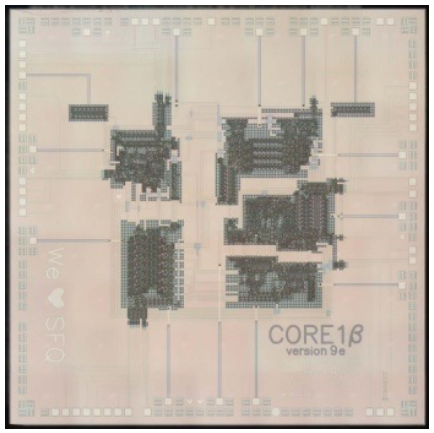
K. Ishida et al., VLSI 2020

Architectural Challenge on SFQ-based Computing

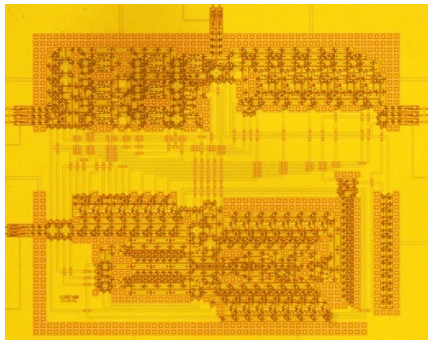
OLD Challenge in RSFQ-based Computer #1
~ SFQ microprocessor designs @ 2003-2016 ~



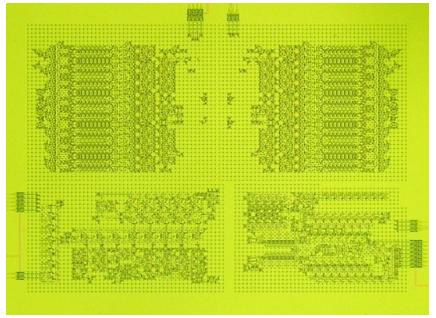
CORE1α v5 (2003)
4999 JJs, 15 GHz
167 MIPS, 1.6 mW



CORE1β v9e (2006)
10955 JJs, 25 GHz
1400 MOPS, 3.3 mW

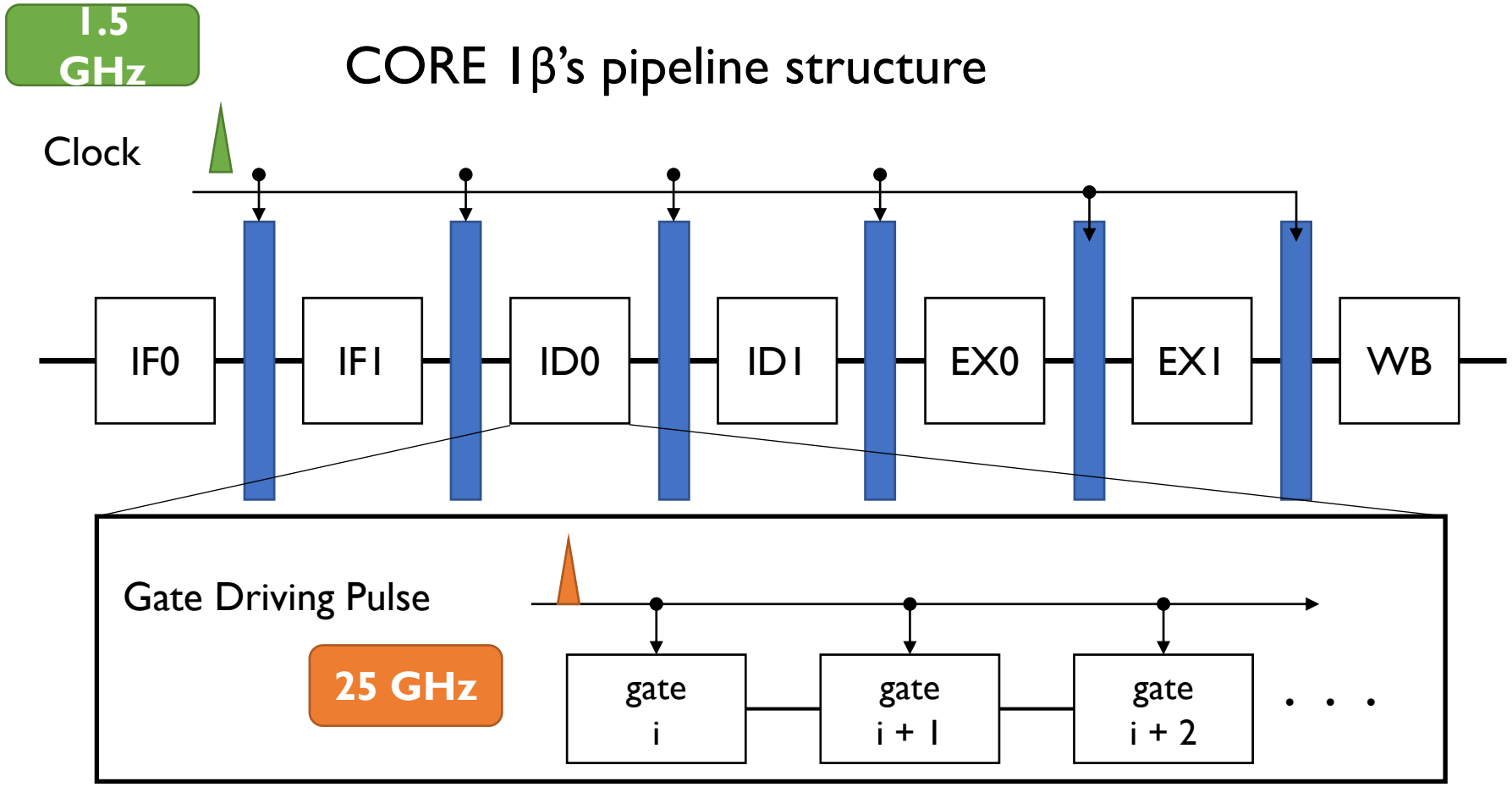


CORE100 (2015)
3073 JJs, 100 GHz
800 MIPS, 1.0 mW



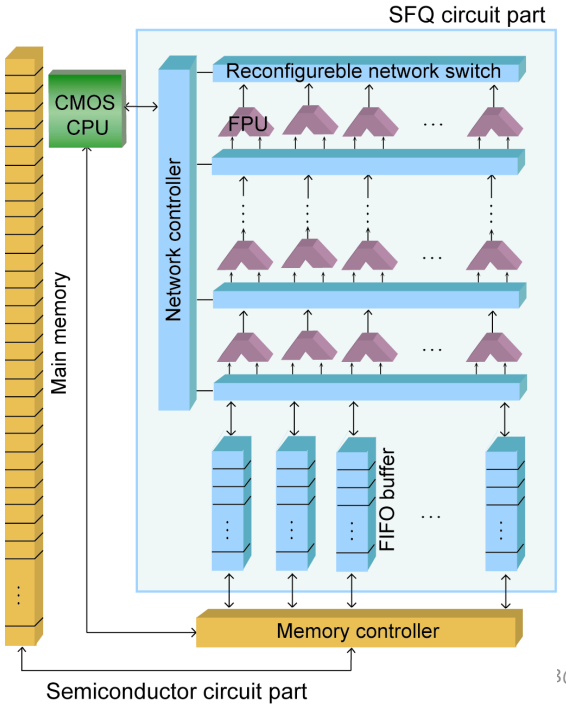
CORE e2 v5h (2016)
10603 JJs, 50 GHz
333 MIPS, 2.5 mW

OLD Challenge in RSFQ-based Computer #1
~ SFQ microprocessor designs @ 2003-2016 ~



OLD Challenge in RSFQ-based Computer #2
~ SFQ Reconfigurable Data-Path @ 2006-2012 ~

Large-Scale Reconfigurable Data-Path for SFQ : Architecture Level



- 1K FPUs operate at 80 GHz
- Re-configurable operand network
- Much simple organization for SFQ design (No feedback loops)
- Make a good balance between “Parallel Exe. Vs. Sequential Exe.”

How To Exploit A Number of FPUs: Algorithm Level

$$\frac{2\pi^{5/2}\exp\left(-\frac{ab}{a+b}(\mathbf{A}-\mathbf{B})^2\right)\exp\left(-\frac{cd}{c+d}(\mathbf{C}-\mathbf{D})^2\right)}{(a+b)(c+d)\sqrt{a+b+c+d}}F_m(T)$$

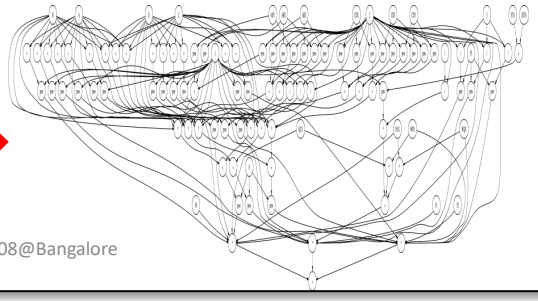
Computation of molecular orbital

while (I < 1000):

tei(4,4,4,4)=(((3+2*p*(4*PAx*PBx+PBx**2*PAx**2*(1+2*p*PBx**2)))*(3+2*q*(4*QCx*QDx+QDx**2+QCx**2*(1+2*q*QDx**2)))*f(0,t))/(p**2*q**2)+(4*(3+2*p*(4*PAx*PBx+PBx**2+PAx**2*(1+2*p*PBx**2)))*PQx*(QCx+QDx)*(3+2*q*QCx*QDx)*f(1,t))/(p*q*(p+q))(4*(PAx+PBx)*(3+2*p*PAx*PBx)*PQx*(3+2*q*(4*QCx*QDx+QDx**2+QCx**2*(1+2*q*QDx**2)))*f(1,t))/(p*q*(p+q))(8*(PAx+PBx)*(3+2*p*PAx*PBx)*(QCx+QDx)*(3+2*q*QCx*QDx)*(((p+q)*f(1,t))+2*p*PQx**2*q*f(2,t)))/(p*q*(p+q)**2)+(2*(3+2*p*(4*PAx*PBx+PBx**2+PAx**2*(1+2*p*PBx**2)))*(3+q*(QCx**2+4*QCx*QDx+QDx**2))*(((p+q)*f(1,t))+2*p*PQx**2*q*f(2,t)))/(p*q**2*(p+q)**2)+(2*(3+p*(PAx**2+4*PAx*PBx+PBx**2))*(3+2*q*(4*QCx*QDx+QDx**2+QCx**2*(1+2*q*QDx**2)))*(((p+q)*f(1,t))+2*p*PQx**2*q*f(2,t)))/(p**2*q*(p+q)**2)+(4*(3+2*p*(4*PAx*PBx+PBx**2+PAx**2*(1+2*p*PBx**2)))*PQx*(QCx+QDx)*(3*(p+q)*f(2,t)+2*p*PQx**2*q*f(3,t)))/(q*(p+q)**3)\+(8*(3+p*(PAx**2+4*PAx*PBx+PBx**2))*PQx*(QCx+QDx)*(3+2*q*QCx*QDx)*(3*(p+q)*f(2,t)+2*p*PQx**2*q*f(3,t)))/(p*(p+q)**3))(8*(PAx+PBx)*(3+2*p*PAx*PBx)*PQx*(3+q*(QCx**2+4*QCx*QDx+QDx**2))*(3*(p+q)*f(2,t)+2*p*PQx**2*q*f(3,t)))/(q*(p+q)**3)(4*(PAx+PBx)*PQx*(3+2*q*(4*QCx*QDx+QDx**2+QCx**2*(1+2*q*QDx**2)))*(3*(p+q)*f(2,t)+2*p*PQx**2*q*f(3,t)))/(p*(p+q)**3)+((3+2*p*(4*PAx*PBx+PBx**2+PAx**2*(1+2*p*PBx**2)))*(3*(p+q)**2*f(2,t)+4*p*PQx**2*q*(3*(p+q)*f(3,t)+p*PQx**2*q*f(4,t)))))/(q**2*(p+q)**4)(8*(PAx+PBx)*(3+2*p*PAx*PBx)*(QCx+QDx)*(3*(p+q)**2*f(2,t)+4*p*PQx**2*q*(3*(p+q)*f(3,t)+p*PQx**2*q*f(4,t)))))/(q*(p+q)**4)(8*(PAx+PBx)*(QCx+QDx)*(3+2*q*QCx*QDx)*(3*(p+q)**2*f(2,t)+4*p*PQx**2*q*(3*(p+q)*f(3,t)+p*PQx**2*q*f(4,t)))))/(p*q*(p+q)**4)+(4*(3+p*(PAx**2+4*PAx*PBx+PBx**2))*(3+q*(QCx**2+4*QCx*QDx+QDx**2))*(3*(p+q)**2*f(2,t)+4*p*PQx**2*q*(3*(p+q)*f(3,t)+p*PQx**2*q*f(4,t)))))/(p*q*(p+q)**4)+((3+2*q*(4*QCx*QDx+QDx**2+QCx**2*(1+2*q*QDx**2)))*(3*(p+q)**2*f(2,t)+4*p*PQx**2*q*(3*(p+q)*f(3,t)+p*PQx**2*q*f(4,t)))))/(p*q*(p+q)**4)+(4*(3+2*p*(4*PAx*PBx+PBx**2+PAx**2*(1+2*p*PBx**2)))*PQx*(15*(p+q)**2*f(3,t)+4*p*PQx**2*q*(5*(p+q)*f(4,t)+p*PQx**2*q*f(5,t))))/(q*(p+q)**5)+(8*(3+p*(PAx**2+4*PAx*PBx+PBx**2))*PQx*(QCx+QDx)*(15*(p+q)**2*f(3,t)+4*p*PQx**2*q*(5*(p+q)*f(4,t)+p*PQx**2*q*f(5,t))))/(p+q)**5+(4*PQx*q*(QCx+QDx)*(3+2*q*QCx*QDx)*(15*(p+q)**2*f(3,t)+4*p*PQx**2*q*(5*(p+q)*f(4,t)+p*PQx**2*q*f(5,t))))/(p+q)**5)(8*(PAx+PBx)*PQx*(3+q*(QCx**2+4*QCx*QDx+QDx**2))*(15*(p+q)**2*f(3,t)+4*p*PQx**2*q*(5*(p+q)*f(4,t)+p*PQx**2*q*f(5,t)))8*p**3*PQx**6*q**3*f(6,t))))/(p+q)**6+(2*(3+q*(QCx**2+4*QCx*QDx+QDx**2))*(15*(p+q)**3*f(3,t)30*p*PQx**2*q*(p+q)*(3*(p+q)*f(4,t)+2*p*PQx**2*q*f(5,t))+8*p**3*PQx**6*q**3*f(6,t))))/(p*(p+q)**6)+(2*(3+q*(QCx**2+4*QCx*QDx+QDx**2))*(15*(p+q)**3*f(3,t)30*p*PQx**2*q*(p+q)*(3*(p+q)*f(4,t)+2*p*PQx**2*q*f(5,t))+8*p**3*PQx**6*q**3*f(6,t))))/(p*(p+q)**6)

→ 787 MUL, 261 ADD, 69 FUNC

I ++:

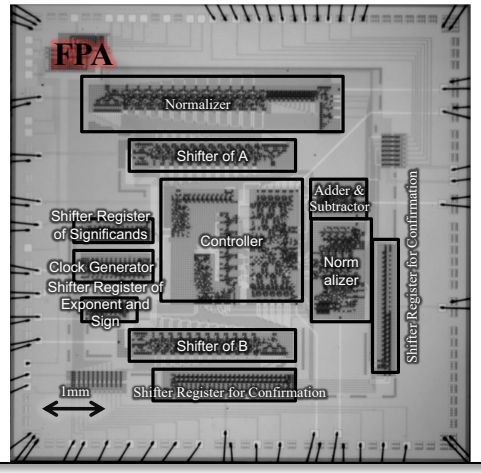


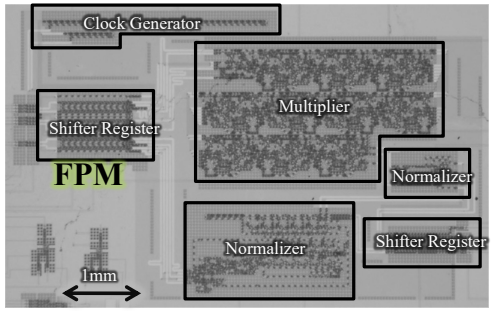
OLD Challenge in RSFQ-based Computer #2
~ SFQ Reconfigurable Data-Path @ 2006-2012 ~

Summary of Half-Precision FPUs

	Floating Point Adder	Floating Point Multiplier
# of JJs	11700	11044
Size (mm^2)	6.76 x 4.96	6.22 x 3.78
Minimum interval (clocks)	12 (n_f + 1)	
Latency (clocks)	23 (2 n_f + 1)	

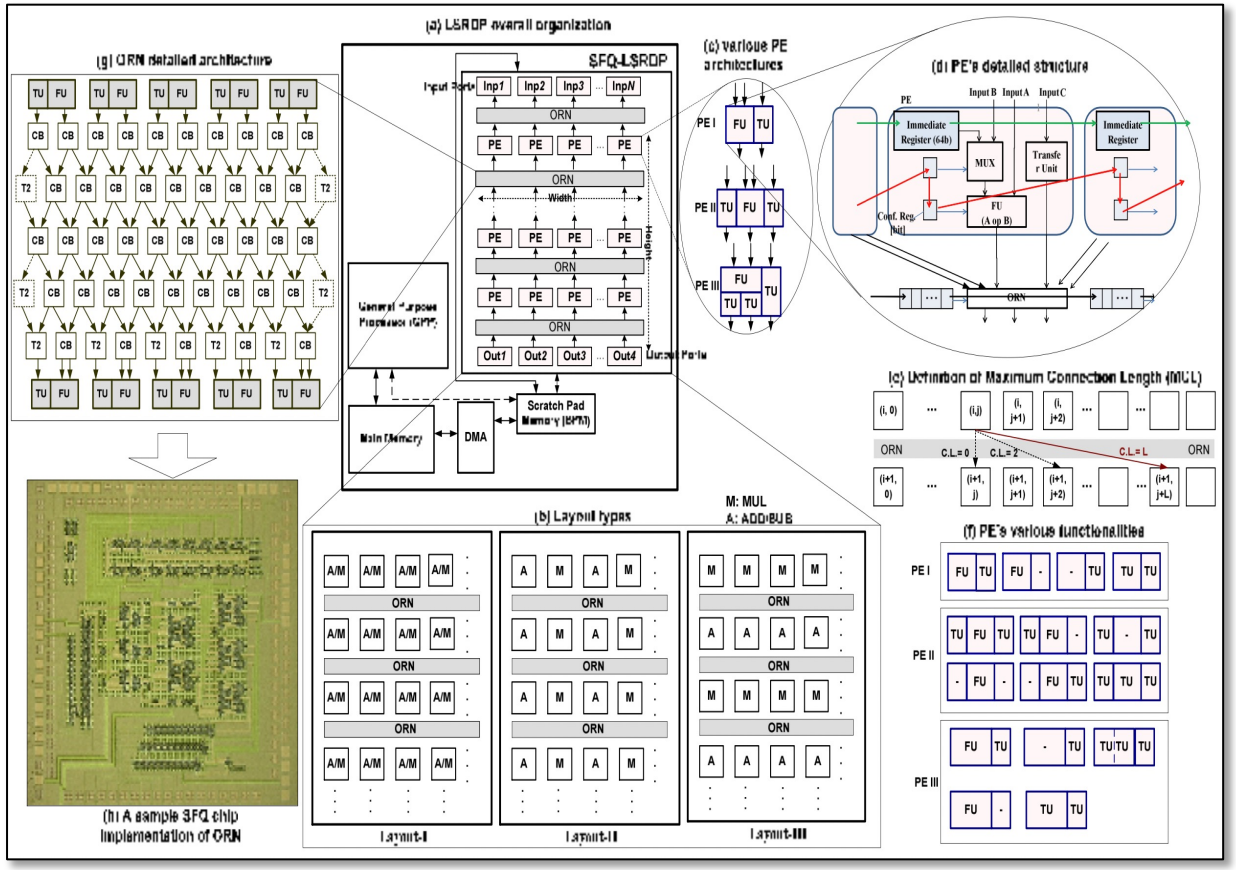
n_f : bit length of fraction part





CONNECT
cooperated with SRL, NiCT, NU & YNU

N. Yoshikawa, "RSFQ Project in Japan," 5th FLUXONICS RSFQ workshop, 2008.

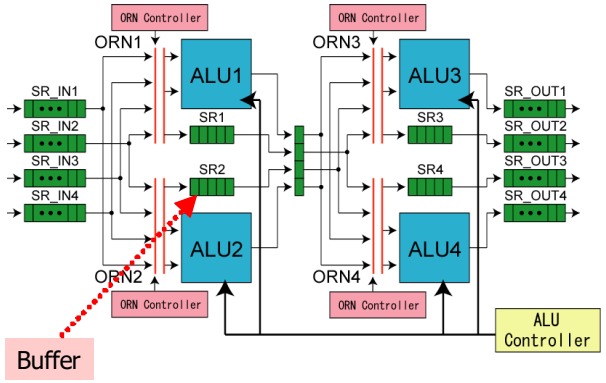


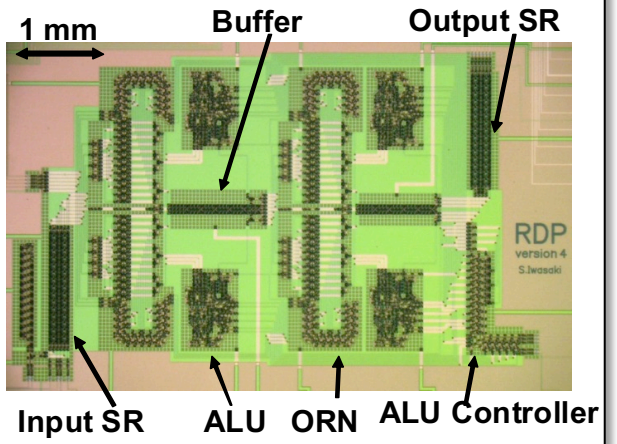
F. Mehdipour et al., "Mapping scientific applications on a large-scale data-path accelerator implemented by single-flux quantum (SFQ) circuits," DATE 2010.

OLD Challenge in RSFQ-based Computer #2
~ SFQ Reconfigurable Data-Path @ 2006-2012 ~

Design of 2x2 SFQ-RDP

- 11 pipeline stages
- Designed frequency : 25 GHz
- InSR & OutSR length : 16-bits
- Data length: 7-bits
- Bias current: 1.27 A
- Circuit area : 5.90 x 3.68 mm^2
- 10839 JJs





	SFQ-RDP
Application	HPC (MO)
Data Reuse (on-chip)	Low
Operation	Bit-Serial FP
On-chip network	Complex & Frexible
On-chip memory	Simple input/output buf
Optimization mainly focused	DFG mapping and routing

Panel Discussion @ ISLPED 2008

What we learned…

+ Stream processing sounds suitable for SFQ logics (no feedback loops)

- Bit-Serial designs significantly degrade the computation performance
- Memory wall problem becomes critical
- Complex on-chip communications consume a lot of JJs

Revisiting Microarchitecture for SFQ Logic ~Our First Trial~

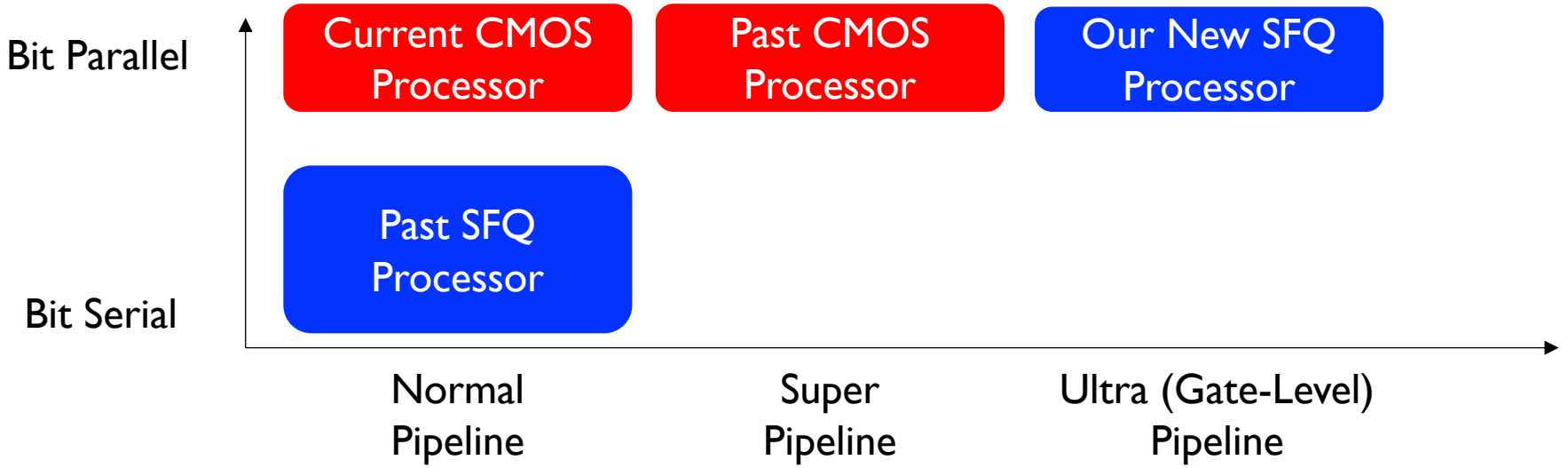
Revisiting RSFQ Microarchecture

Pitfall

Bit-serial operation is suitable for RSFQ designs!

Our Approach

Bit-parallel operation + Gate-level deep pipelining

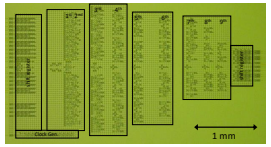


Our Contributions

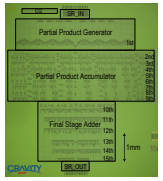
- Demonstration of bit-parallel gate-level pipelining [ISLPED'17]
- 30-50 GHz SFQ circuit design and fabrication [ISSCC'19, VLSI'20]
- Neural network SFQ accelerator [MICRO'20, IEEE Micro Top Picks]
- General purpose SFQ processor [MICRO'24, MICRO'25]
- Design space exploration of superconductor QCs [ISCA'22, ISCA'23]
- Superconductor quantum and classical hybrid [TQE'24]

World Fastest Chips!

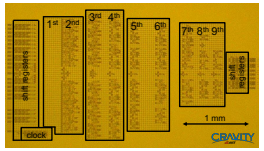
Fabricated Chip	Purpose	Frequency [GHz]	Power [mW]	Efficiency [TOPS/W]	#of JJs	Year
1: 8-bit ALU	First demo. of gate-level pipeline	56	1.6	40	4,846	2017
2: 8-bit array-type multiplier	large-scale circuit design	48	5.6	8.5	20,251	2018
3: low voltage 8-bit ALU	0.5mV low-voltage operation	30	0.276	109	7,451	2019
4: low-voltage 4-bit multiplier	large-scale low-voltage operation	51	0.134	381	4,498	2019
5: 4-bit microprocessor	large-scale datapath	32	6.5	2.5	25,403	2019
6: low-voltage 4-bit MAC	basic function for AI acceleration	38	0.366	104	9,739	2020
7: 2x2 systolic PE array	prototype of *SuperNPU*	34	0.711	382	9,263	2021



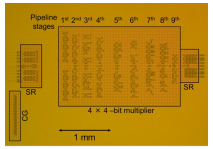
1



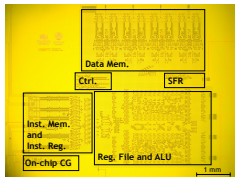
2



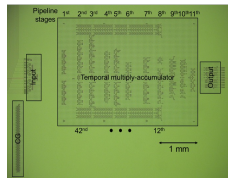
3



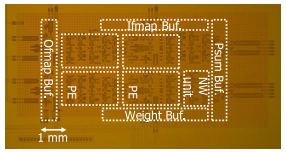
4



5



6

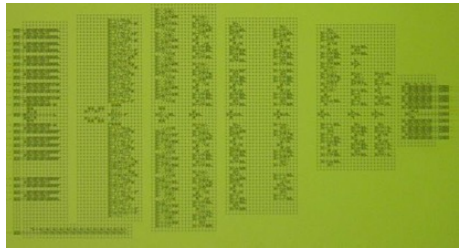


7

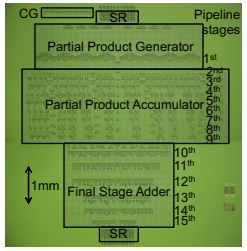
Our Approach & Outcome

Device/Circuit Optimization

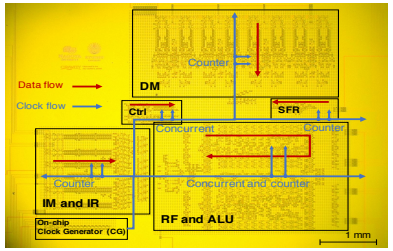
Architectural Optimization



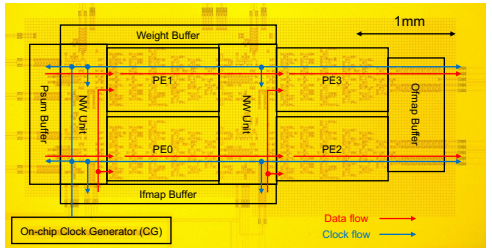
56GHz 1.6mW ALU
ISLPED'17 Design Contest
Honorable Mention



48GHz 5.6mW Multiplier
ISSCC'19
SilkRoad Award



32GHz 6.2mW Processor
VLSI Symposium'20
Selected as a featured paper

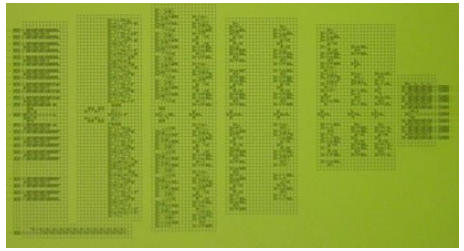


50GHz AI Accelerator
MICRO'20

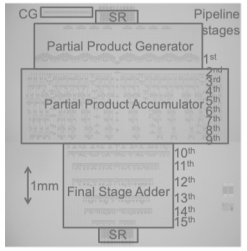
Our Approach & Outcome

Device/Circuit Optimization

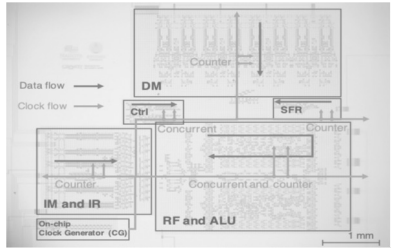
Architectural Optimization



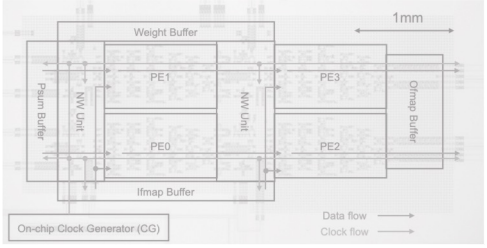
56GHz 1.6mW ALU
ISLPED'17 Design Contest
Honorable Mention



48GHz 5.6mW Multiplier
ISSCC'19
SilkRoad Award



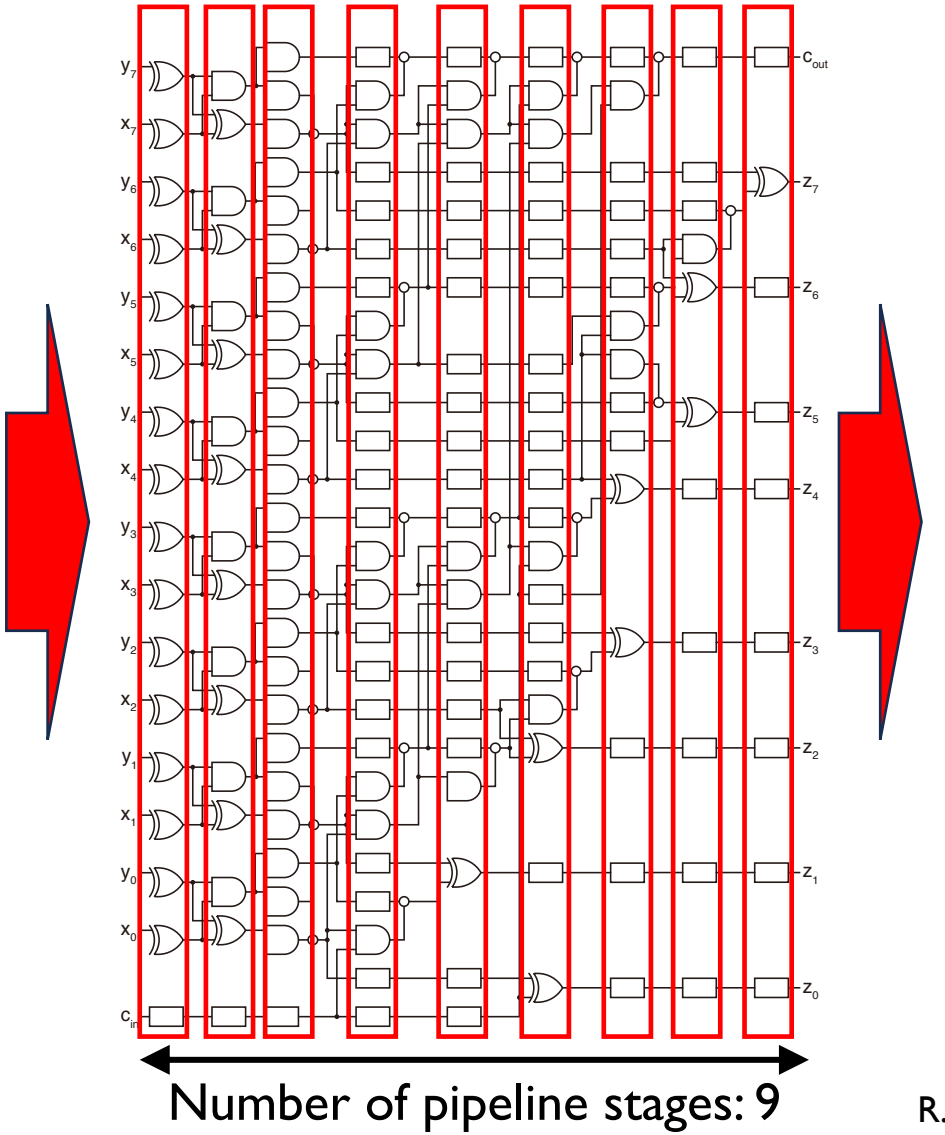
32GHz 6.2mW Processor
VLSI Symposium'20
Selected as a featured paper



50GHz AI Accelerator
MICRO'20

Masamitsu Tanaka et al., "1.6-mW, 56-GHz Arithmetic Logic Unit Based on Superconductor Single-Flux-Quantum Logic Circuit," ACM/IEEE International Symposium on Low Power Electronics and Design, Design Contest Honorable Mention, Aug. 2017.

8-bit Bit-Parallel ALU Design



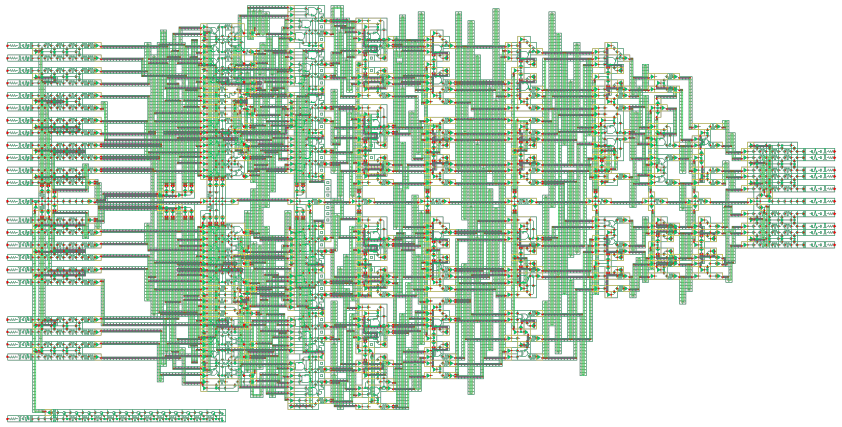
- ✓ **Target frequency: 50 GHz**
- ✓ **Gate-level pipelining**
- ✓ Functions: ADD, SUB, AND OR, XOR, NOR, etc.
- ✓ Data length: 8 bits

Based on Brent-Kung adder

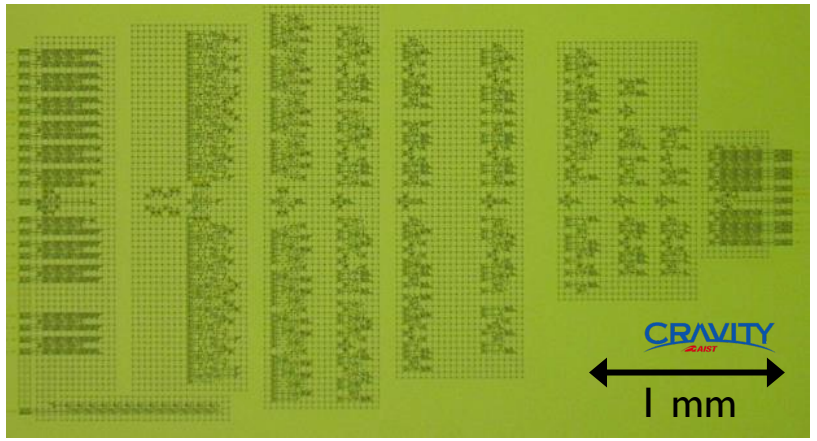
- Minimum number of logic gates (w/o D flip-flops)
- Sparse wiring tracks
- Small fanouts (Max. 3)
- Maximum logic depth

R. Brent and H. Kung, *IEEE Trans. Comput.* **G31** (1982) 260

It Works!

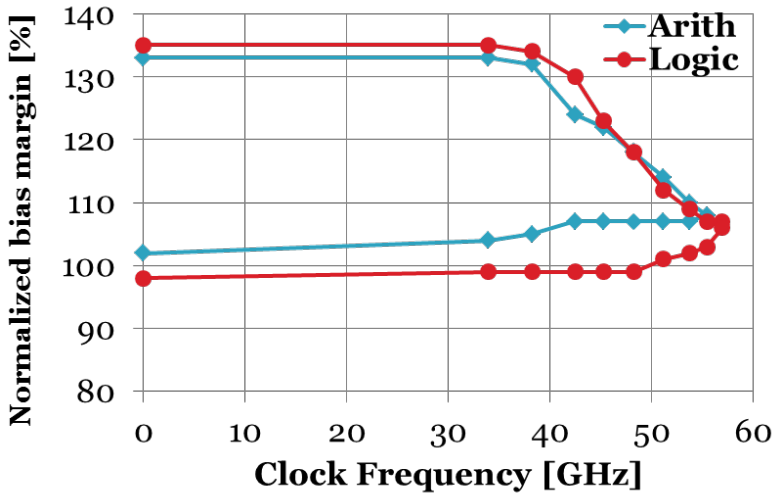


ISLPED'17 Design Contest
Honorable Mentions



1.6 mW, 56 GHz 8-bit ALU
~35 TOPS/W

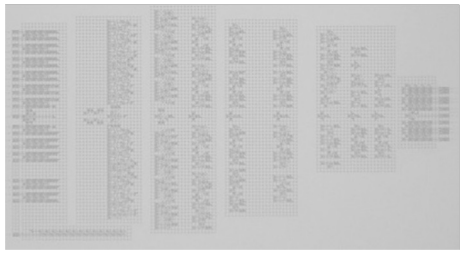
→Next design achieved
112 TOPS/W



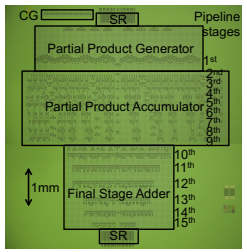
Our Approach & Outcome

Device/Circuit Optimization

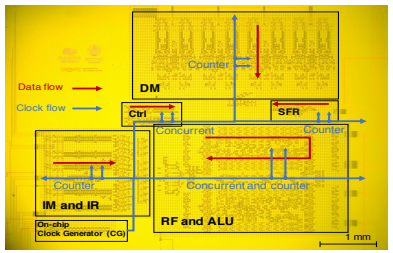
Architectural Optimization



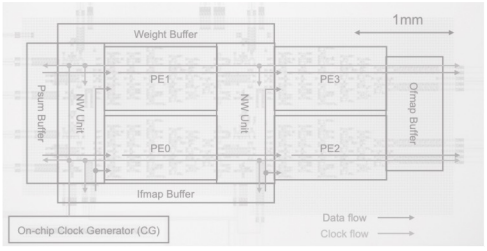
56GHz 1.6mW ALU
ISLPED'17 Design Contest
Honorable Mention



48GHz 5.6mW Multiplier
ISSCC'19
SilkRoad Award



32GHz 6.2mW Processor
VLSI Symposium'20
Selected as a featured paper

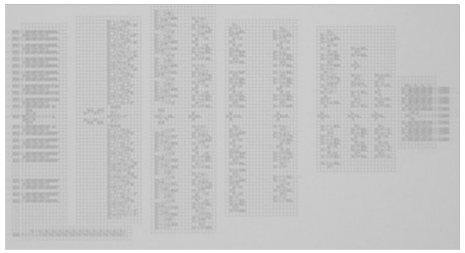


50GHz AI Accelerator
MICRO'20

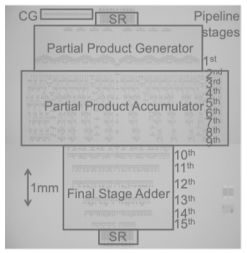
Our Approach & Outcome

Device/Circuit Optimization

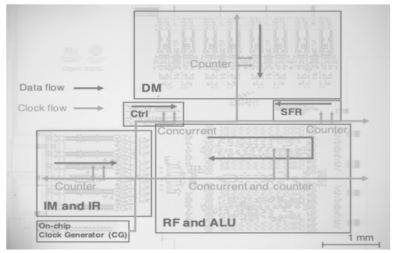
Architectural Optimization



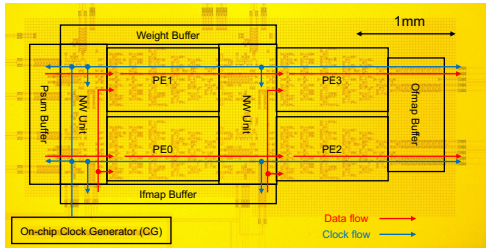
56GHz 1.6mW ALU
ISLPED'17 Design Contest
Honorable Mention



48GHz 5.6mW Multiplier
ISSCC'19
SilkRoad Award



32GHz 6.2mW Processor
VLSI Symposium'20
Selected as a featured paper



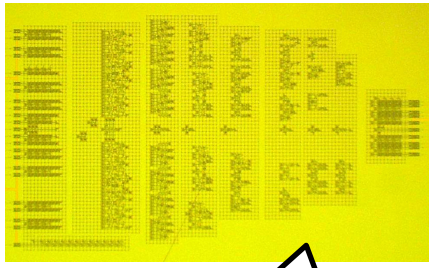
50GHz AI Accelerator
MICRO'20

Koki Ishida et al., "SuperNPU: Architecting an Extremely Fast Neural Processing Unit Using Superconducting Logic Devices," IEEE/ACM International Symposium on Microarchitecture (MICRO-53), pp. 58-72, Oct. 2020.
Koki Ishida et al., "Superconductor Computing for Neural Networks," in IEEE Micro (IEEE's Top Picks), pp.1–8, May/June 2021.

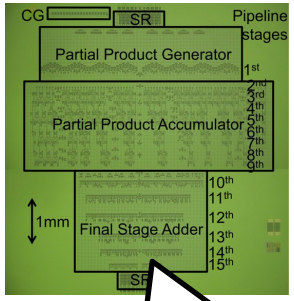
Target: "neural processing unit (NPU)"

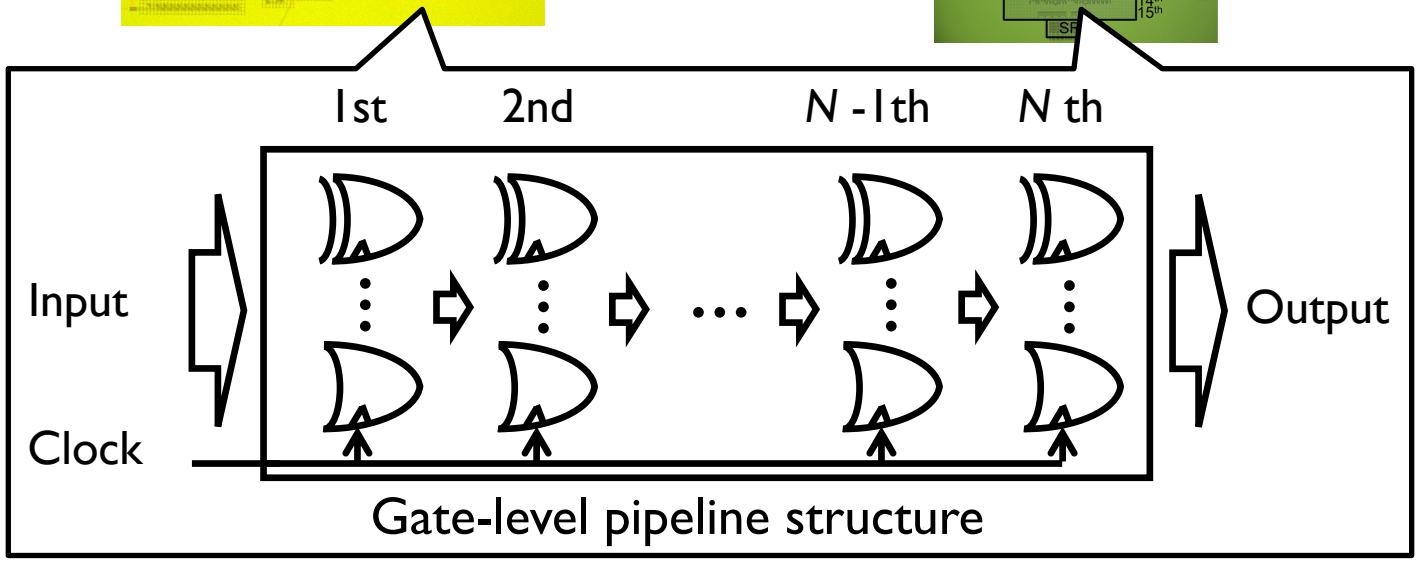
Component circuits

8bit ALU: 56 GHz, 1.6mW [1]



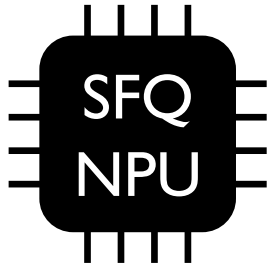
8bit MUL: 48 GHz, 5.6 mW [2]





Next: Architectural Unit

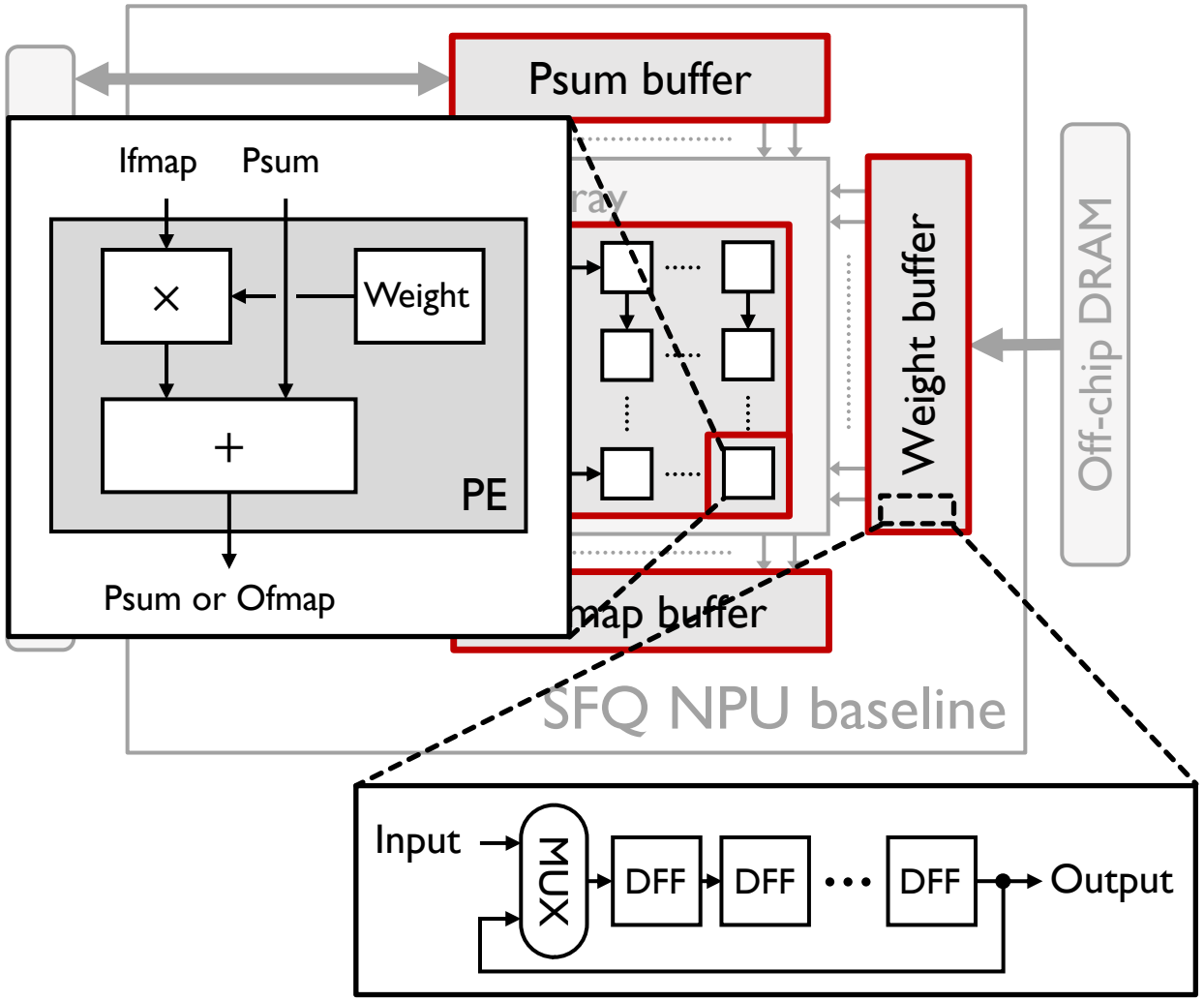
First case study: **NPU**



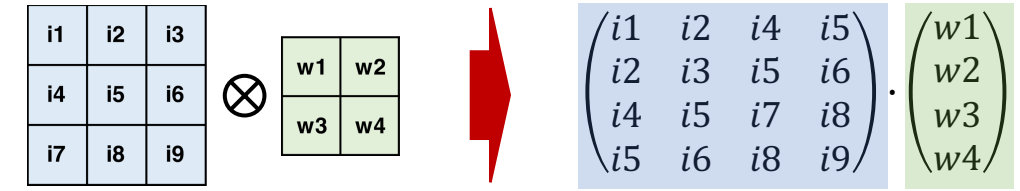
- ✓ Circuit characteristics
  - Gate-level pipelining
- ✓ High potential components
  - 8bit ALU: 56 GHz
  - 8bit MUL: 48 GHz

[1] M. Tanaka et al., "High-throughput bit-parallel arithmetic logic unit using rapid single-flux-quantum logic," in Proc. of ISEC, Jun. 2017
[2] I. Nagaoka et al., "A 48 GHz 5.6mW gate-level-pipelined multiplier using single-flux quantum logic," in ISSCC2019, 2019

Design of SFQ NPU baseline



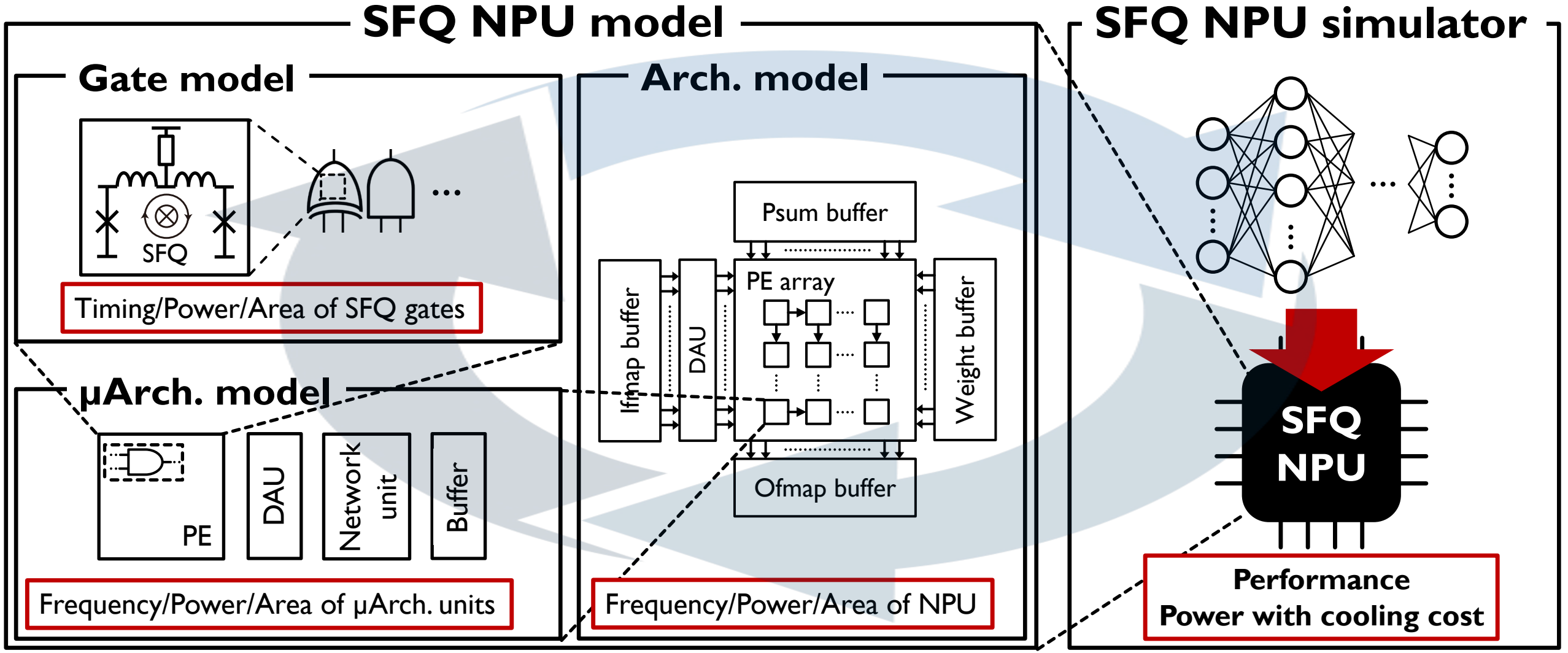
- On-chip network
  - 2D systolic array
- Buffer design
  - Shift-register-based buffers
  - Data alignment unit



Convolutional operation — Matrix multiplication

- Processing Element (PE) design
  - Weight stationary PE

Simulation framework overview

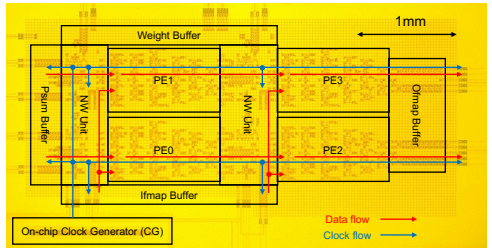


Validation

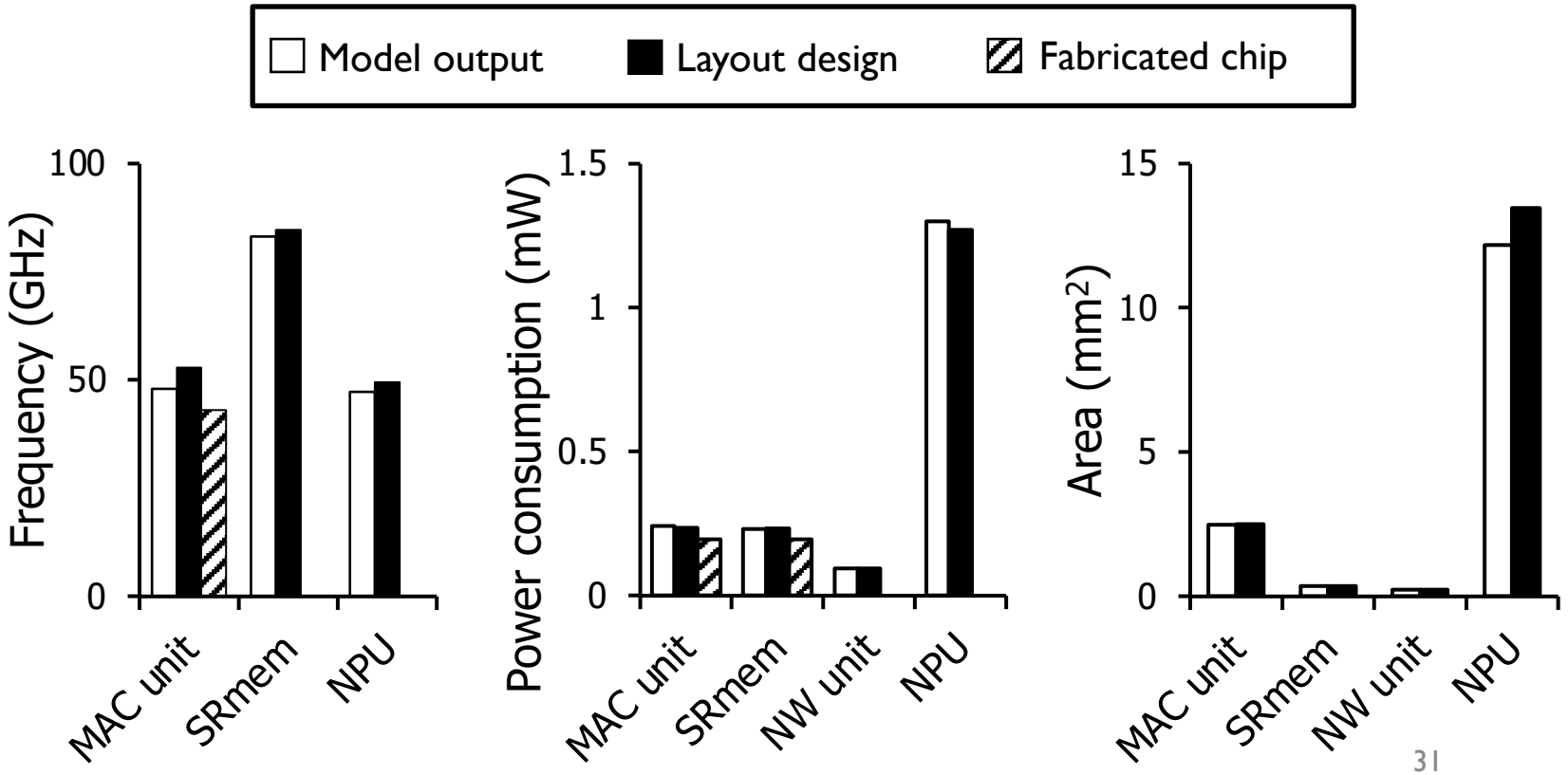
- μArch. and Arch. model accurately estimate freq./power/area
  - Compared with the post-layout simulation and chips fabricated by 1.0 μm Nb [3]

Validation setup

Chip of MAC



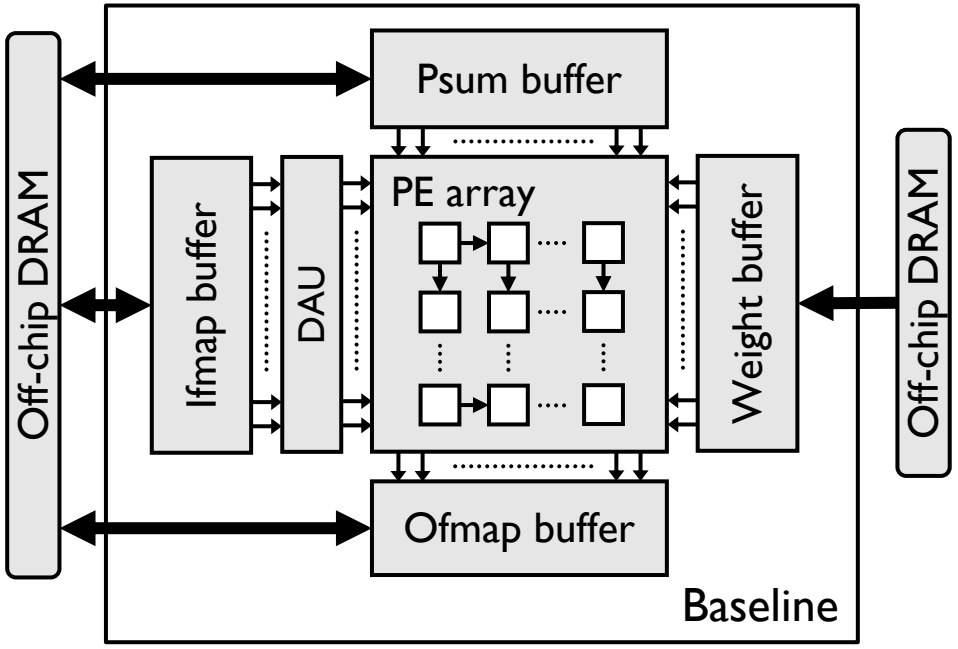
Chip of 2x2 PE array



[3] S. Nagasawa et al., "Nb 9-layer fabrication process for superconducting large-scale SFQ circuits and its process evaluation," IEICE Tran. on Electron., 2014

Architectural optimization setup (simulation)

- Identify bottlenecks of baseline SFQ NPU
  - 6 CNN workloads (AlexNet, FasterRCNN, GoogLeNet, MobileNet, ResNet50, VGG16)

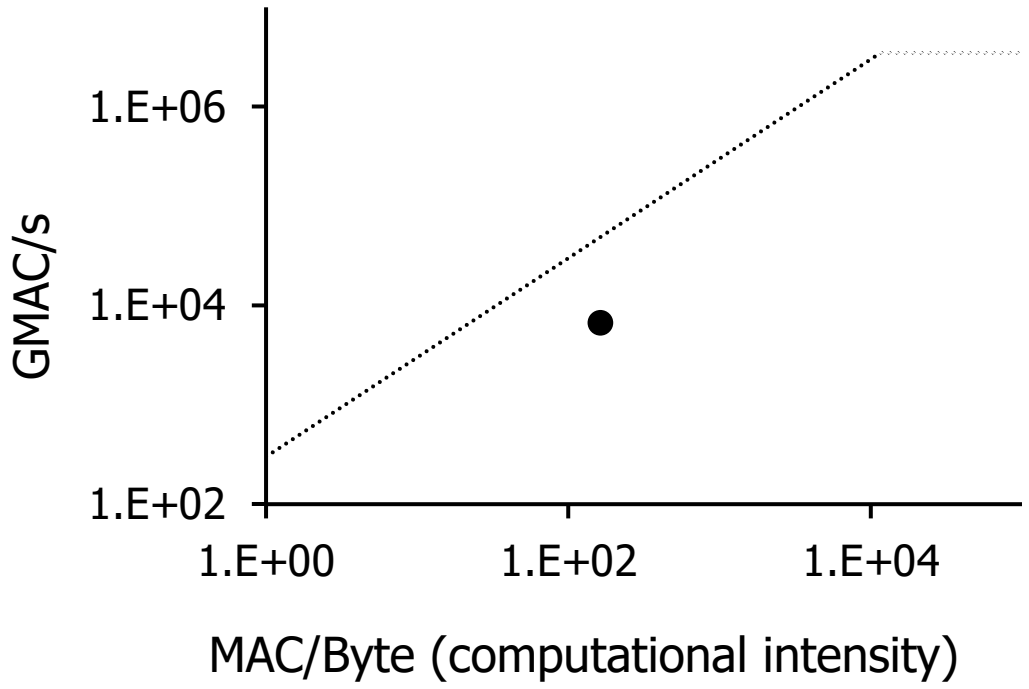


	TPU [4]	Baseline
PE array size (width x height)	256 x 256	256 x 256
Ifmap buf. size	24 MB	8 MB
Ofmap buf. size		8 MB
Psum buf. size		8 MB
Weight buf. size		64 KB
# registers in PE	1	1
Clock frequency	700 MHz	52.6 GHz (1μm Nb)
Peak performance	45 TMAC/s	3366 TMAC/s
Memory bandwidth [5]	300 GB/s	300 GB/s
Area (28 nm)	< 330 mm^2	283 mm^2

[4] N. P. Jouppi et al., "In-datacenter performance analysis of a tensor processing unit," in ISCA '17, 2017
[5] "Hot Chips 2017: A Closer Look At Google's TPU v2," https://www.tomshardware.com/news/tpu-v2-google-machine-learning,35370.html

Architectural optimization setup (simulation)

- Identify bottlenecks of baseline SFQ NPU
  - 6 CNN workloads (AlexNet, FasterRCNN, GoogLeNet, MobileNet, ResNet50, VGG16)

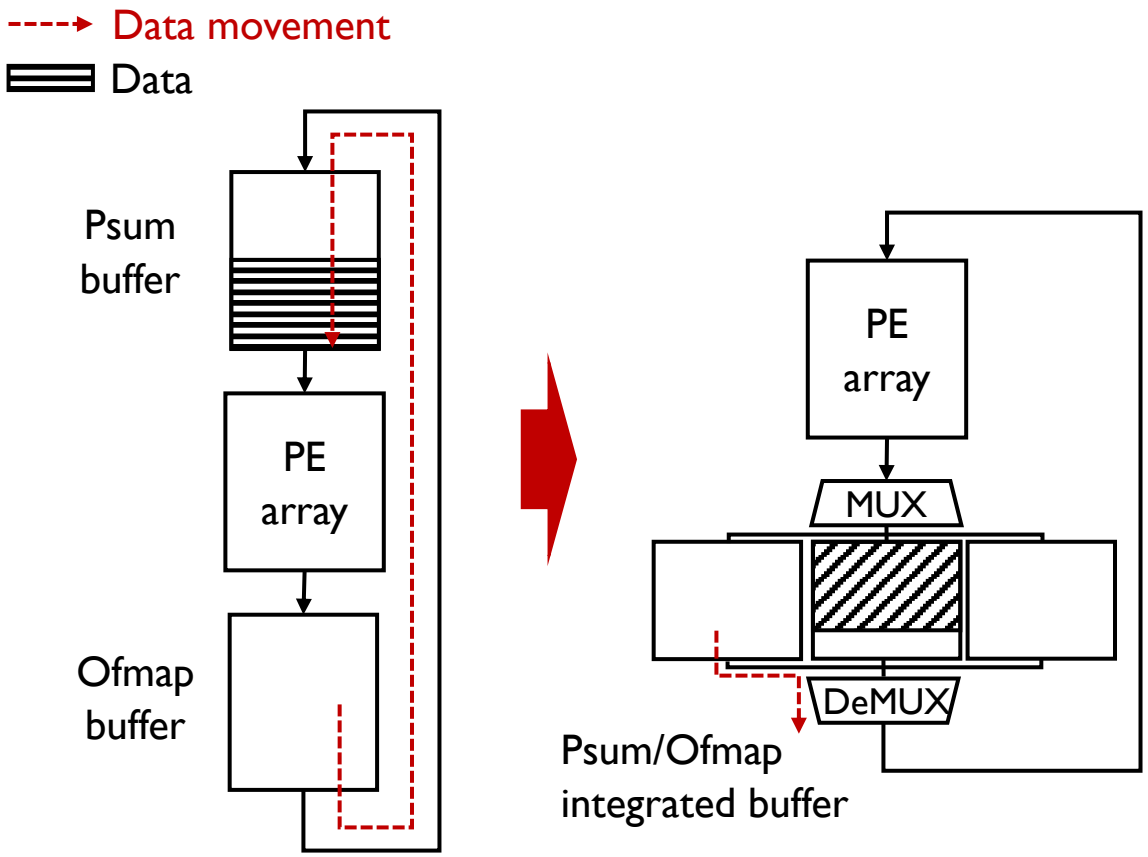


	TPU [4]	Baseline
PE array size (width x height)	256 x 256	256 x 256
Ifmap buf. size	24 MB	8 MB
Ofmap buf. size		8 MB
Psum buf. size		8 MB
Weight buf. size		64 KB
# registers in PE	1	1
Clock frequency	700 MHz	52.6 GHz (1μm Nb)
Peak performance	45 TMAC/s	3366 TMAC/s
Memory bandwidth [5]	300 GB/s	300 GB/s
Area (28 nm)	< 330 mm^2	283 mm^2

[4] N. P. Jouppi et al., "In-datacenter performance analysis of a tensor processing unit," in ISCA '17, 2017
[5] "Hot Chips 2017: A Closer Look At Google's TPU v2," https://www.tomshardware.com/news/tpu-v2-google-machine-learning,35370.html

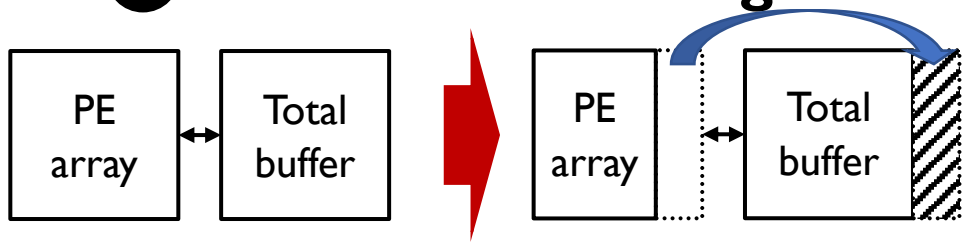
Architectural optimization

❶ Buffer division



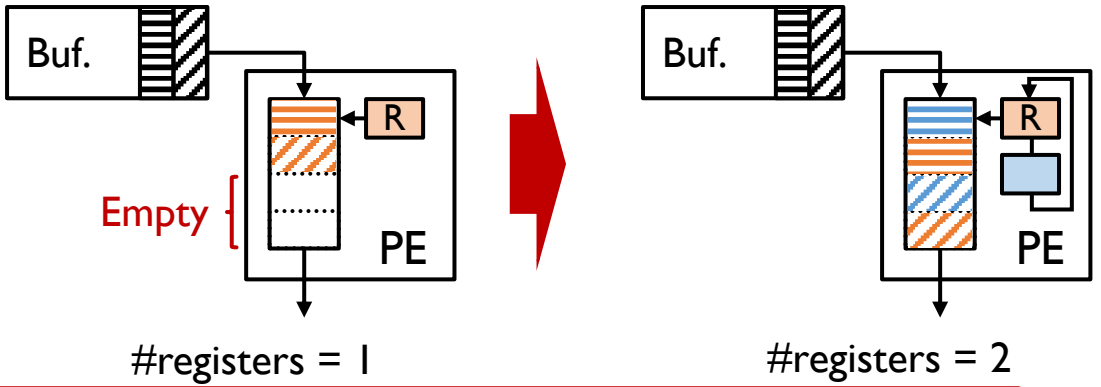
Data movement cycles are reduced!

❷ Resource balancing



Increasing buffer size with PE reduction can improve computational intensity!

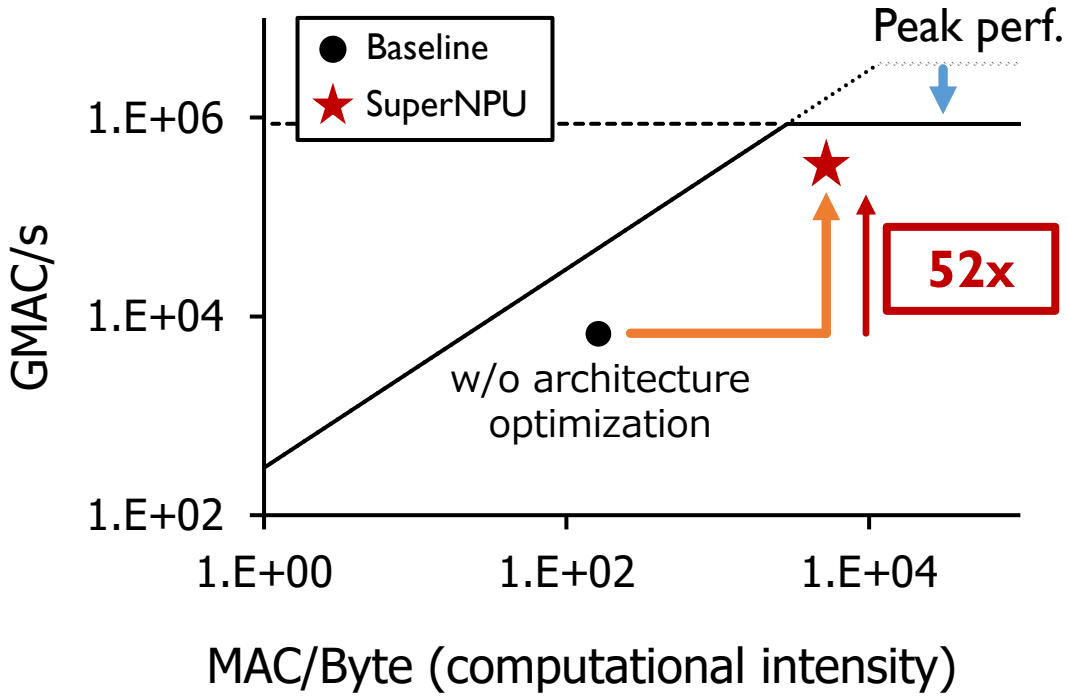
❸ Increase #registers in PE

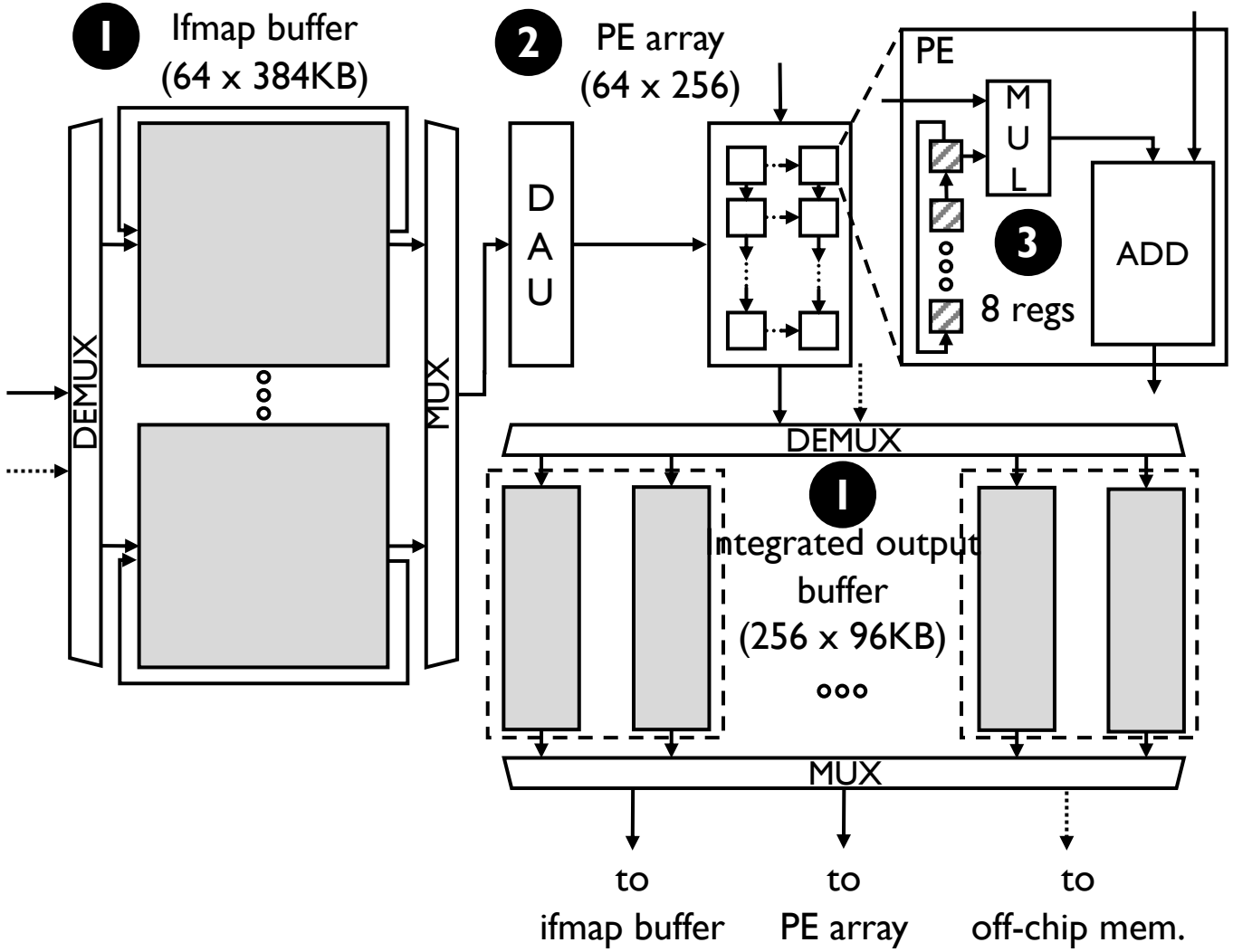


Multiple registers can fill deep pipeline!

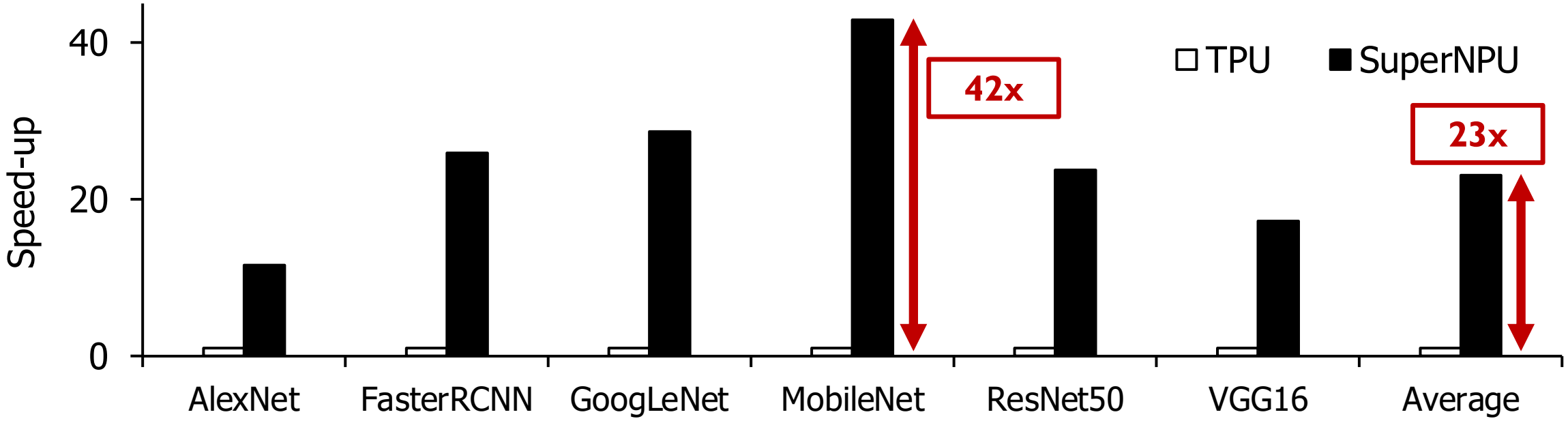
SuperNPU: optimized SFQ NPU architecture

1 Buffer division
2 Recourse balancing
3 Increase #registers in PE





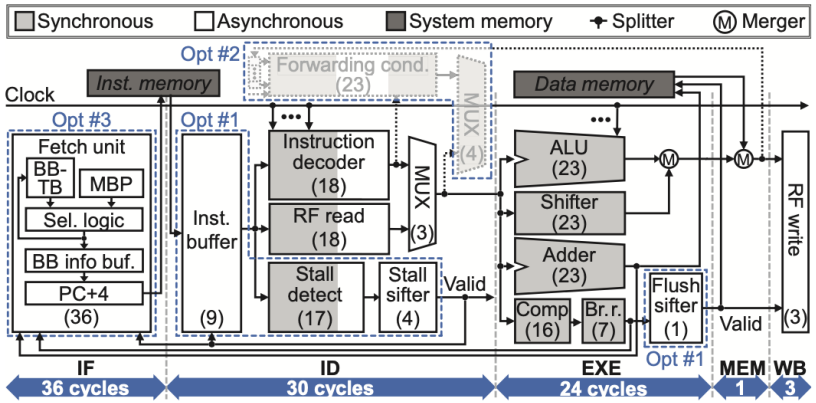
Performance evaluation

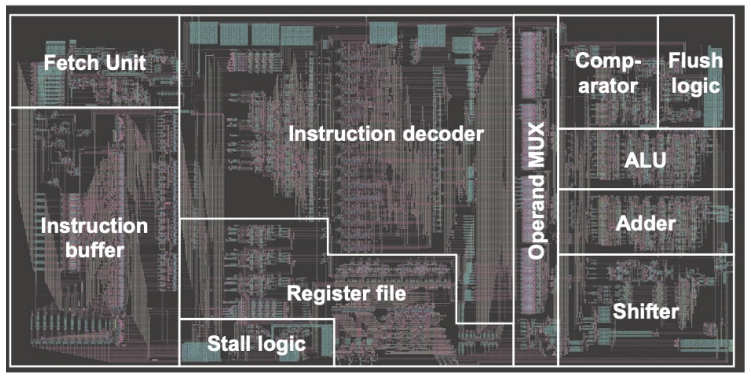


- SuperNPU greatly improves the performance
  - **23x** of speed-up on average, up to **42x** on MobileNet
  - Thanks to high **SFQ potential (high frequency)** and **architectural optimization**

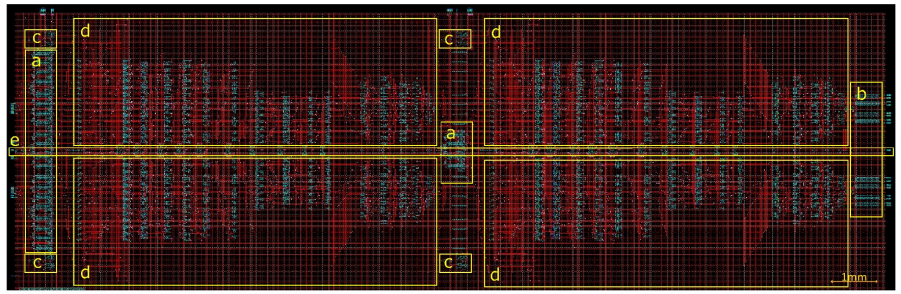
State-of-the-art SFQ Researches

SuperCore + SuperSFQ, MICRO'24, MICRO'25

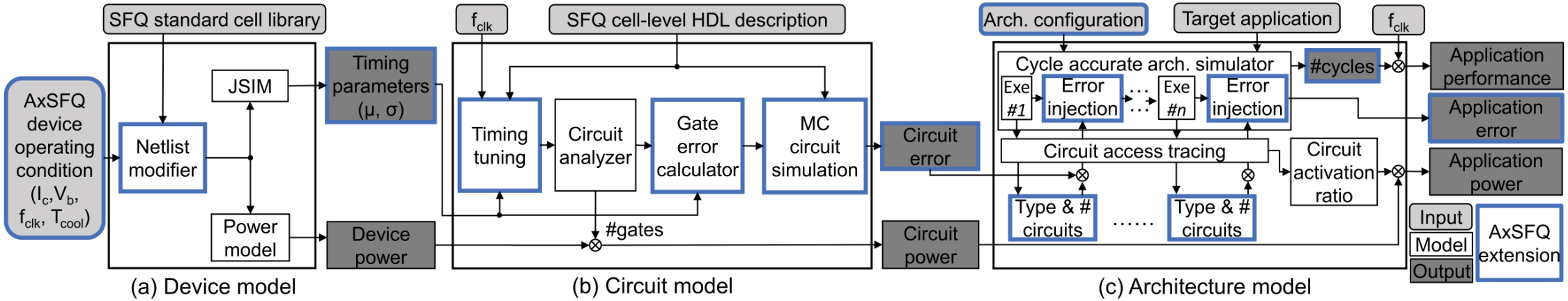




100 GHz SuperNPU 2x2 PE array (to be published, IPSJ Journal)



Approximate SFQ Computing (CAL'24)

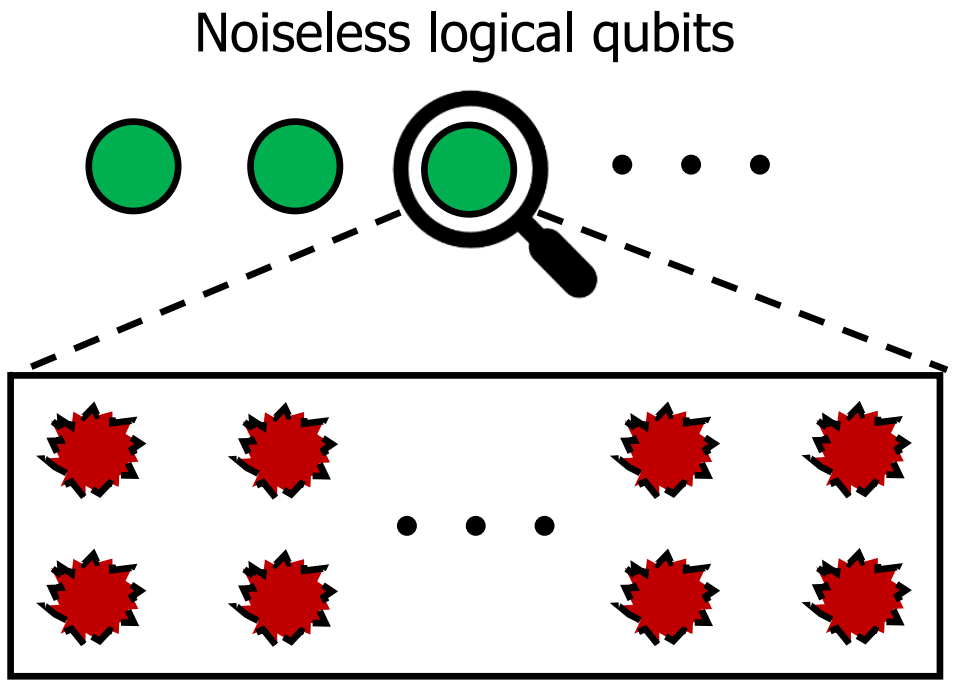


From Classical to Quantum!
~Our Next Step~

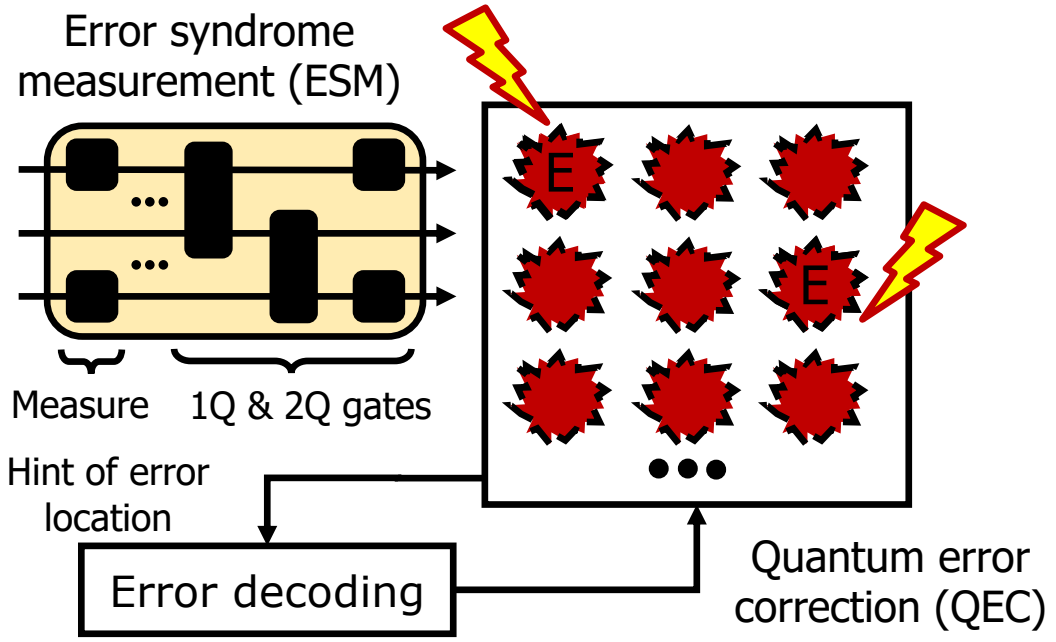
Fault-tolerant quantum computing (FTQC)

- FTQC resolves the error problem of quantum computers!

<Concept of FTQC>

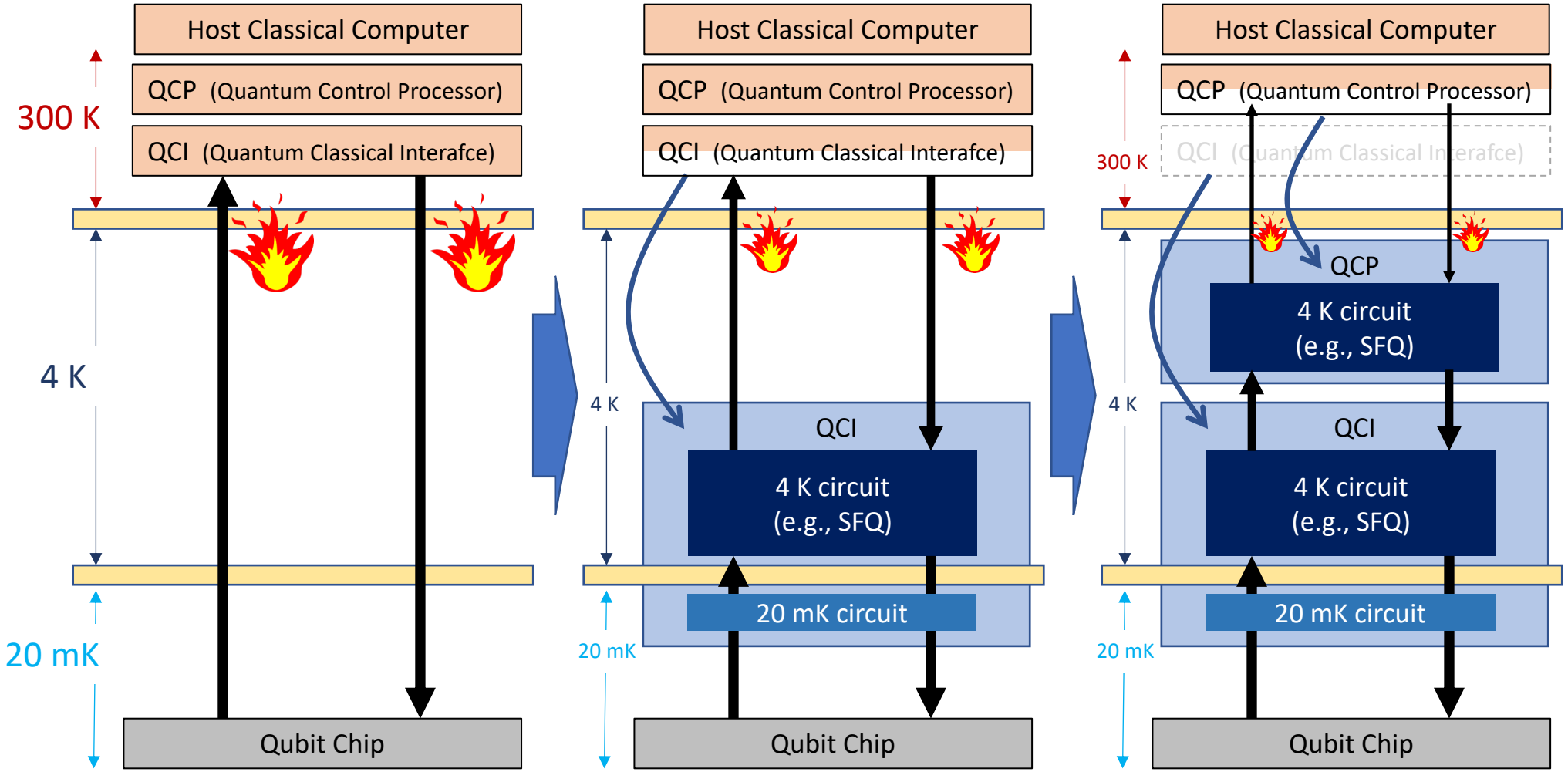


<Details of FTQC>



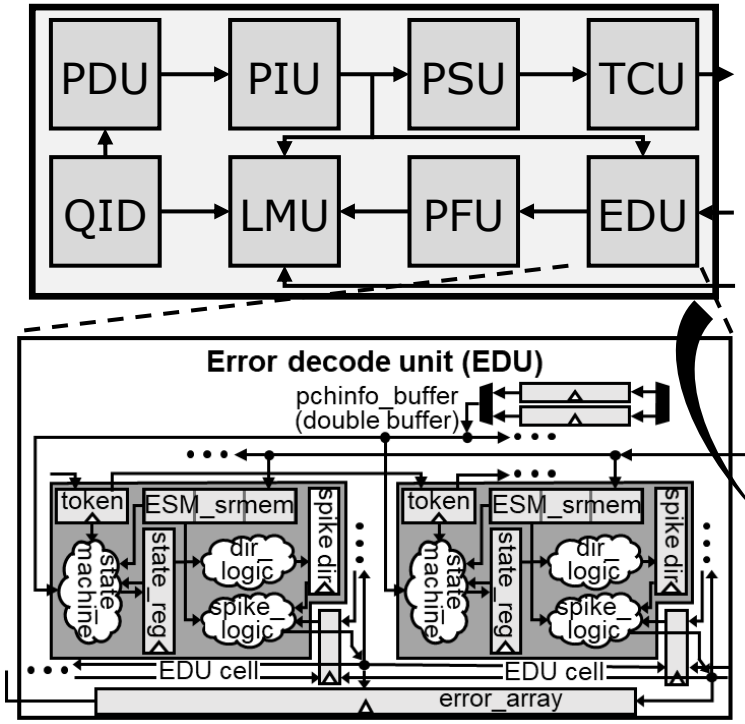
With FTQC, we can greatly reduce the overall errors (e.g., $>10^{10}$ times)

How can SFQ Technology Contribute to Superconducting Quantum Computers?



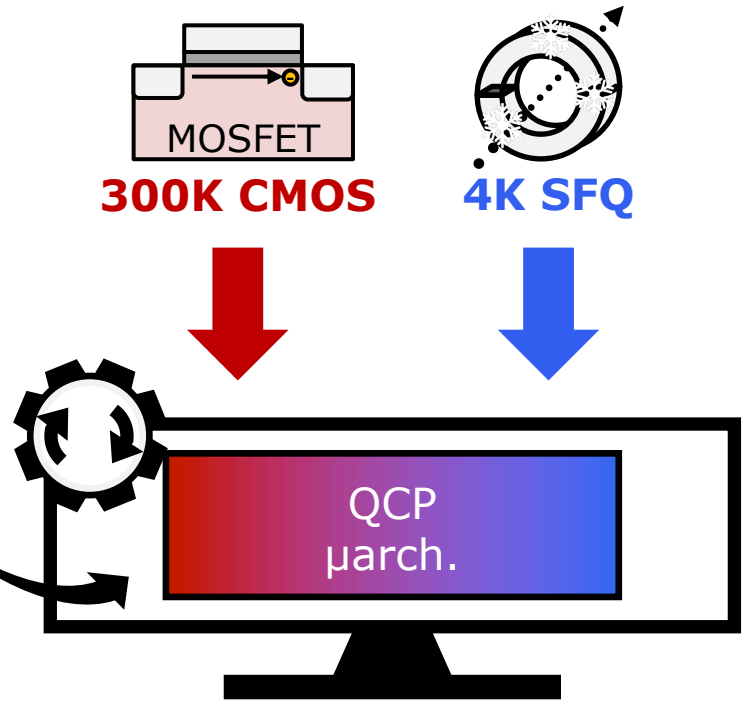
XQsim: Research Overview

Full QCP μarchitecture



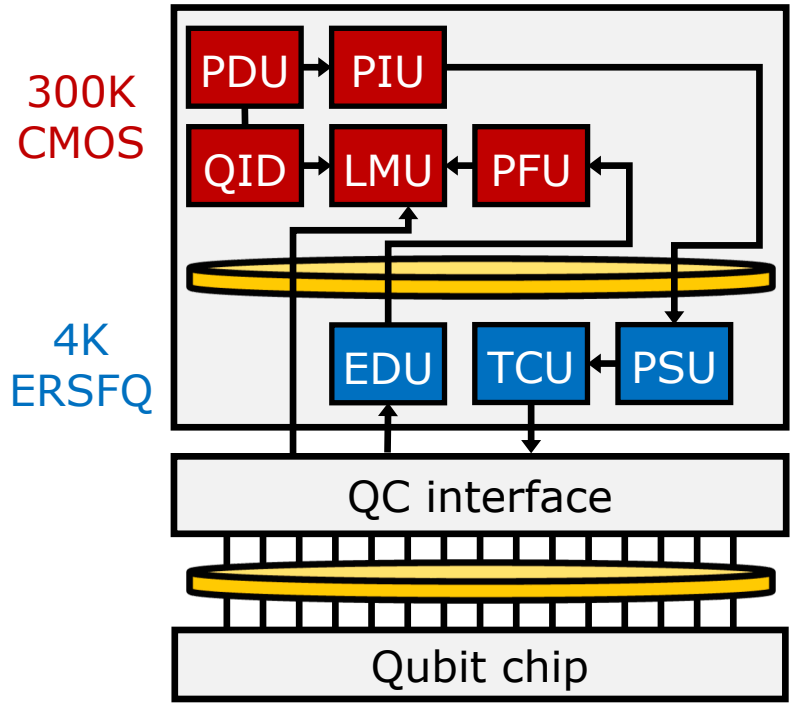
Detailed RTL impl. & validation

QCP modeling tool



Cross-technology modeling & simulation

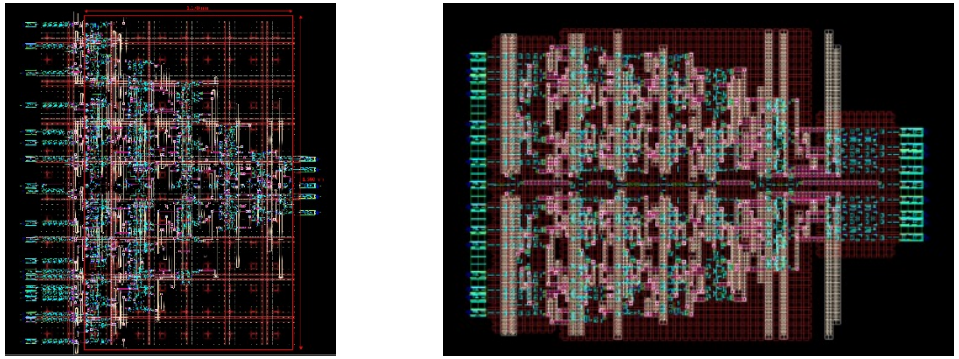
10+K qubit QCP arch.



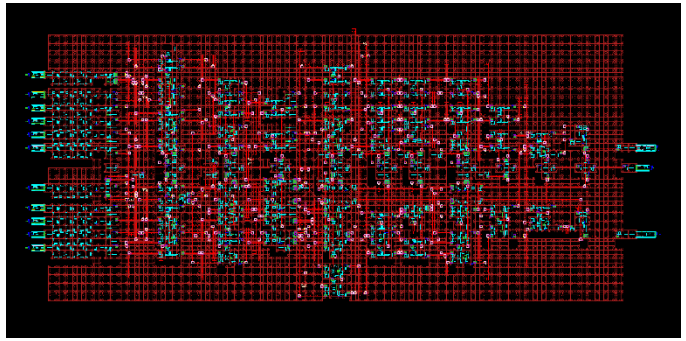
Temp. & Tech. & Arch. optimizations

XQ-estimator: Validation

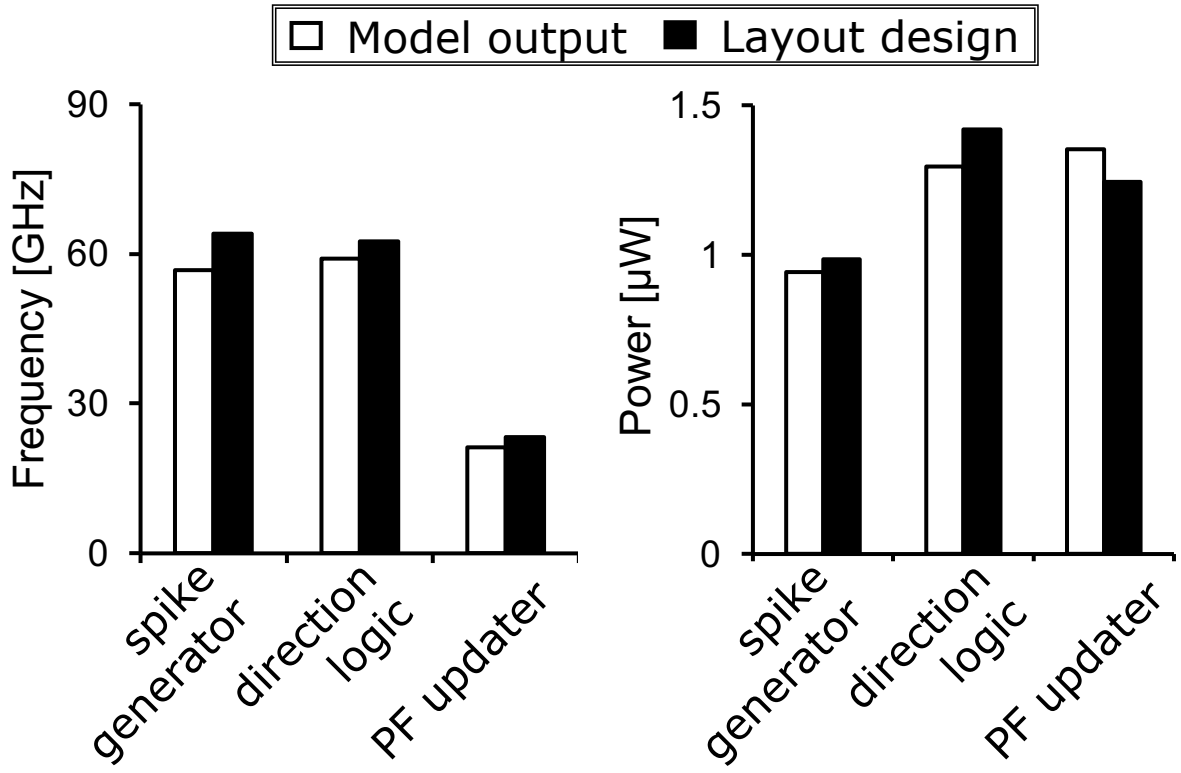
- **SFQ model accurately predicts the frequency and power**
  - Compared with the post-layout analysis using AIST 1.0μm process library
  - Validated with the circuits in various QCP units (e.g., EDU, PFU)



Layout of spike generator and direction logic inside EDU

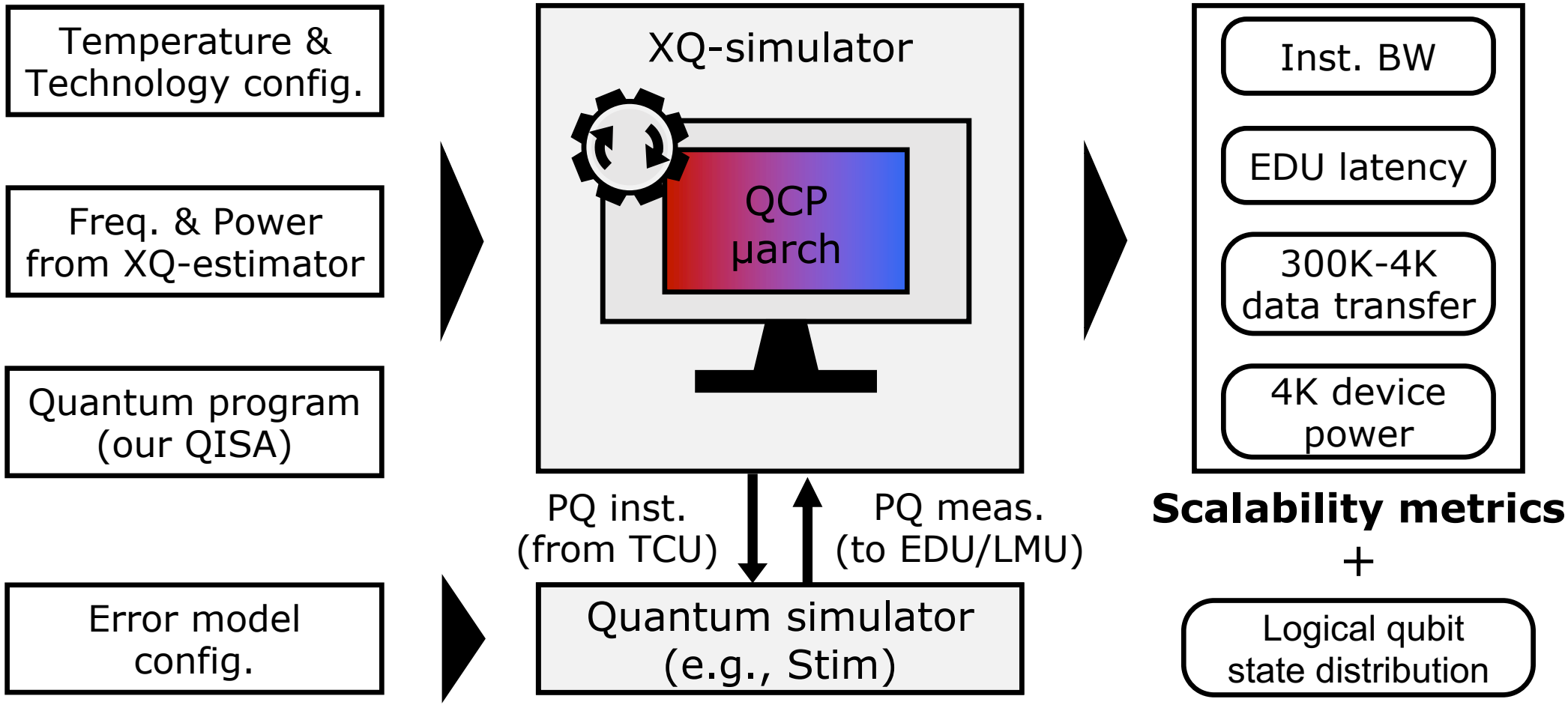


Layout of PF updater inside PFU

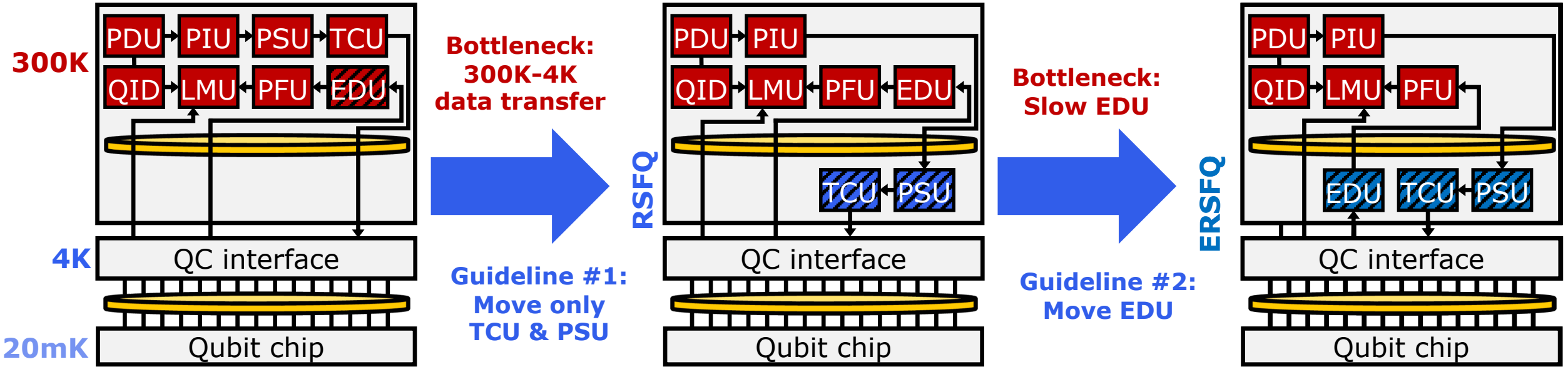


XQ-simulator: Overview

- **Run simulation to report scalability metrics and manageable qubit scale**
- **Integrate a quantum simulator for the functionally correct simulation**



10+K qubit QCP design: Summary



Bottleneck: Slow EDU

μarch #1: Fast EDU

Qubit scale: < 250 → 1,700

Bottleneck: 4K device power

μarch #2 & #3: Low-power PSU & TCU

Qubit scale: 970 → 4,600

Bottleneck: 4K device power

μarch #4: Low-power EDU

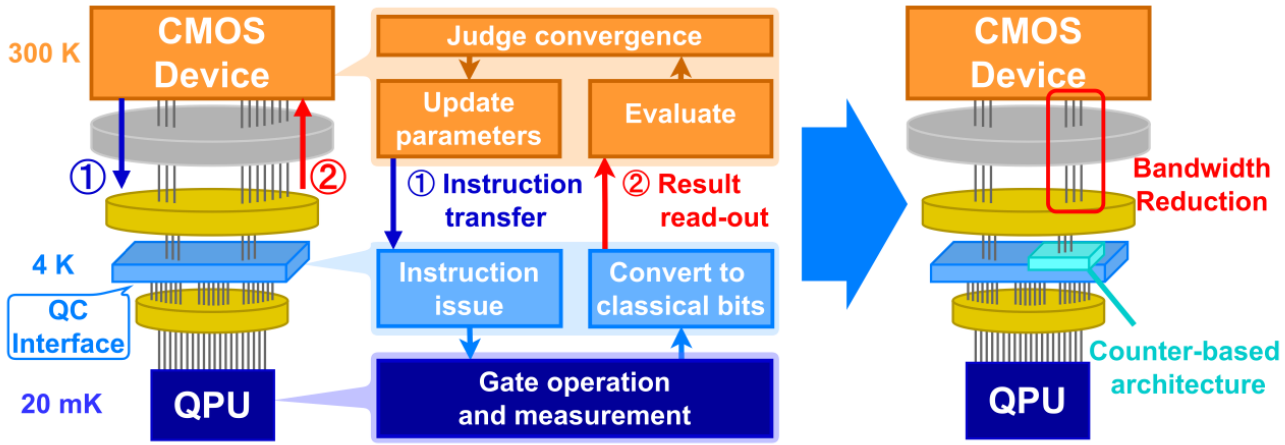
Qubit scale: 8,100 → 59,000

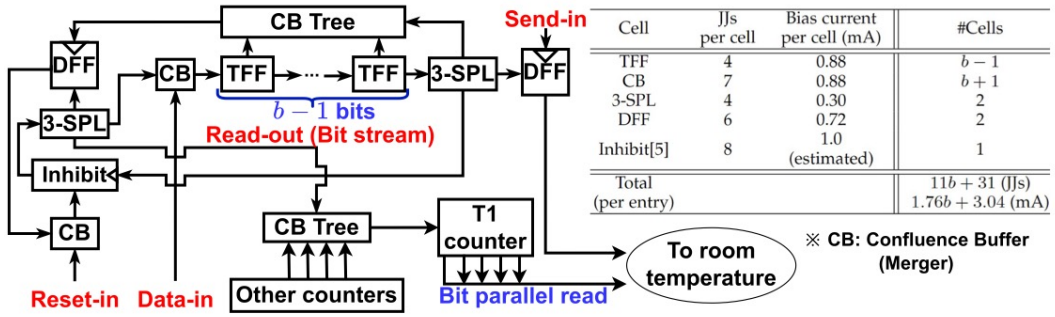
With thorough analyses using XQsim, we could provide directions for designing a 10+K qubit QCP using SFQ technology!

IEEE-CSC, ESAS and CSSJ SUPERCONDUCTIVITY NEWS FORUM (global edition), Issue No. 62, Jan. 2026. Presentation given at QUEST 2025, Yokohama, Japan, Oct. 2025.

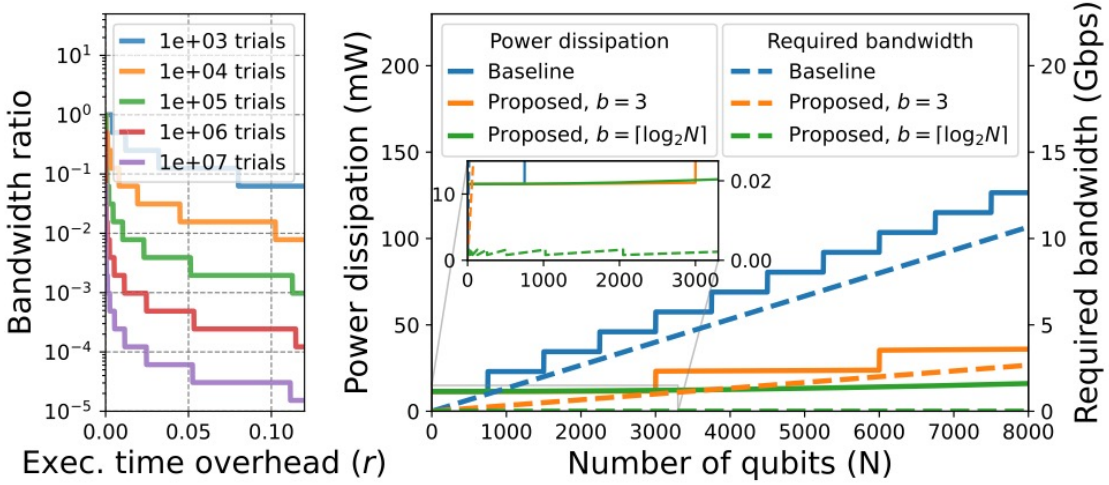
Yosuke Ueno et al., "*C3-VQA: Cryogenic Counter-Based Co-Processor for Variational Quantum Algorithms,*" IEEE Transactions on Quantum Engineering, Vol. 6, pp.1-17, Dec. 2024.
Yosuke Ueno et al., "*Inter-Temperature Bandwidth Reduction in Cryogenic QAOA Machines,*" IEEE Computer Architecture Letters, Vol. 23, No. 1, pp. 9-12, Jan.-June 2024.

System Level Architecture Optimization





Cell	JJs per cell	Bias current per cell (mA)	#Cells
TFF	4	0.88	$b-1$
CB	7	0.88	$b+1$
3-SPL	4	0.30	2
DFF	6	0.72	2
Inhibit[5]	8	1.0 (estimated)	1
Total (per entry)			$11b+31$ (JJs) $1.76b+3.04$ (mA)

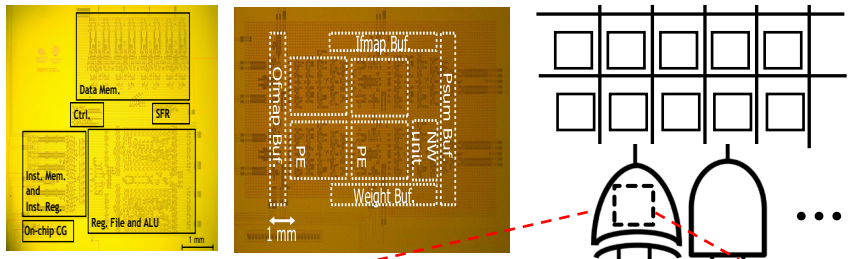


- Superconducting quantum computer requires many inter-temperature cables
  - Hardware complexity, heat inflow, peripheral power, etc.
- For QAOA, qubit measurement readout communication is the dominant
- Counter-based SFQ architecture reduces meas. Bandwidth

Message

What We Need?

Logic/Circuit Level Modeling



Device Interface (Device Spec.: Fixed)

Many constraints, such as performance, cost, power, reliability, etc.

Device-Level Exploration

Limitation

Architecture exploration under GIVEN devices

CS: Computer Sciense

Design Wall

DS: Device Sciense

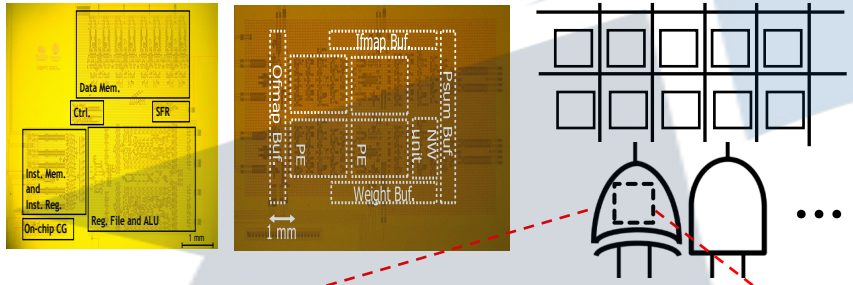
Limitation

Devices have to satisfy ALL constraints (or requirements)

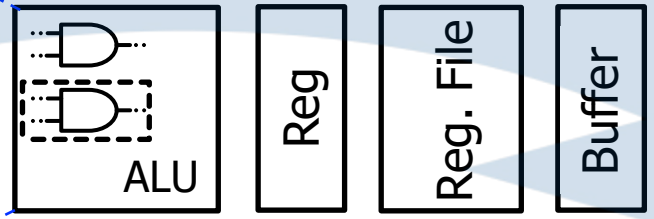
CS×DS Design Cycle!

~System Design Methodology to Exploit Device Diversity~

Logic/Circuit Level Modeling

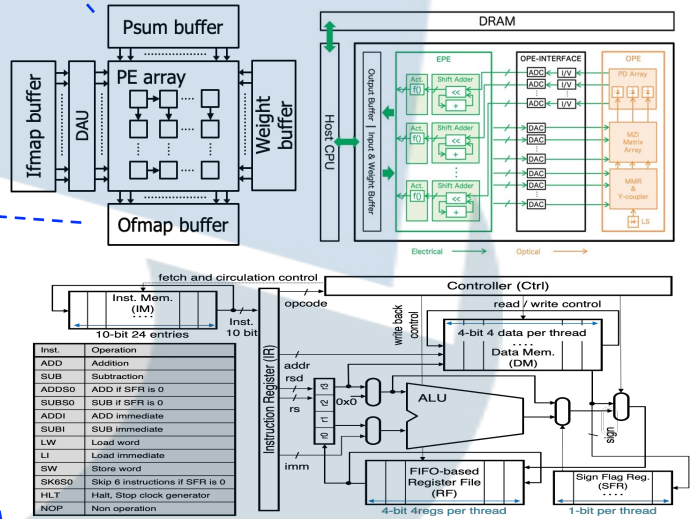


µArch. Level Modeling

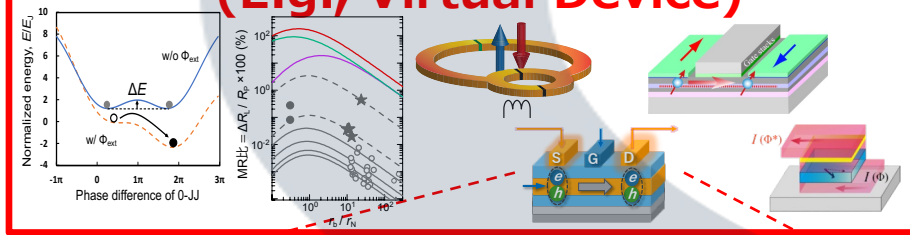


Arch. Level Modeling

Simulation/Emulation

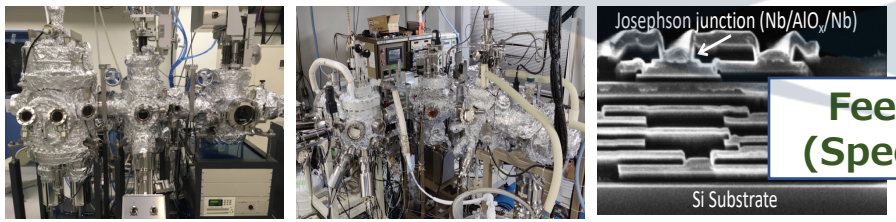


Device Modeling (E.g., Virtual Device)



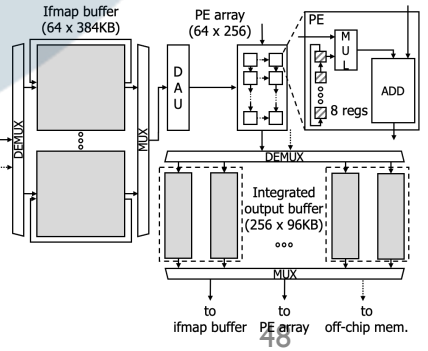
System/Application Level Evaluation

Device-Level Exploration

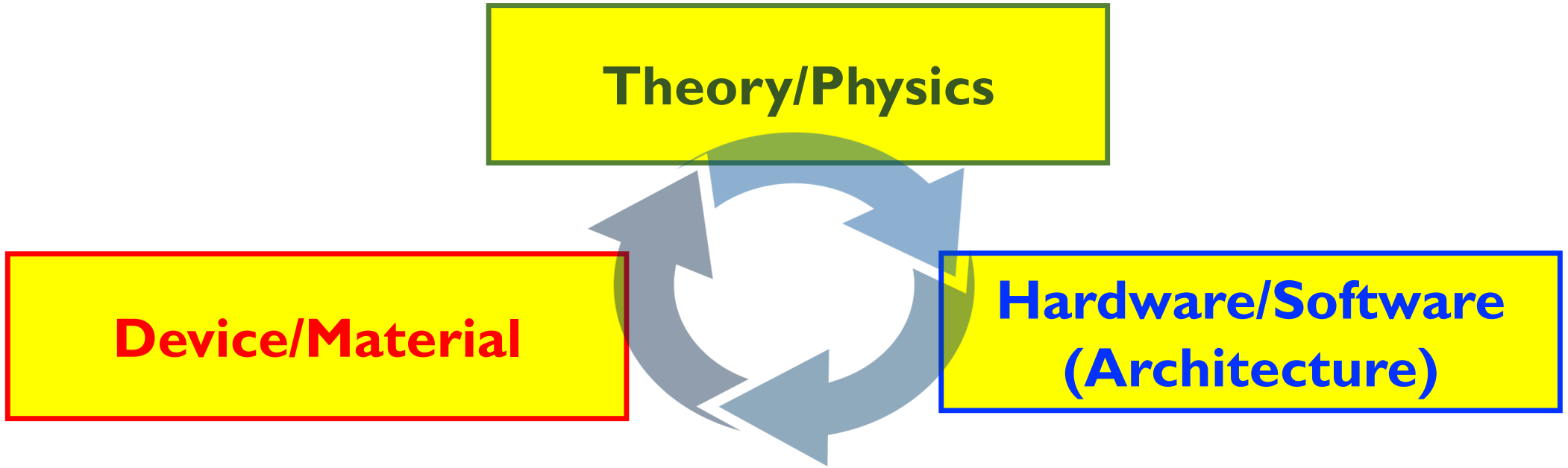


Feedback (Spec., etc.)

- Programming
- Algorithm
- OS
- Compiler
- µArchitecture



Need A-Z Co-Design for Emerging Device Computing!



Closs-layer interaction is required for next generation computing with emerging devices!

Acknowledgments

This work was supported by JST-Mirai Program Grant Number JPMJMI18E1, JSPS KAKENHI Grant Numbers JP19H01105, JP18H05211, JP18J21274, JP22H05000, JST Moonshot R&D Grant Number JPMJMS2067. The circuit is designed with the support by VDEC of the University of Tokyo in collaboration with Cadence Design Systems, Inc., and fabricated in the CRAVITY of AIST. We also appreciate the support from National Research Foundation of Korea (NRF) grant funded by the Korean Government (NRF-2019R1A5A1027055, NRF-2020M3H6A1084857).