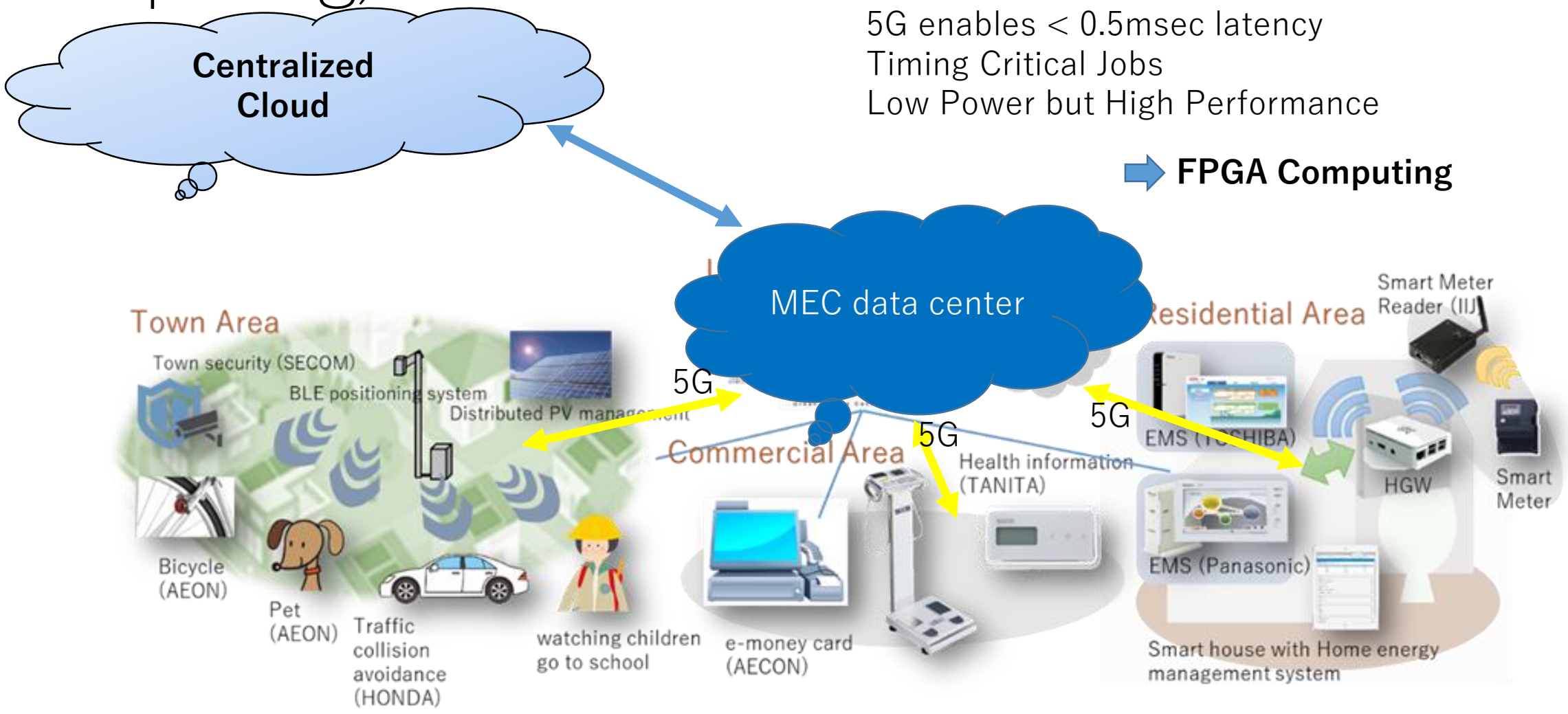
AI processing at MEC(Multi-access Edge Computing)

5G enables < 0.5msec latency
Timing Critical Jobs
Low Power but High Performance

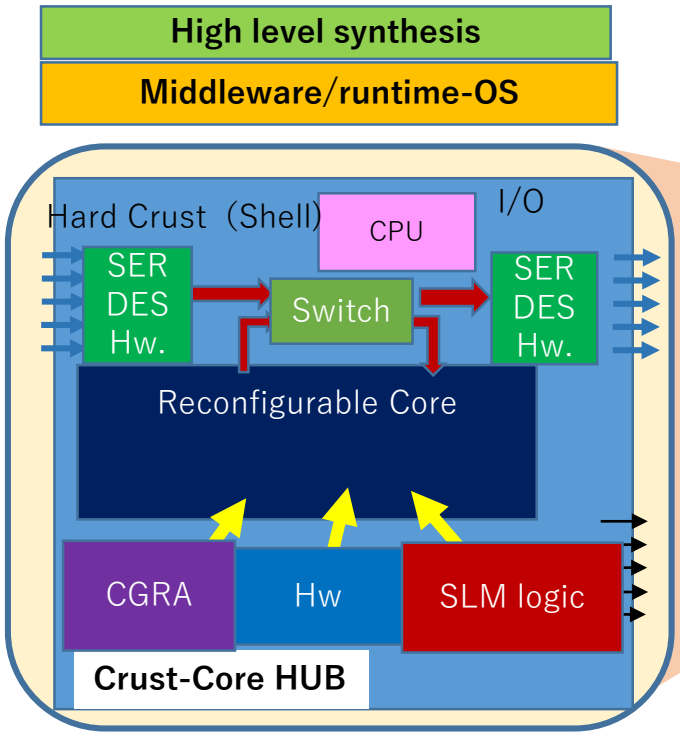
FPGA Computing

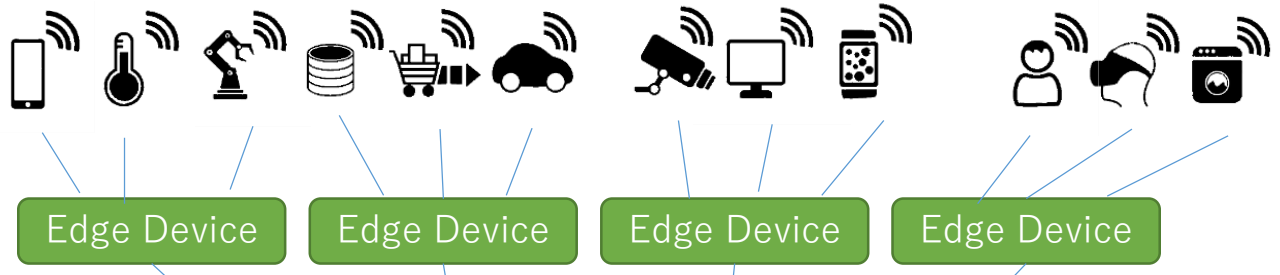


CREST Project: Computing basis for Society5.0 General Manager: Prof.Sakai

- An integrated multi-node system for MEC (PJ leader: Amano) :
  - 300million yen for 5.5 years (2019.10-2025.3)
  - Group 1: Hardware/Architecture
    - Amano (Keio U.), Kudoh (U.of Tokyo), Namiki (TAT)
  - Group 2: Reconfigurable Logic
    - Iida, Kuga, Amagasaki (Kumamoto U.), Zhou (KIT)
  - Group 3: Runtime software
    - Sugaya (Shibaura Tech.), Ohkawa(U. of Tokai), Takano(AIST)
  - Group 4: Application
    - Nishi (Keio U.), Nakadai(TIT), Inagaki(Alps),
  - Group 5: CAD
    - Wakabayashi, Takahashi , et. al(NEC)

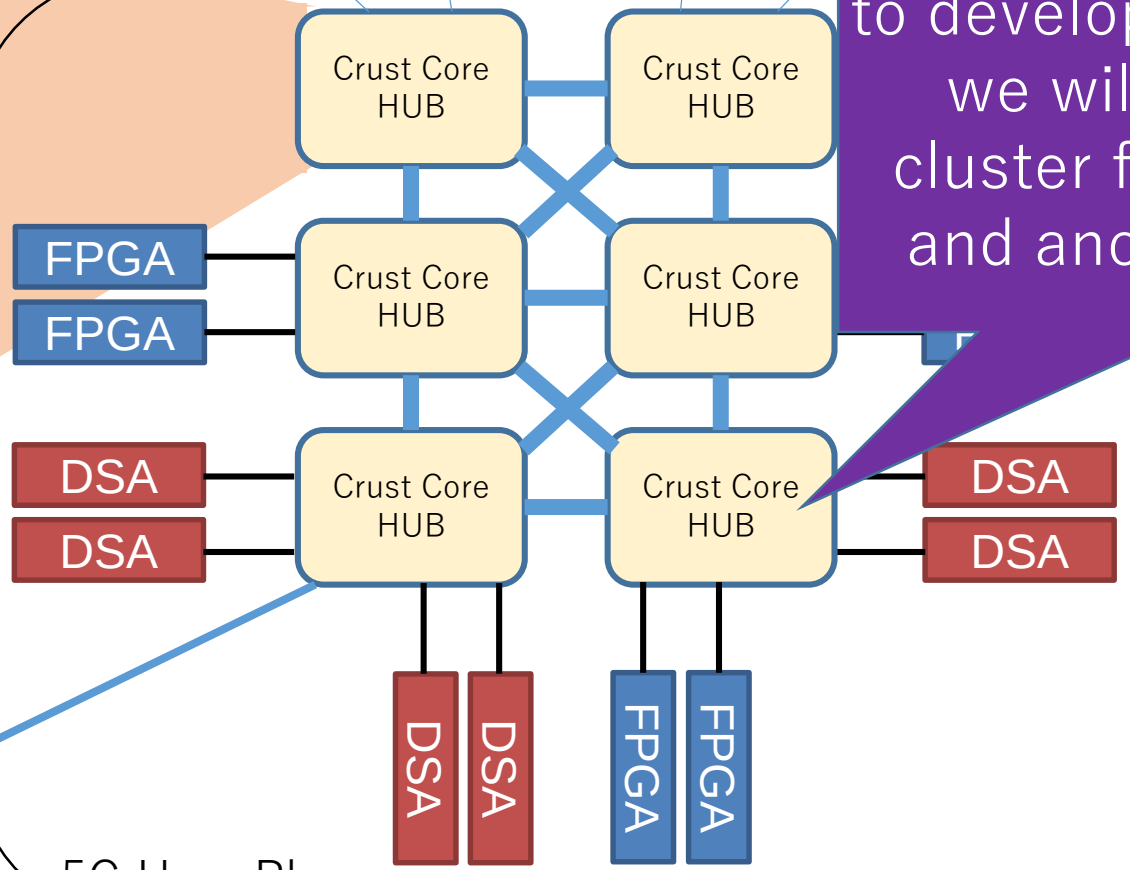
Final goal





5G Radio

Although the final goal is to develop the original chip, we will use an FPGA cluster for the prototype and another possibility.



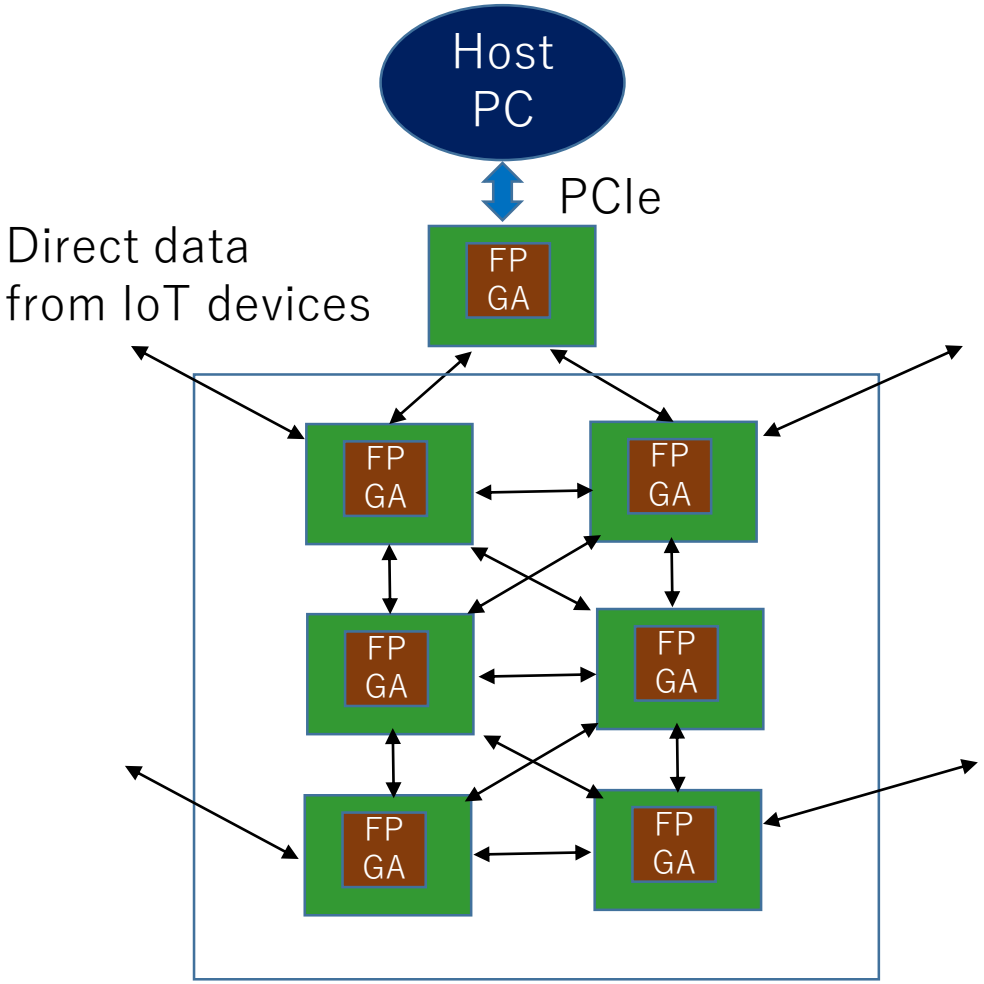
Cloud

Multi-Access Edge

FPGA computing in MEC data center

- Benefits:
  - Suitable to timing critical applications.
    - Direct data input/output from/to IoT devices
    - The constant execution time with the hardware engine
  - Low energy computation.
    - Much energy efficient than GPUs
  - Scalability for multiple access.
    - Easy to increase the number of FPGAs
    - An FPGA itself can be used as a switch.
- Programming environment has been improved:
  - Open-CL is widespread for computational usage.
  - Vivado-HLS is popularly used for general usage.
- High performance FPGAs are available.
  - Intel's Stratix 10 with a lot of floating DSPs
  - Xilinx's Ultrascale+ with UltraRAM

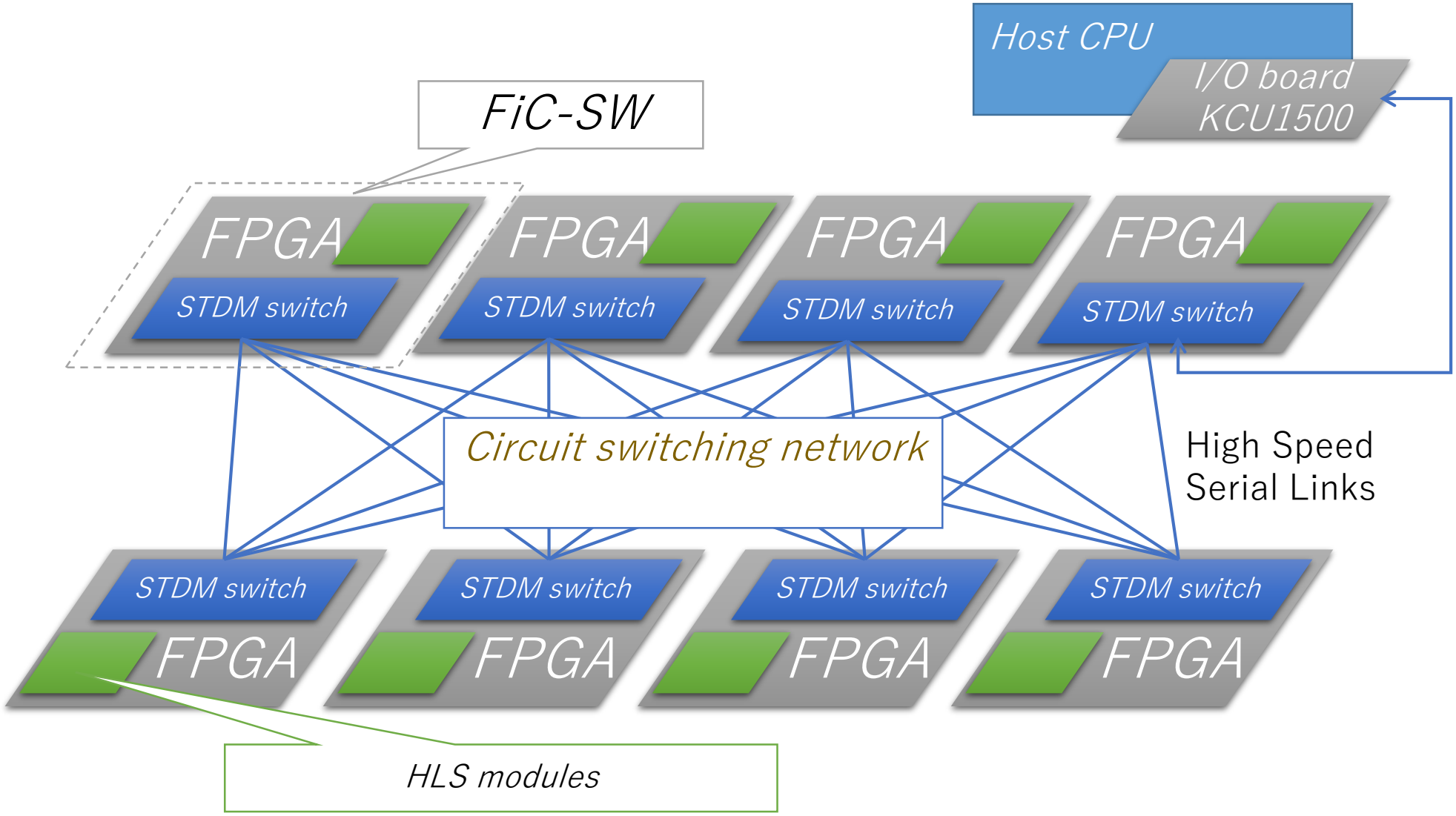
Our Proposal : A virtual FPGA system



A lot of cost-efficient middle-scale FPGAs are tightly connected.
They can be treated as if they were a single FPGA in HLS description level.
The data can be directly input/output to/from serial links
The latency and bandwidth are guaranteed.

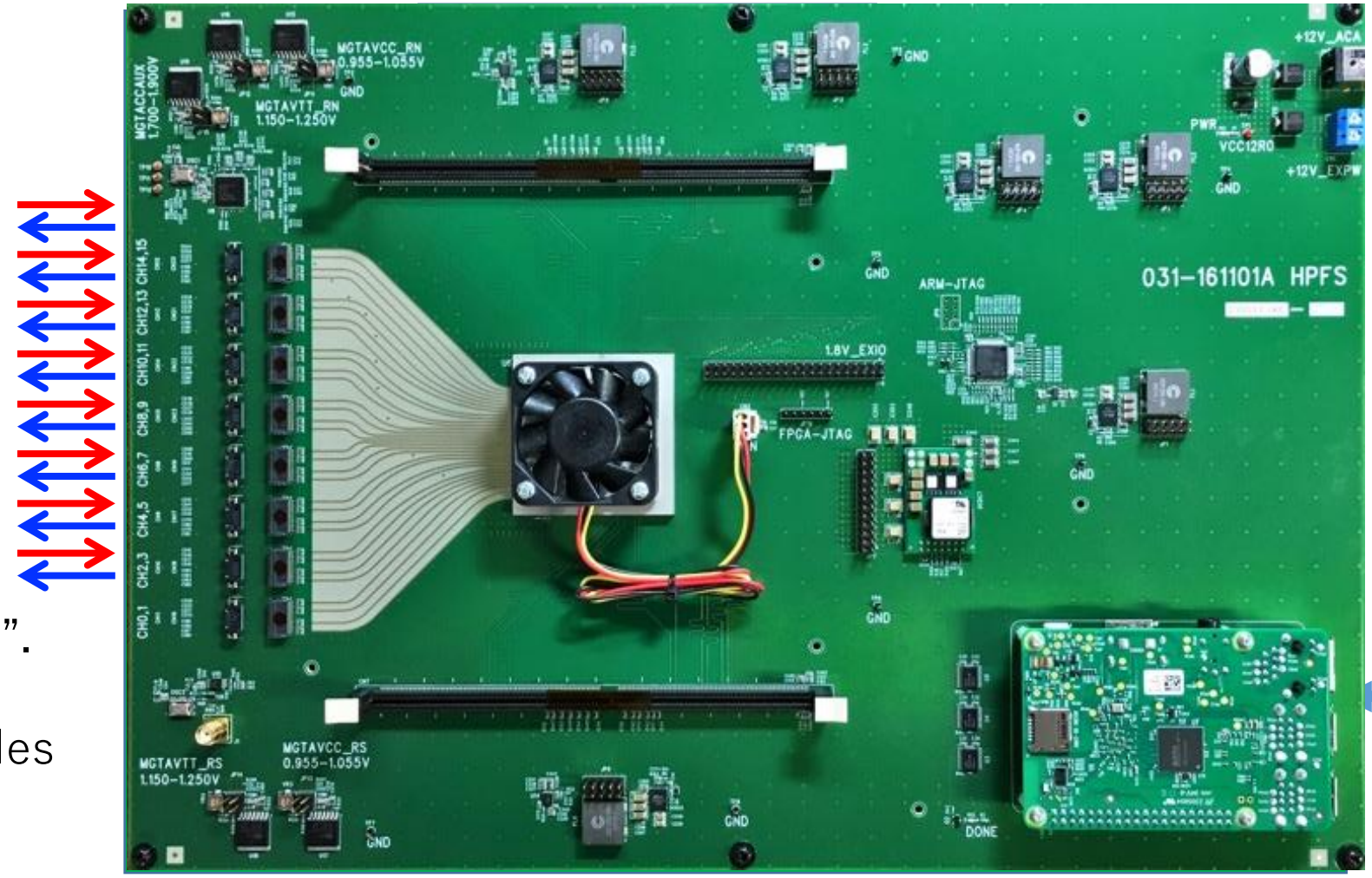
Higher performance per cost than conventional FPGA in cloud.
Practically infinite resource is used.
Separated into a number of virtual FPGAs and shared by the multiple accesses.

Flow-in-Cloud (FiC) overview



Flow-in-Cloud (FiC) SW Board

FiC Network
8x4 9.9Gbps



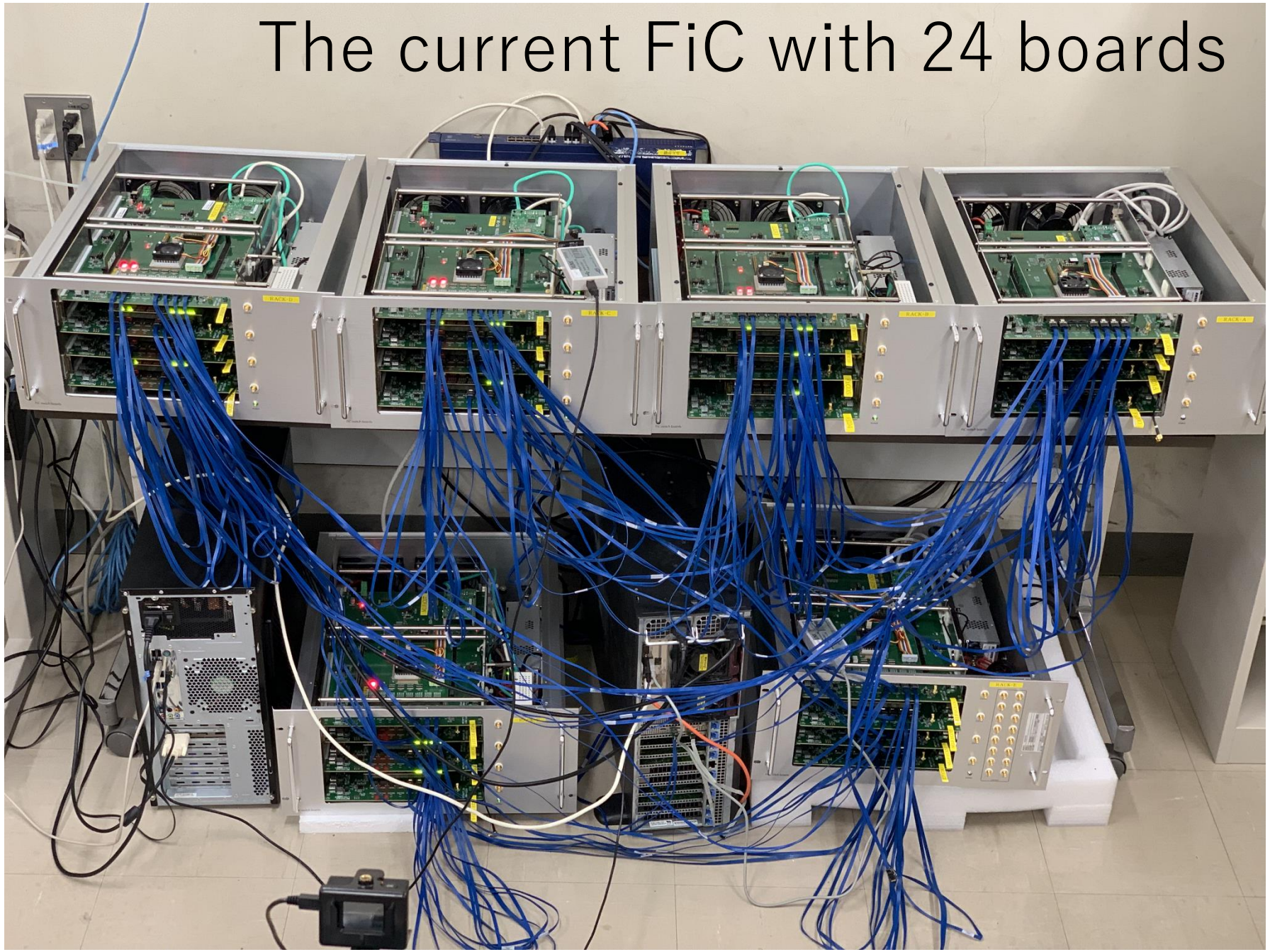
Here, we call each
link “channel”,
and a bundle of
4 channels “bundle”.

A board has 8 bundles
each of which has
4 channels

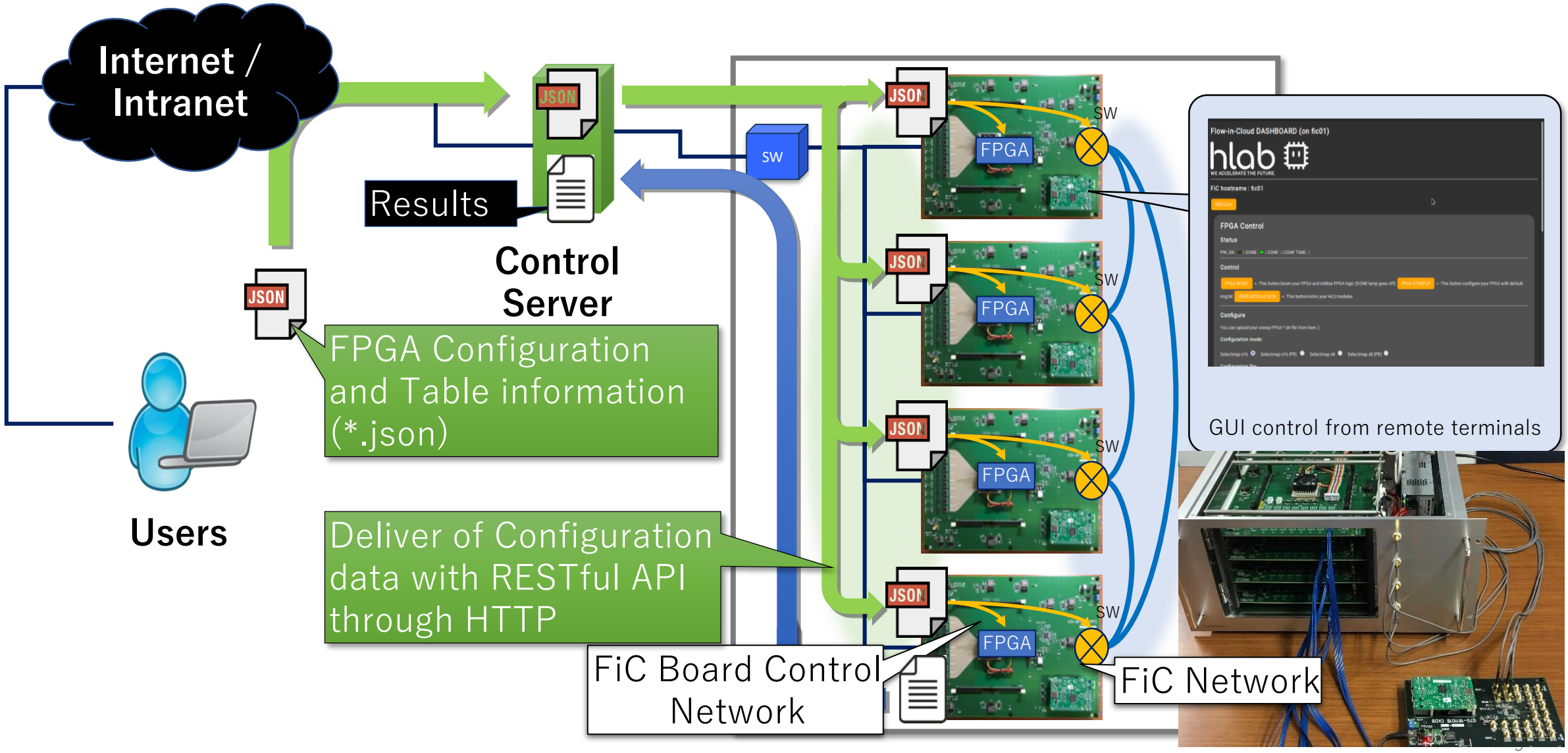
Ethernet

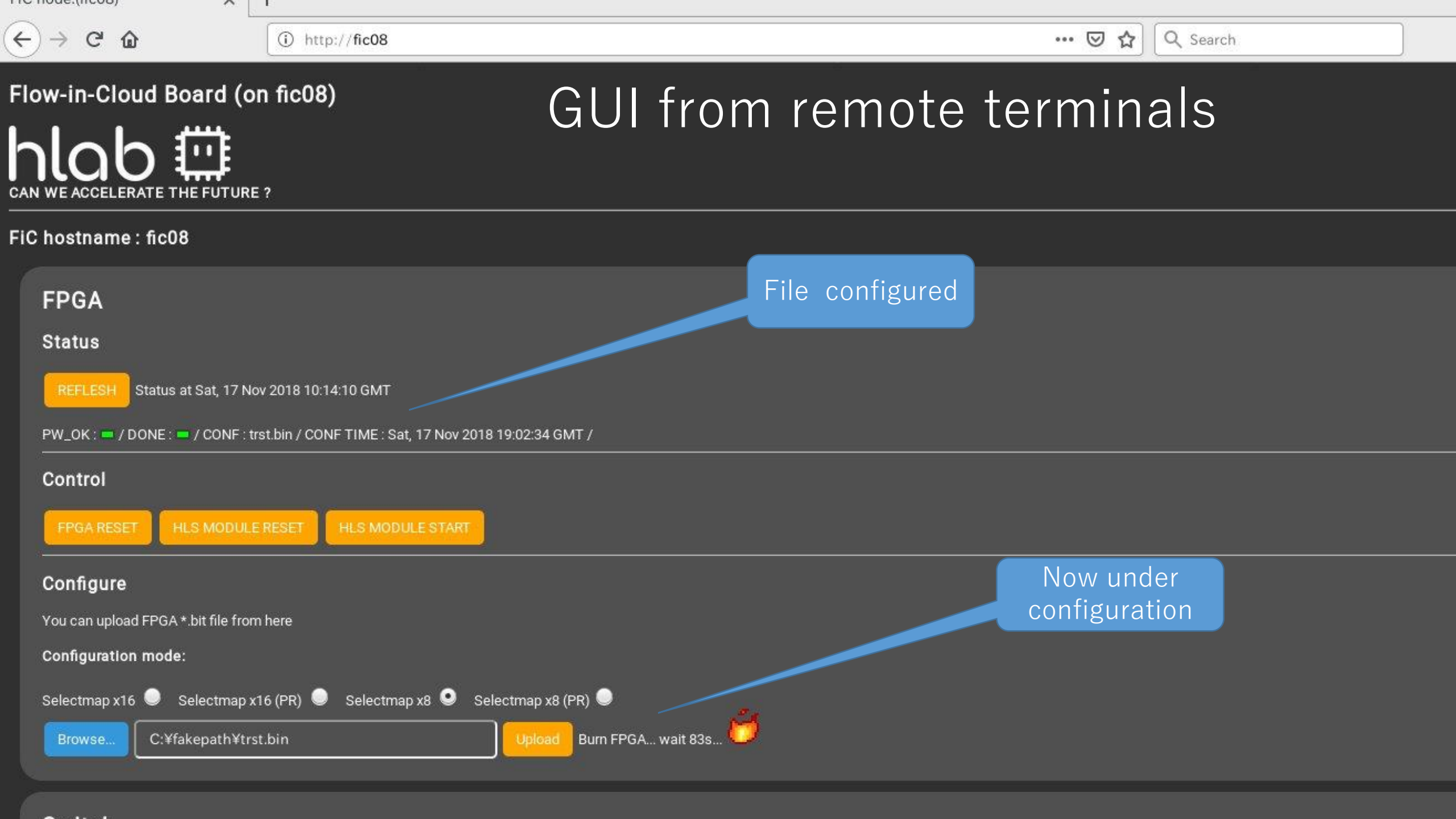
Control Network

The current FiC with 24 boards



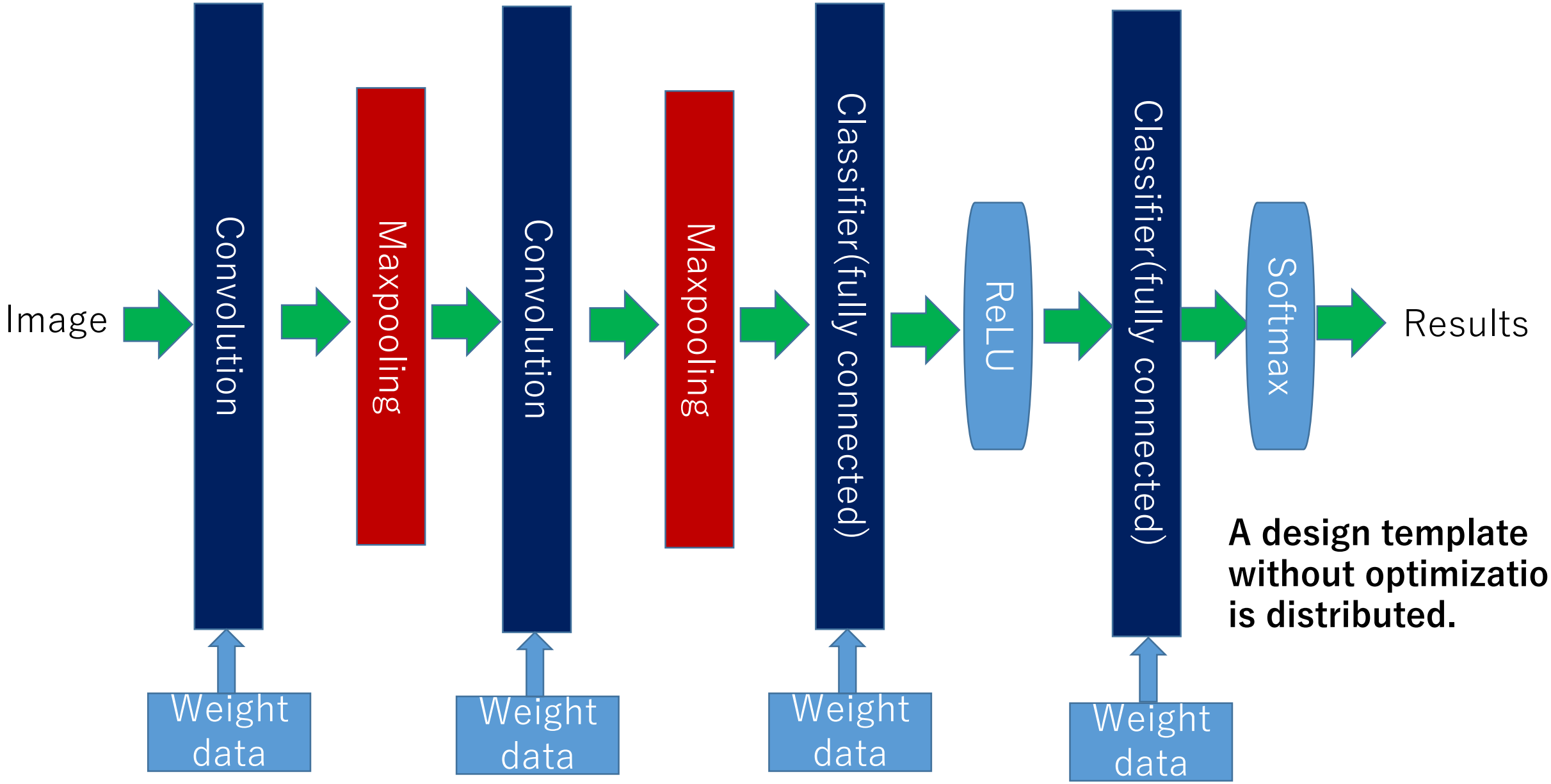
The current FiC system




http://fic08
Search
Flow-in-Cloud Board (on fic08)
hlab
CAN WE ACCELERATE THE FUTURE ?
GUI from remote terminals
FiC hostname : fic08
FPGA
Status
REFLESH
Status at Sat, 17 Nov 2018 10:14:10 GMT
PW_OK : / DONE : / CONF : trst.bin / CONF TIME : Sat, 17 Nov 2018 19:02:34 GMT /
File configured
Control
FPGA RESET
HLS MODULE RESET
HLS MODULE START
Configure
You can upload FPGA *.bit file from here
Configuration mode:
Selectmap x16
Selectmap x16 (PR)
Selectmap x8
Selectmap x8 (PR)
Browse...
C:¥fakepath¥trst.bin
Upload
Burn FPGA... wait 83s...
Now under configuration

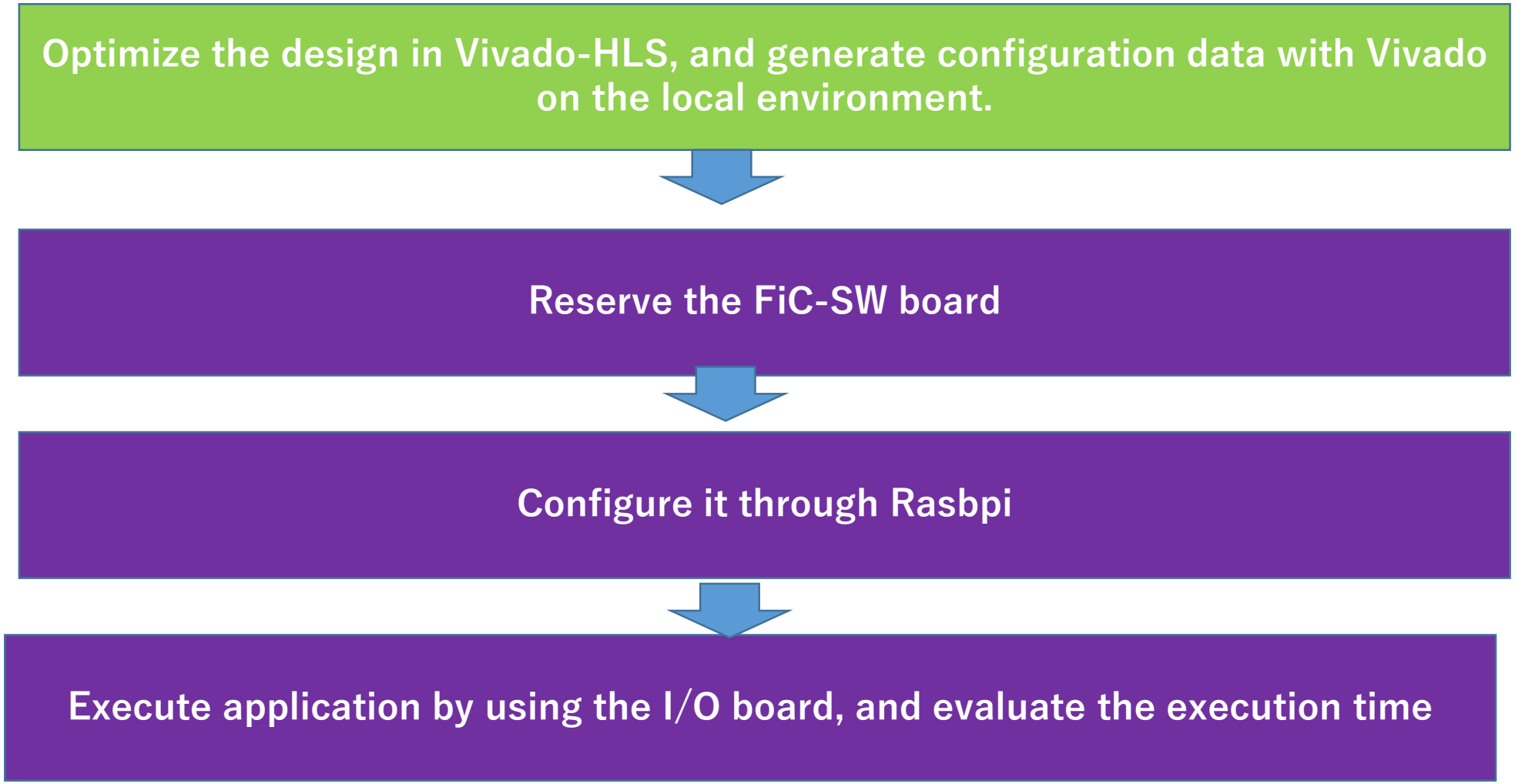
FPGA programming contest for CNN:
Special Course of Computer Architecture (Graduated School)

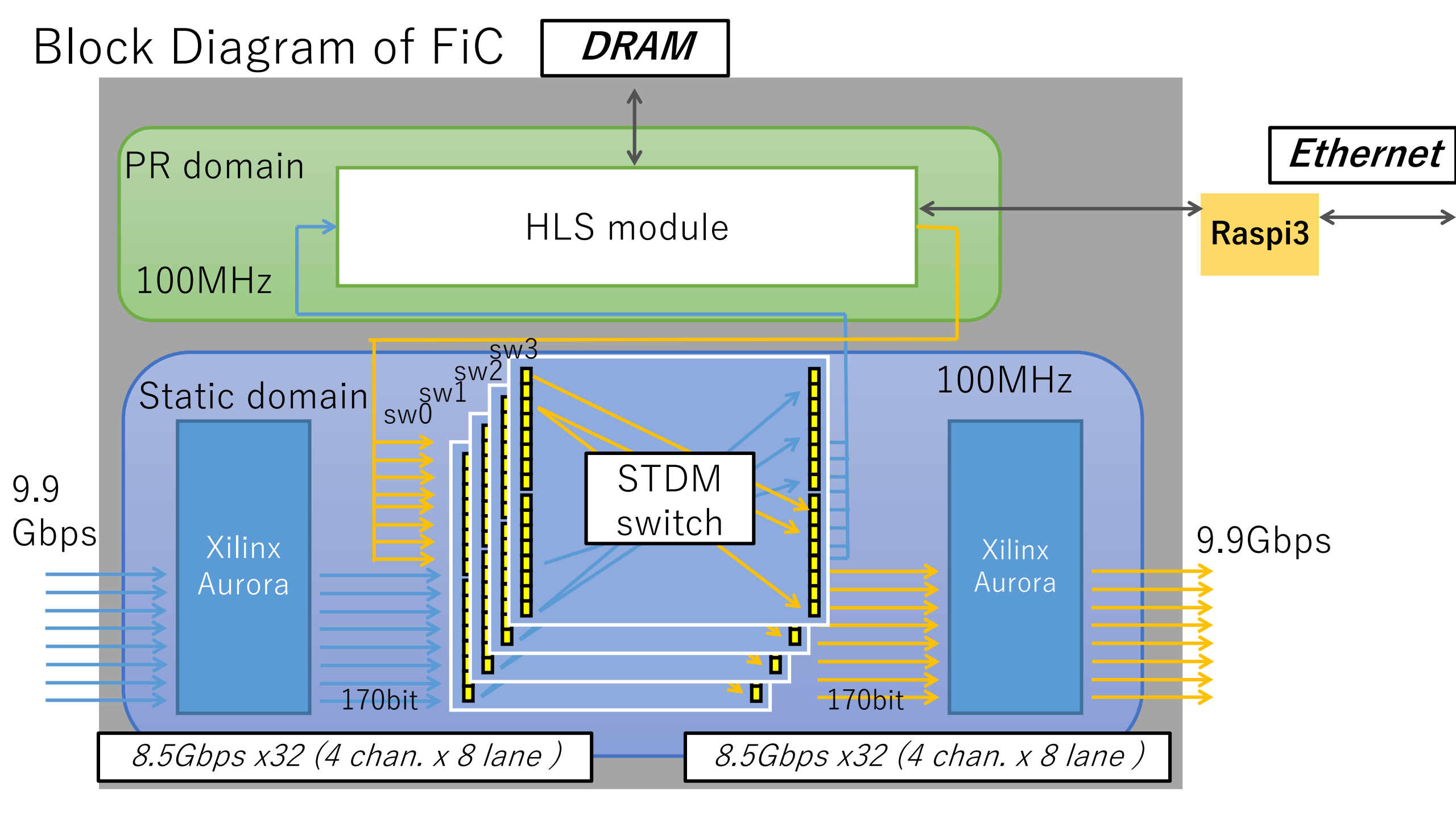
A simple CNN Lennet



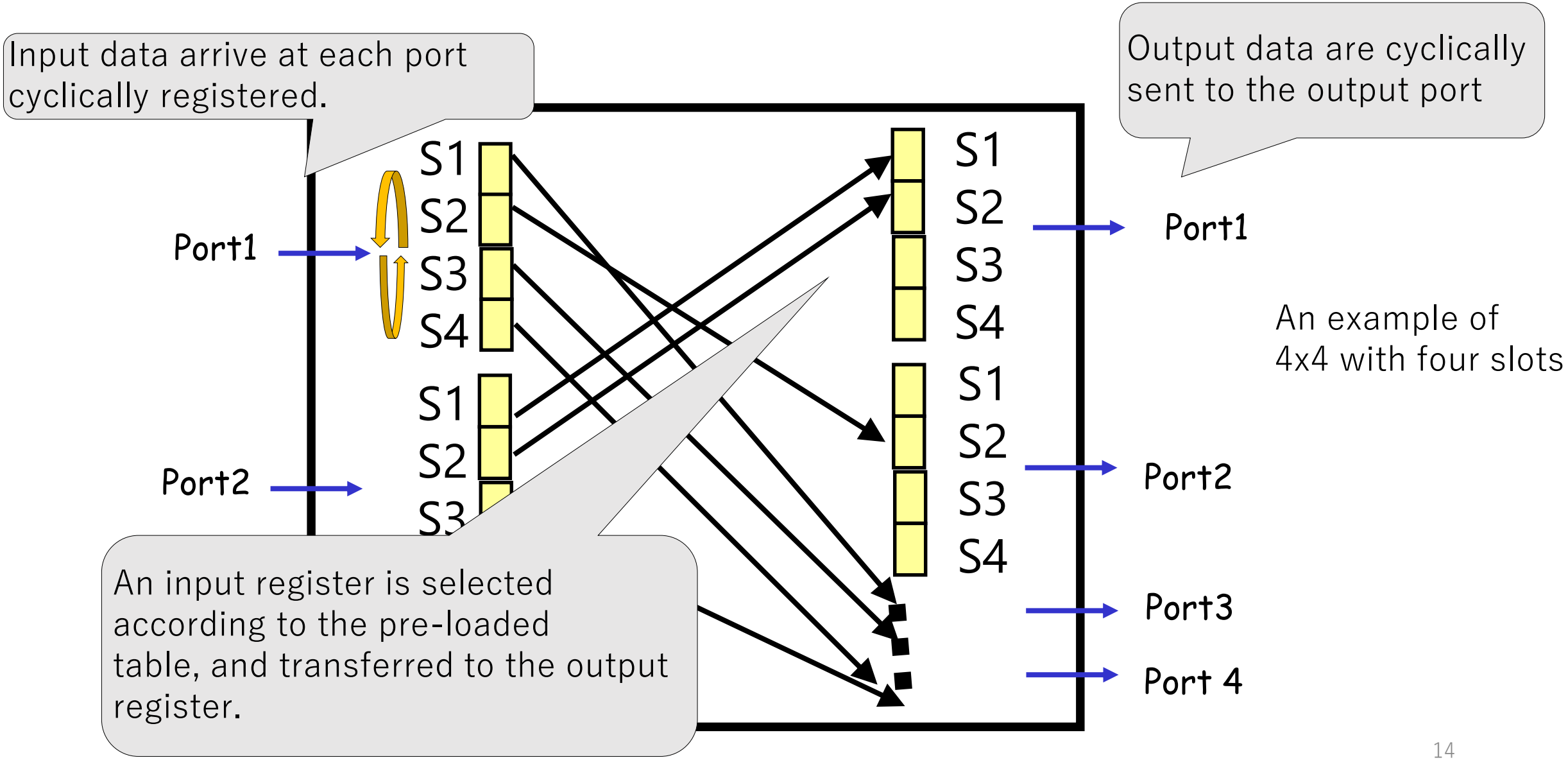
A design template without optimization is distributed.

Executing lenet in the cloud.

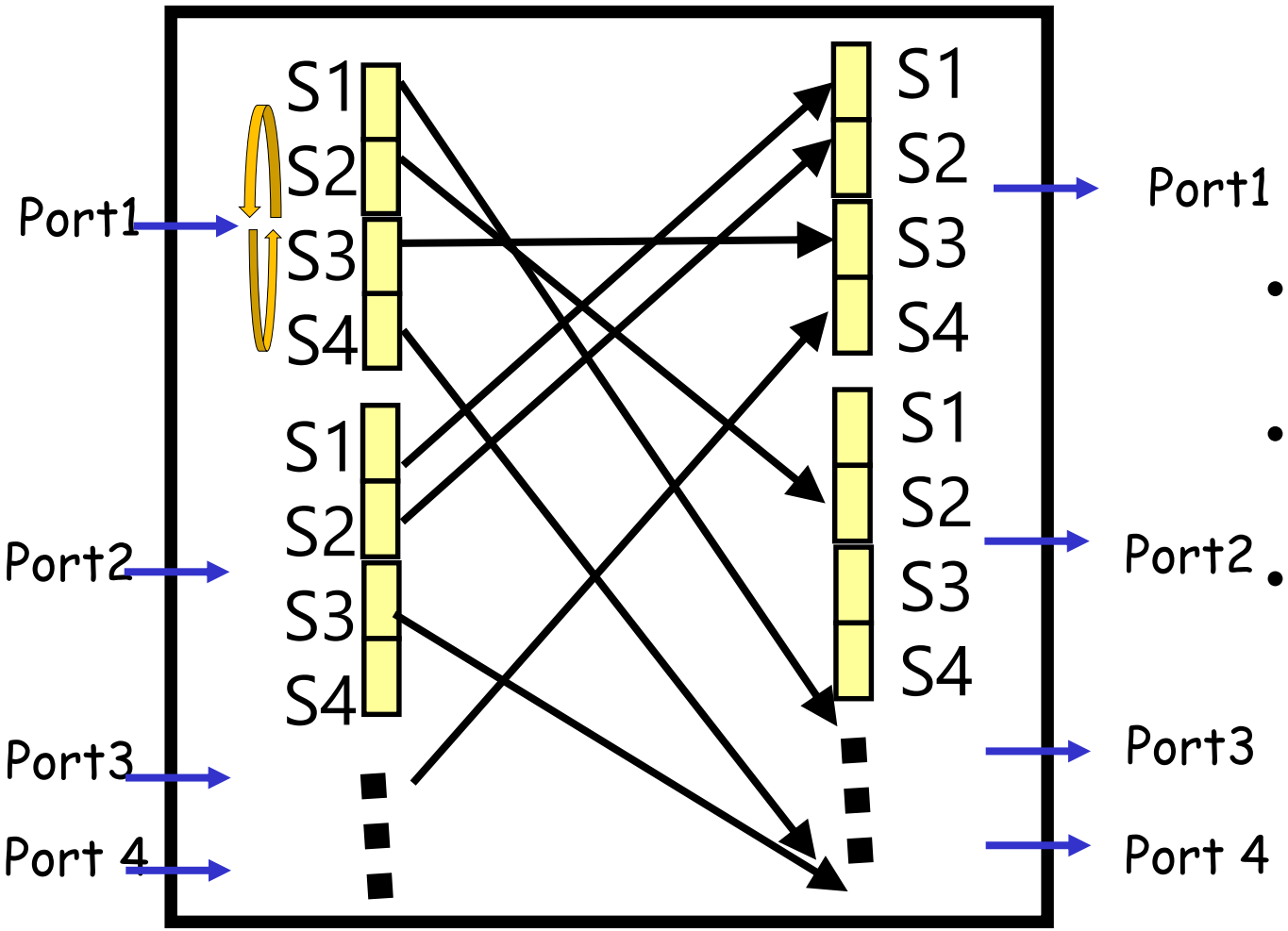



Block Diagram of FiC
DRAM
PR domain
HLS module
100MHz
Ethernet
Raspi3
Static domain
sw0
sw1
sw2
sw3
STDM switch
100MHz
9.9 Gbps
Xilinx Aurora
170bit
170bit
Xilinx Aurora
9.9Gbps
8.5Gbps x32 (4 chan. x 8 lane)
8.5Gbps x32 (4 chan. x 8 lane)

STDM (Static Time Division Multiplexing)

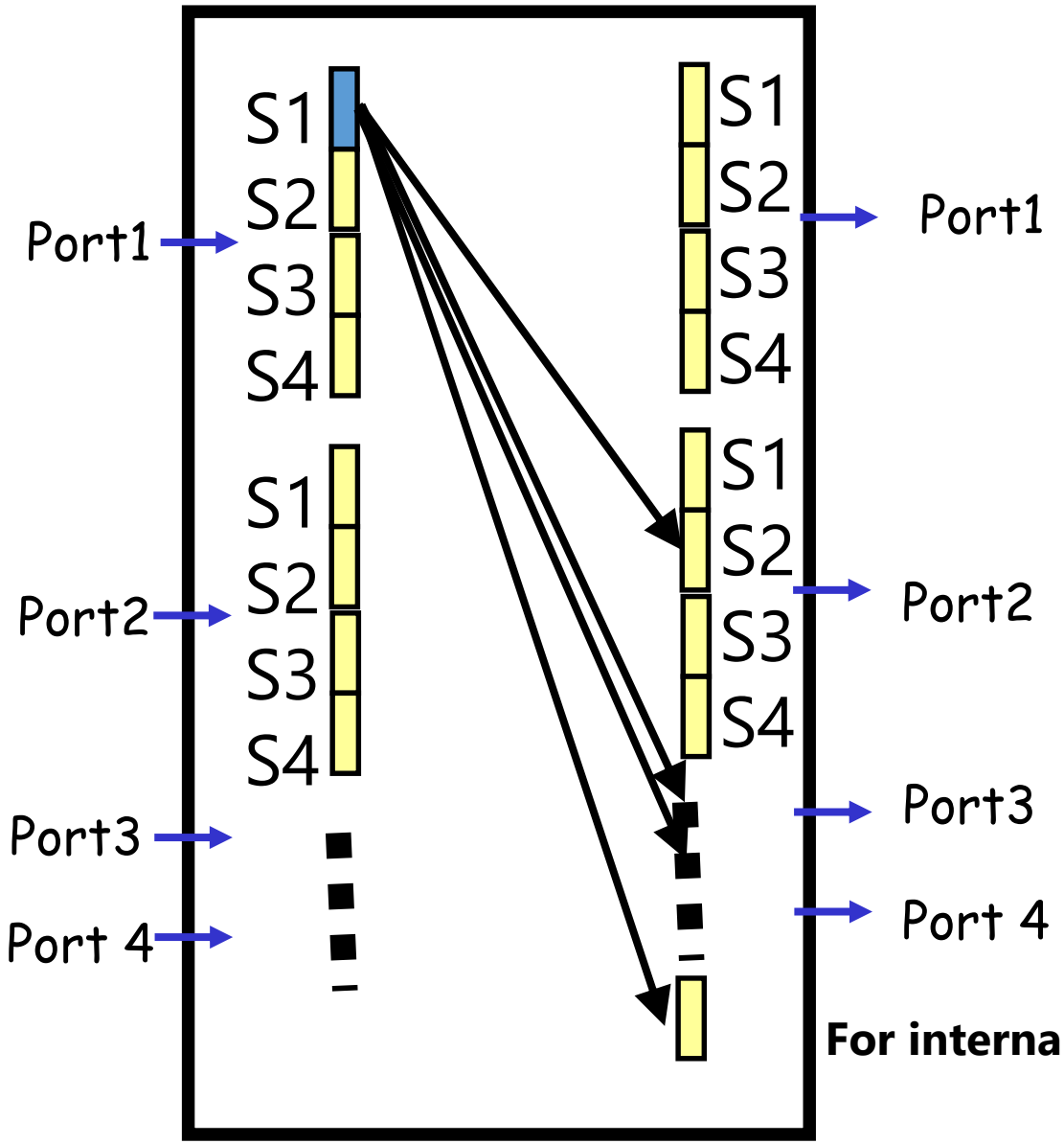


STDM (Static Time Division Multiplexing)



- A circuit is established between source and destination.
- Latency and bandwidth are kept.
- Latency = 45+2 x (# of slots) clock cycles

Multicast using the STDM

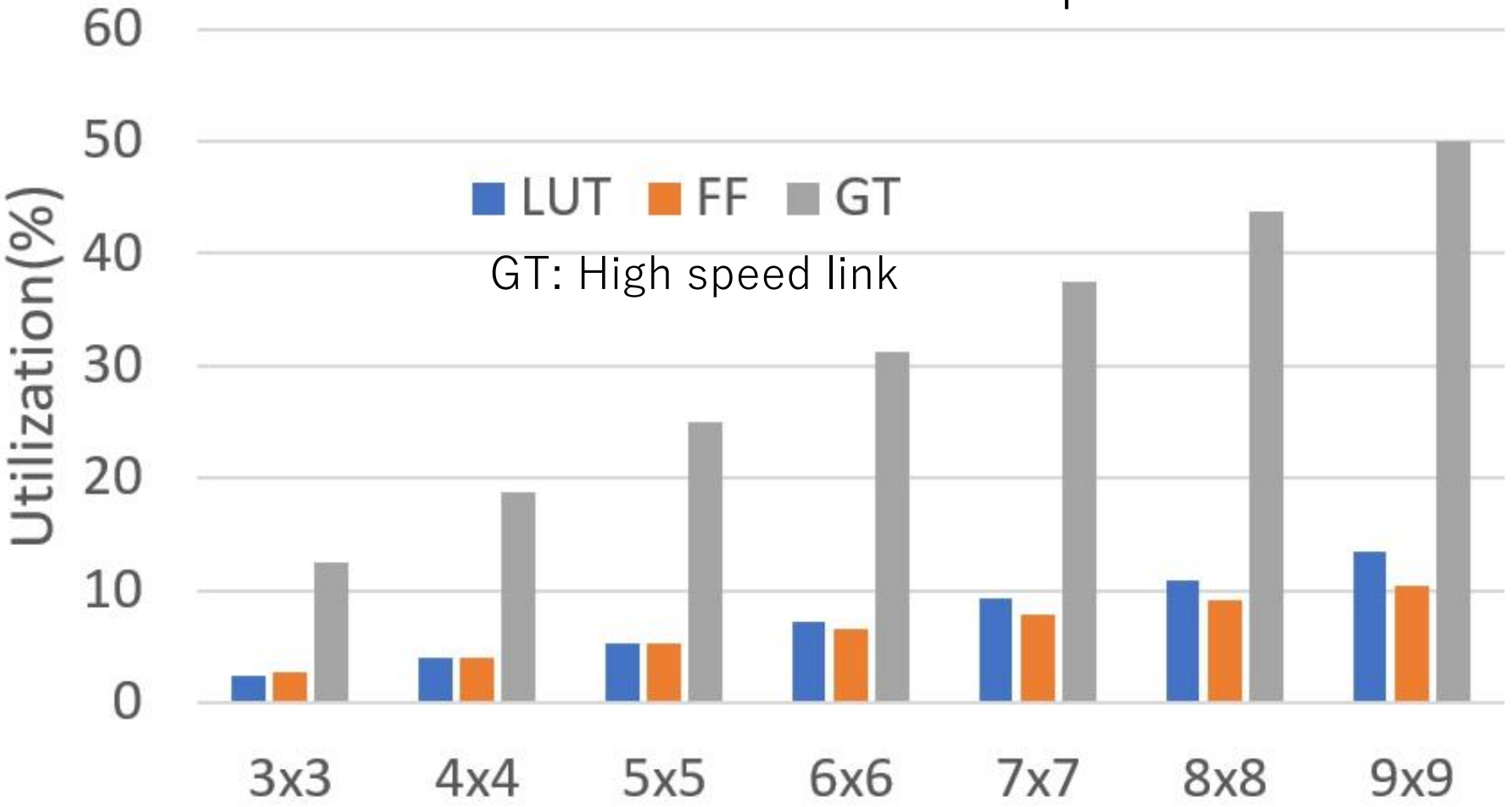


Multicast is done efficiently.

Multiple outputs can receive the same data in a specific slot.

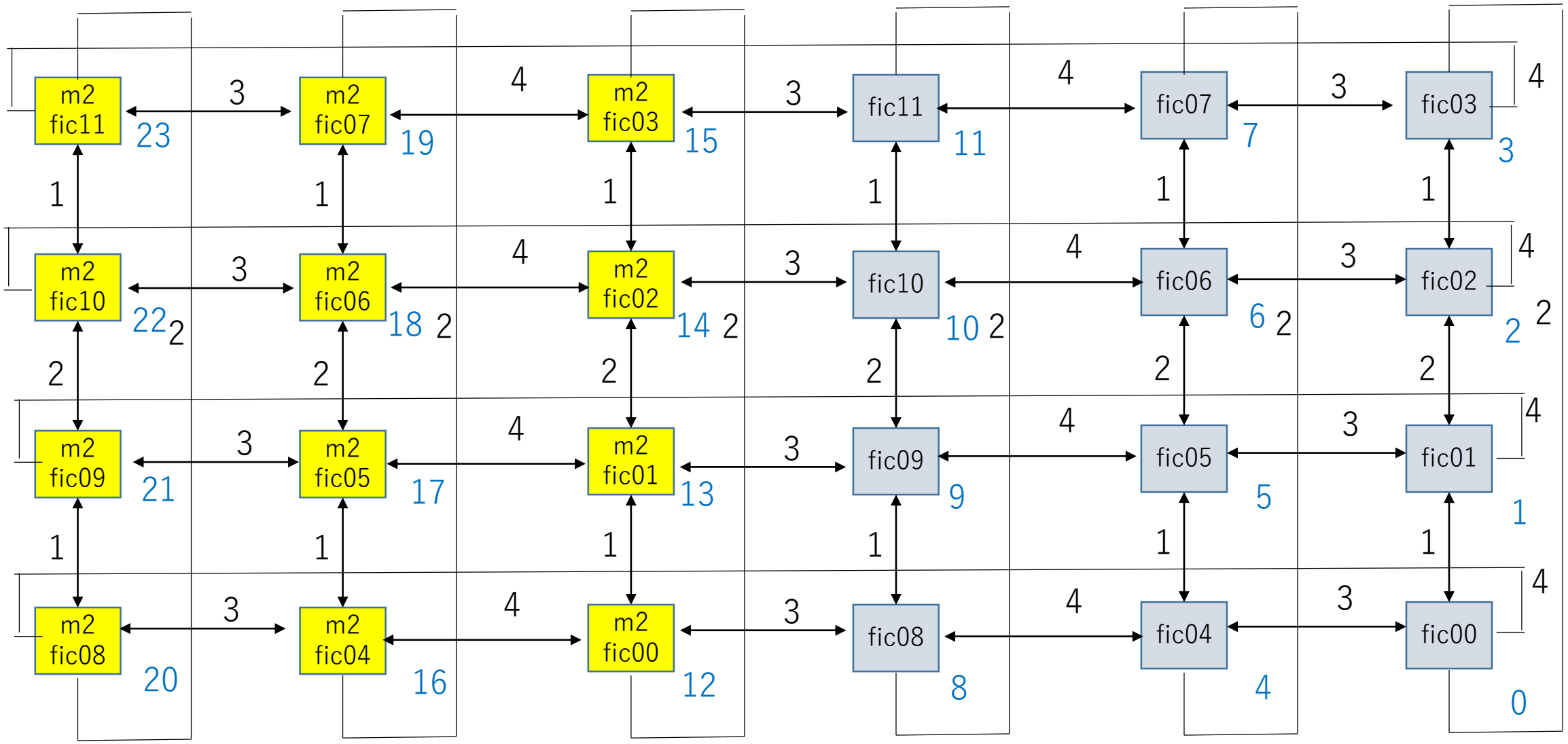
The resource usage

4 switches are provided for each channel.

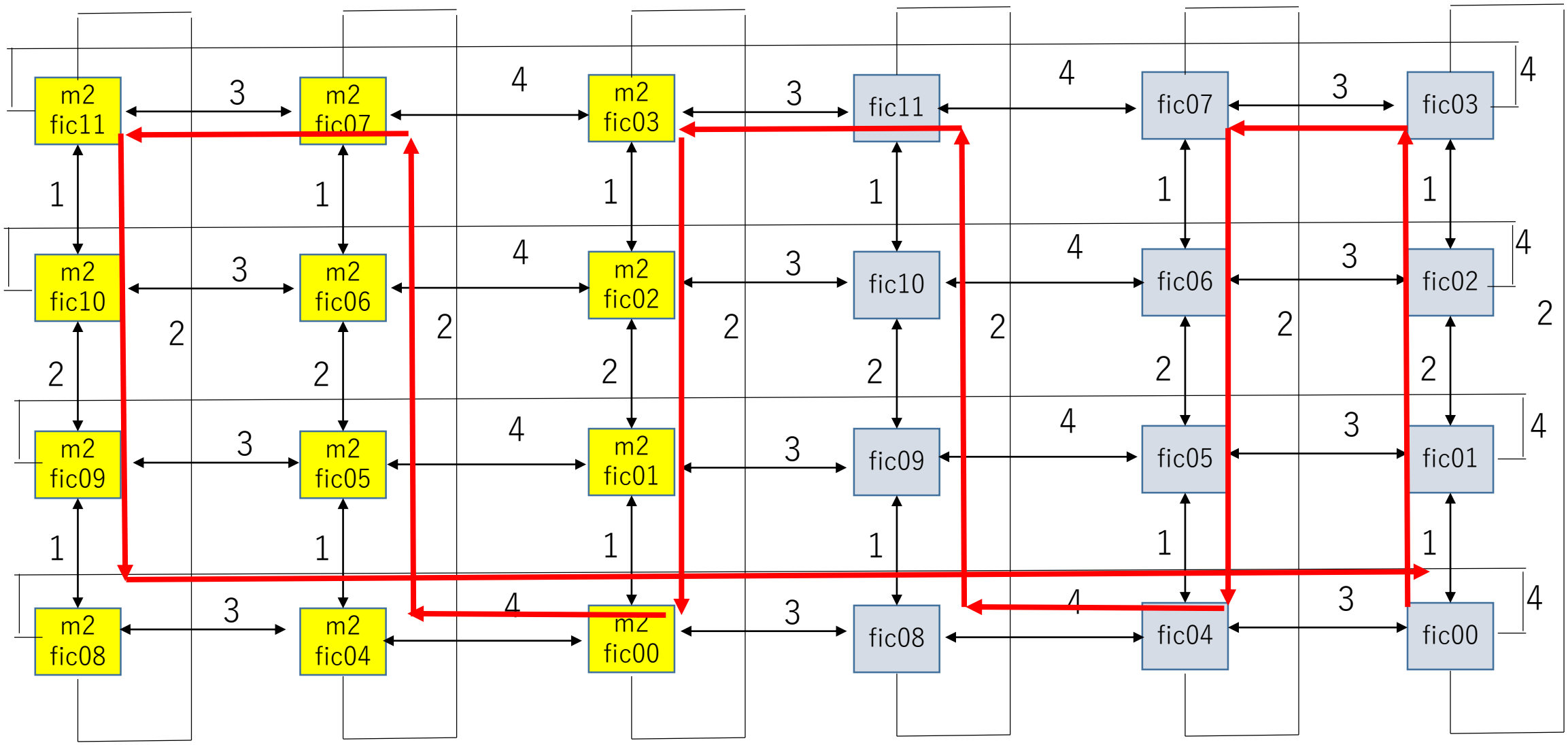


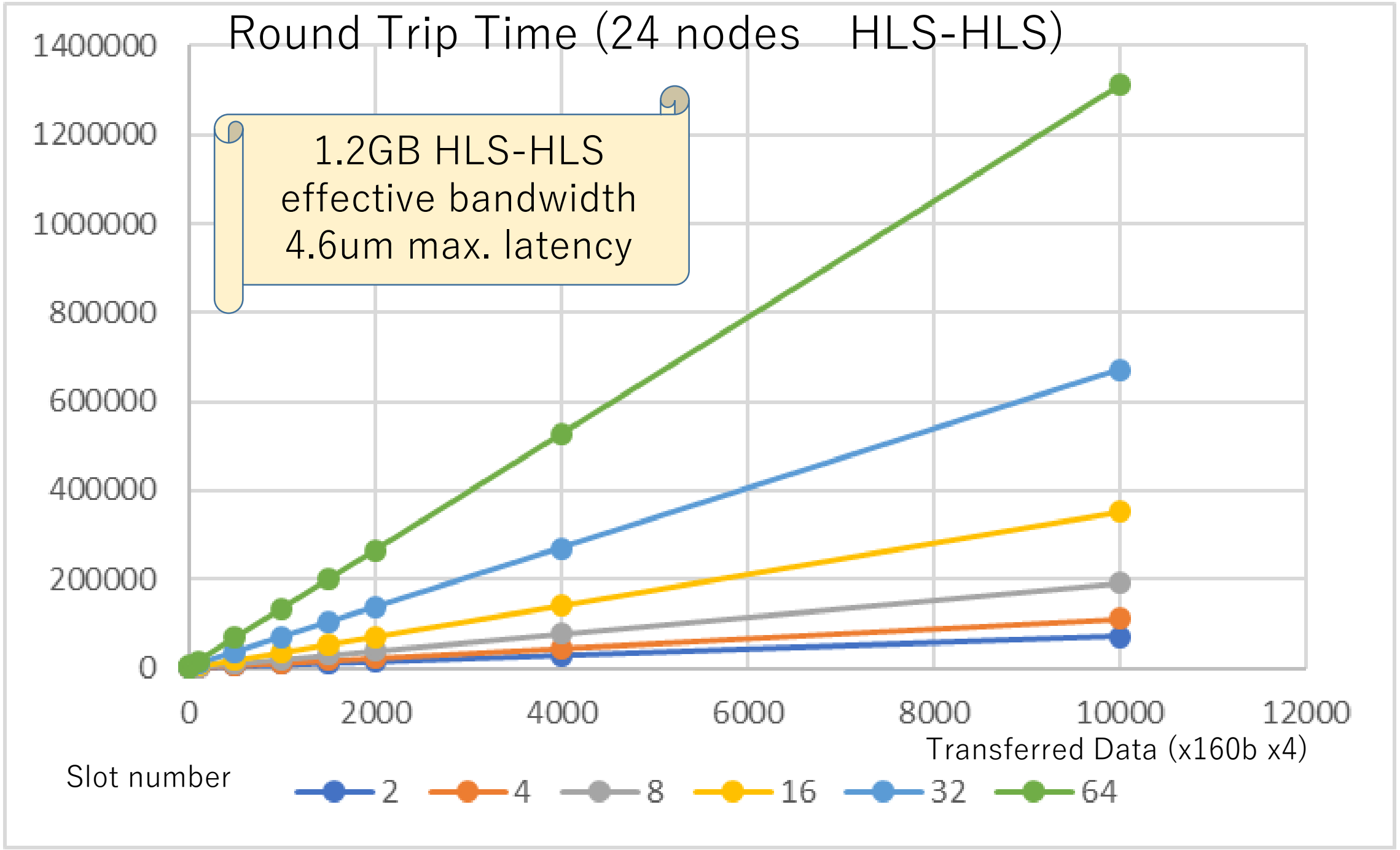
Enough design is remained for HLS design.

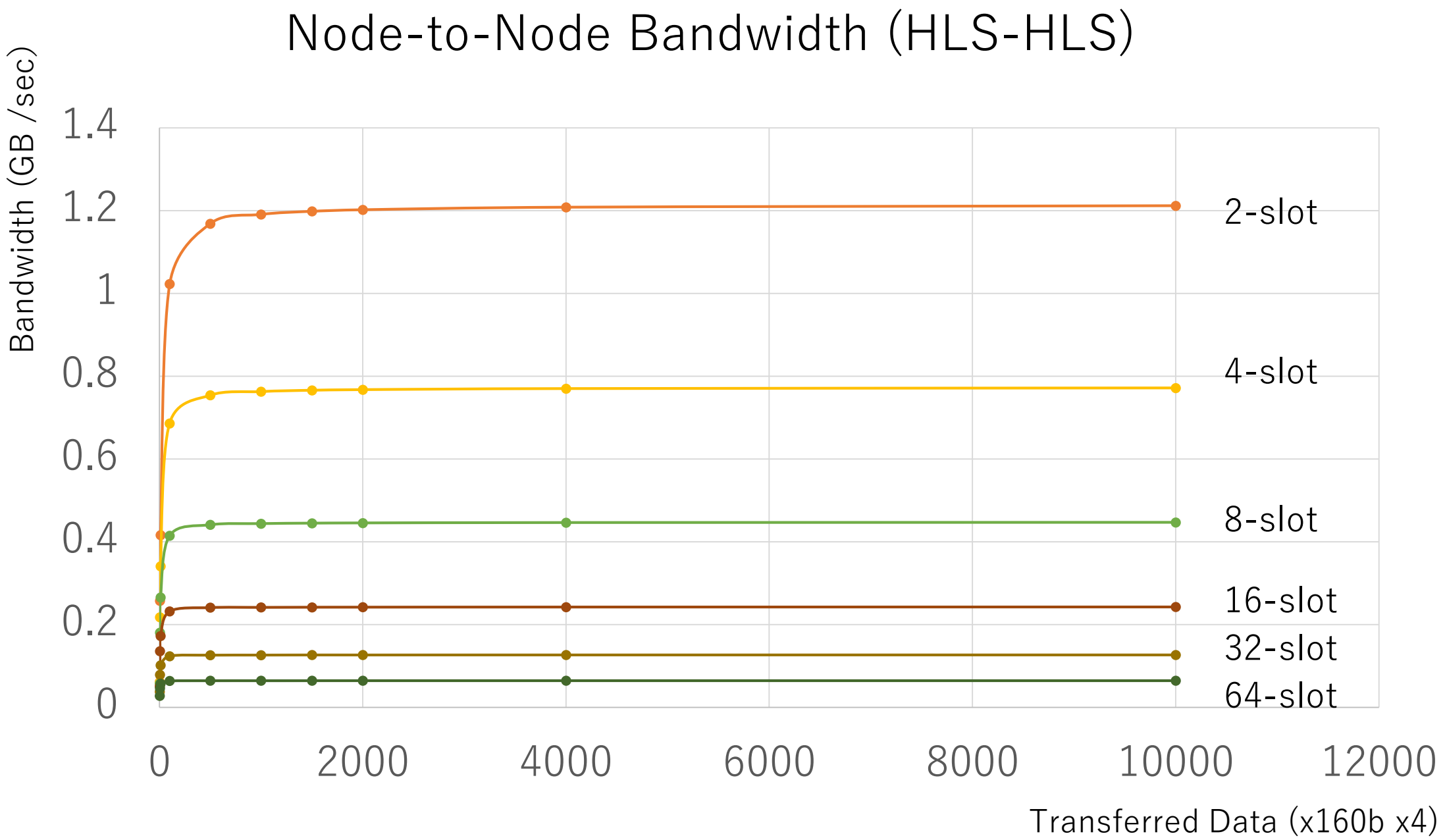
The current configuration: 4x6 torus



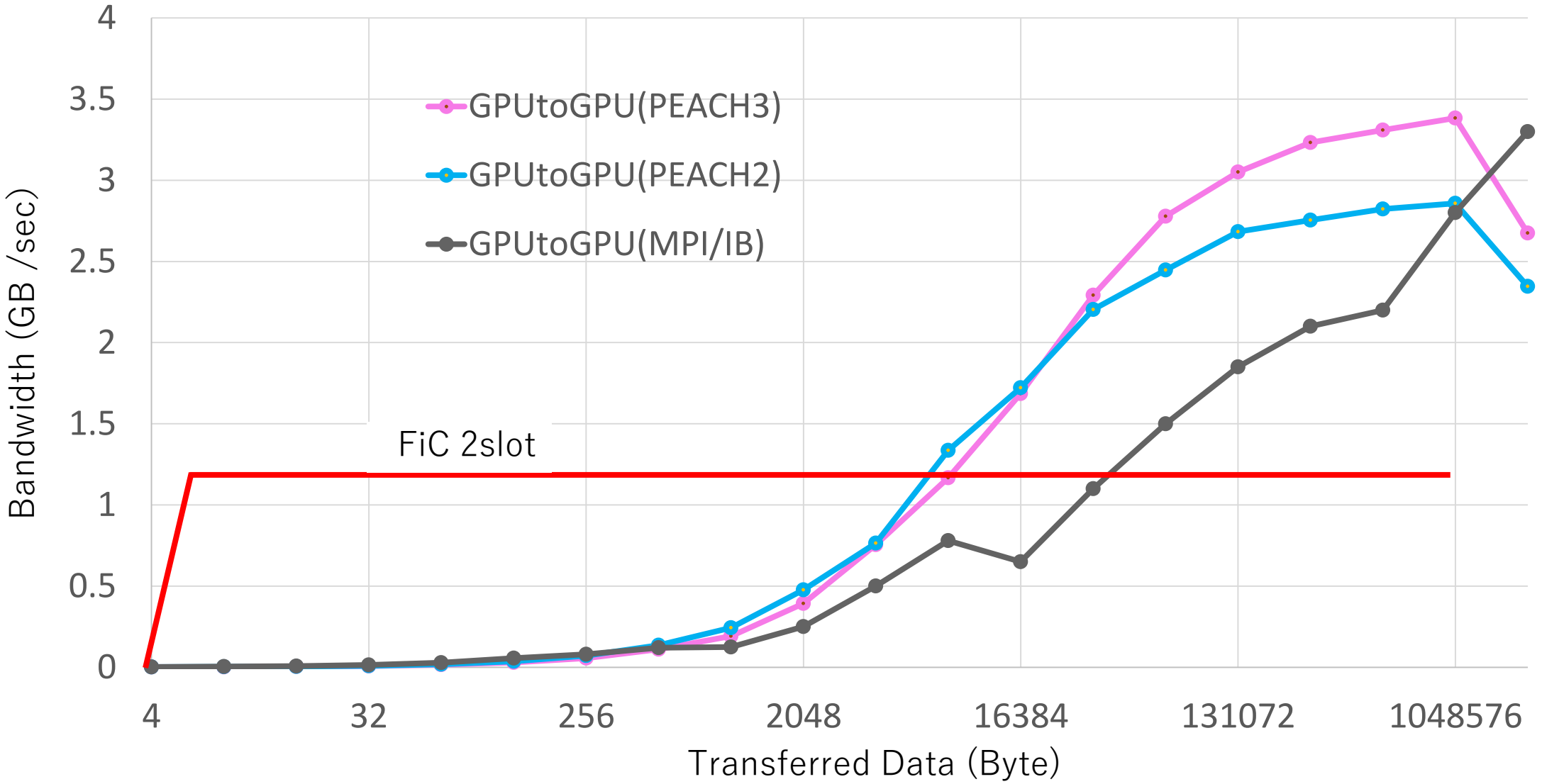
Round Trip Time (24boards)




Round Trip Time (24 nodes　HLS-HLS)
1.2GB HLS-HLS
effective bandwidth
4.6um max. latency
Latency (clock cycles)
1400000
1200000
1000000
800000
600000
400000
200000
0
0
2000
4000
6000
8000
10000
12000
Transferred Data (x160b x4)
Slot number
2
4
8
16
32
64


Node-to-Node Bandwidth (HLS-HLS)
Bandwidth (GB /sec)
1.4
1.2
1
0.8
0.6
0.4
0.2
0
0
2000
4000
6000
8000
10000
12000
Transferred Data (x160b x4)
2-slot
4-slot
8-slot
16-slot
32-slot
64-slot

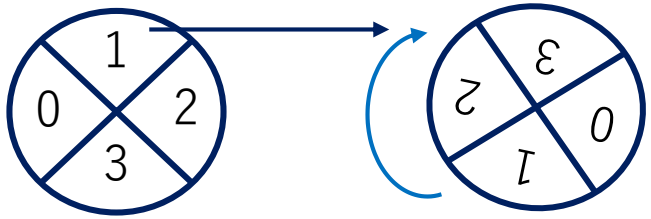
Comparison between other FPGA Connected Multi-GPU system and MPI/Infiniband



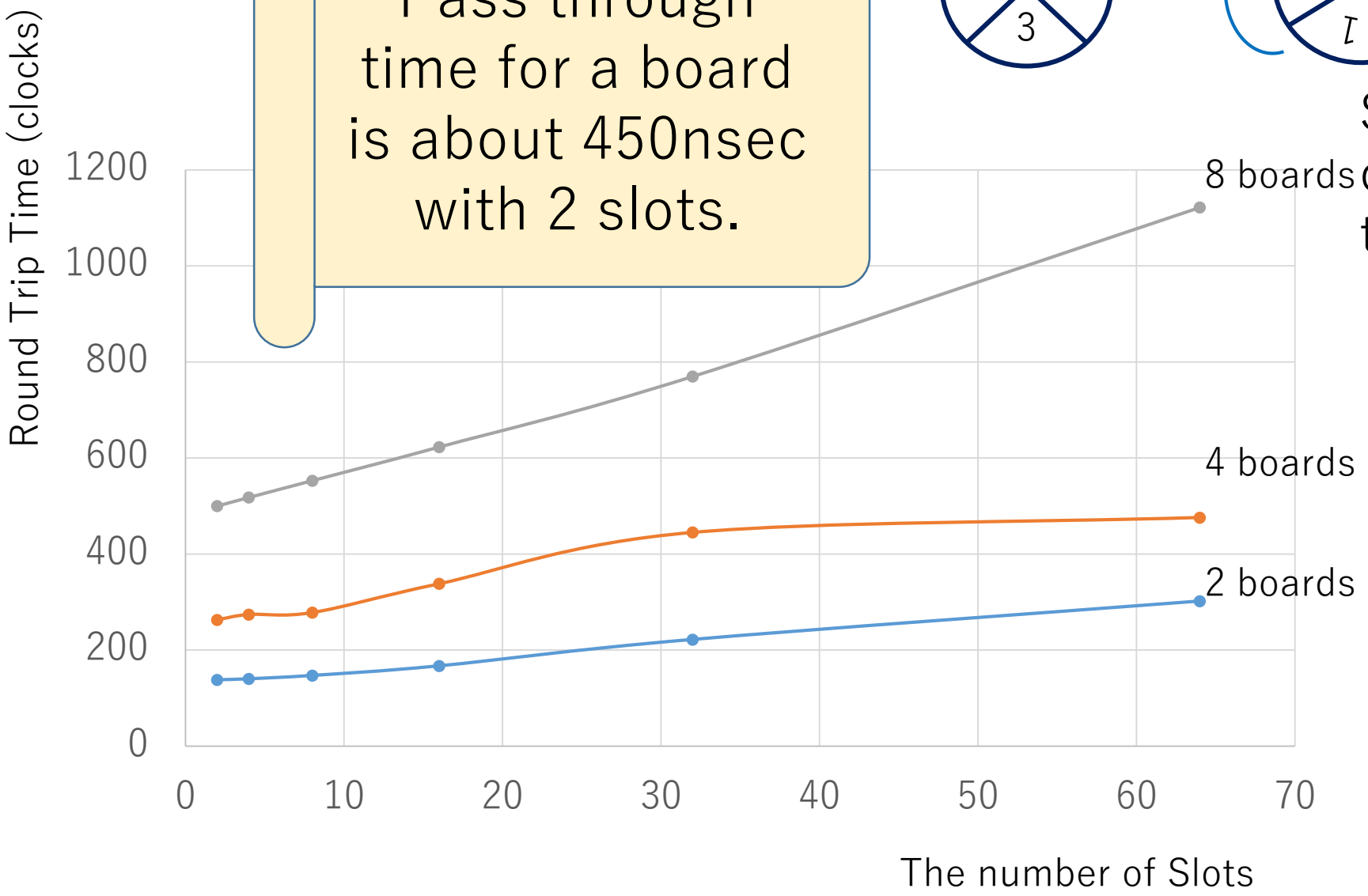
Kaneda, et.al “Performance Evaluation of PEACH3: FPGA switch for Tightly Coupled Accelerators,” HEART 2017

Round Trip Time vs. The number of Slots

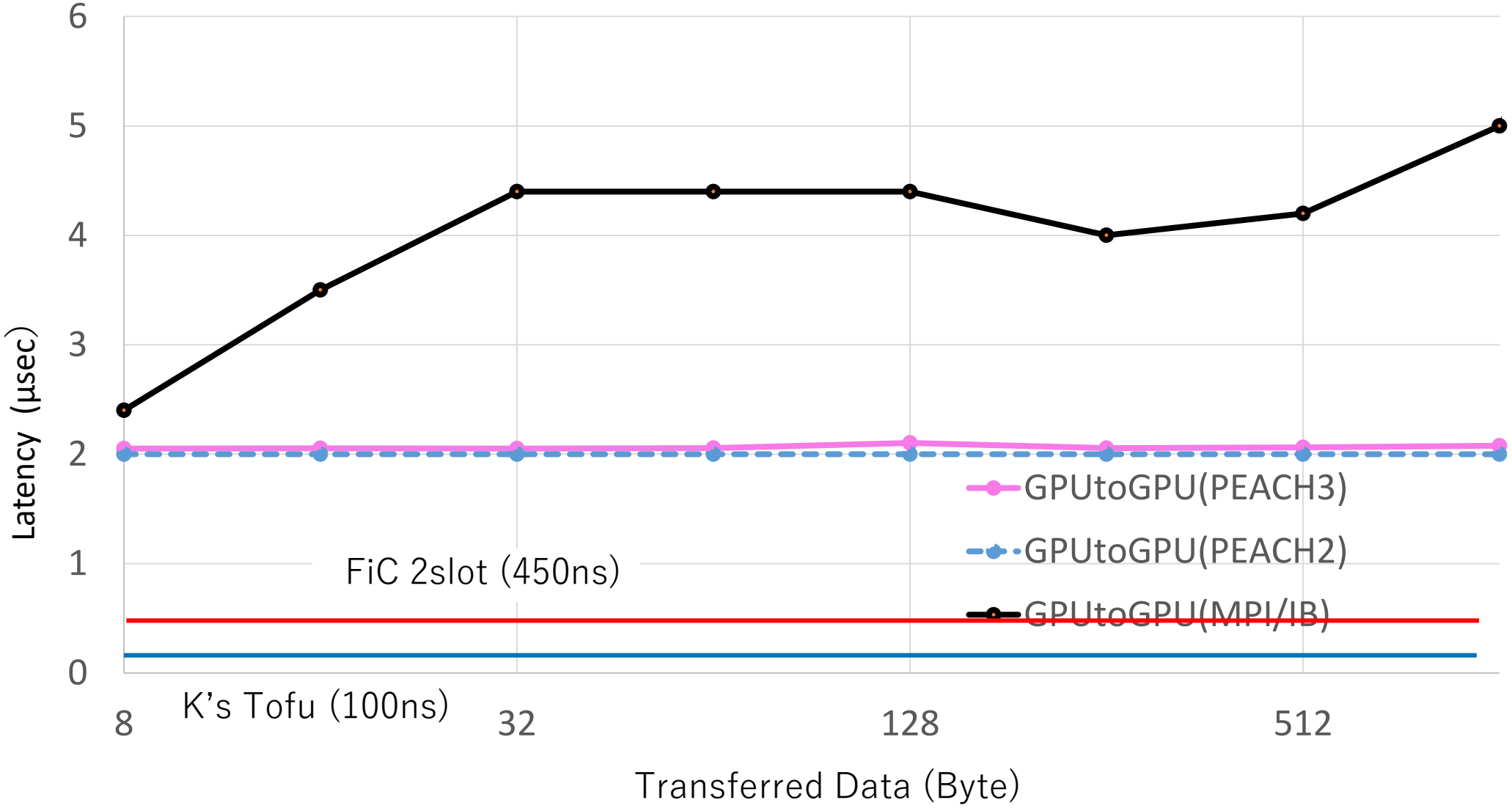
Pass through time for a board is about 450nsec with 2 slots.



Slot synchronous delay increases the round trip time

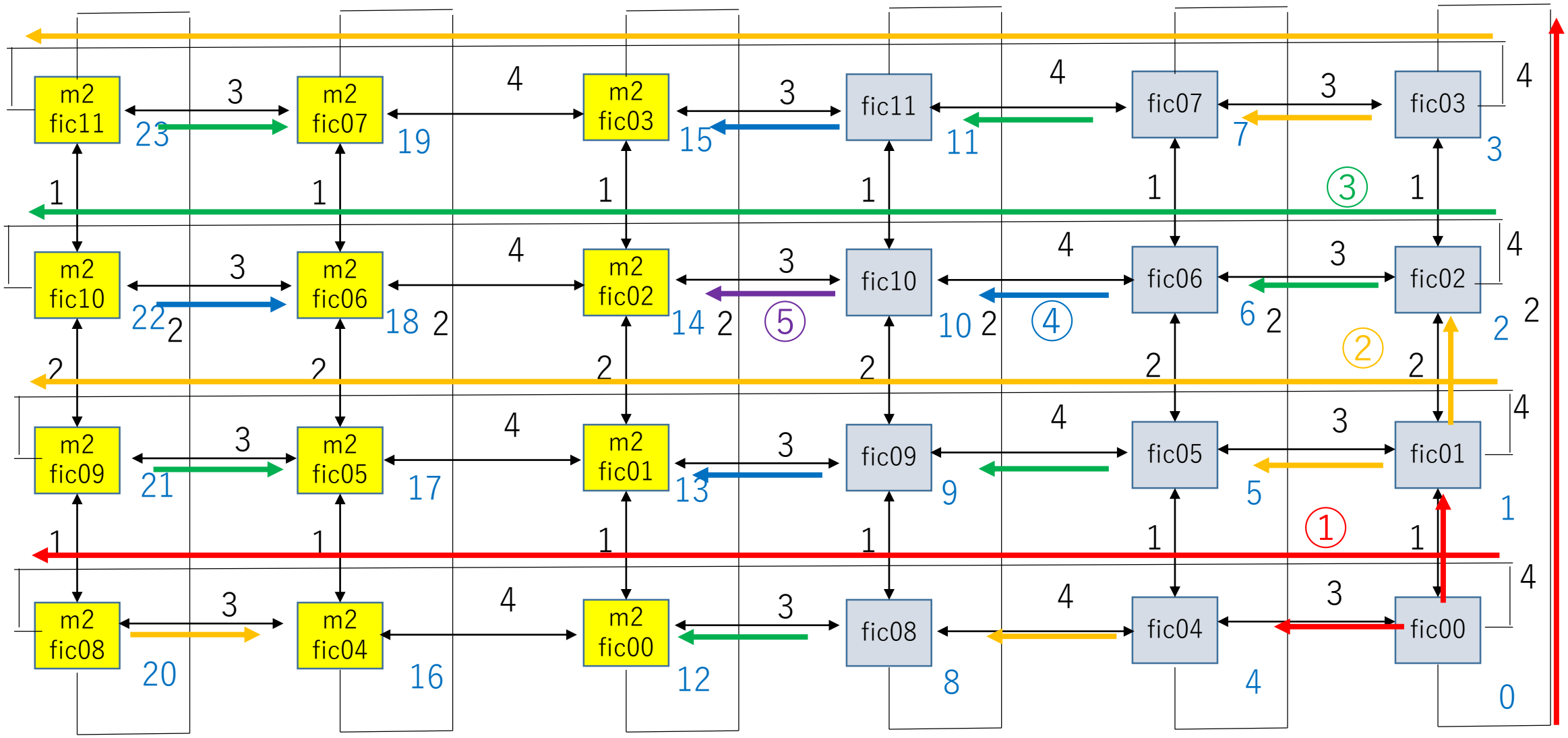


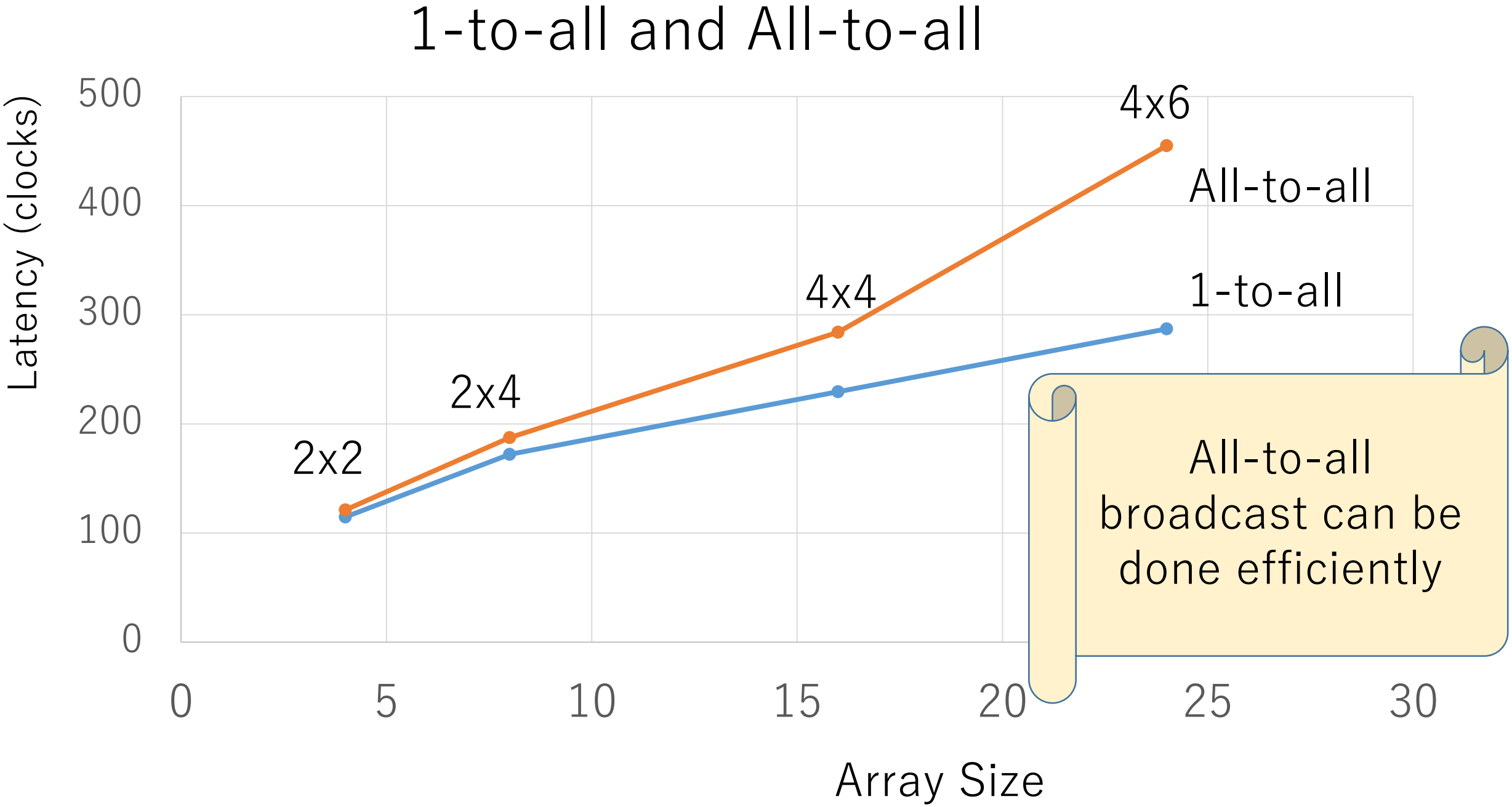
Comparison between other systems



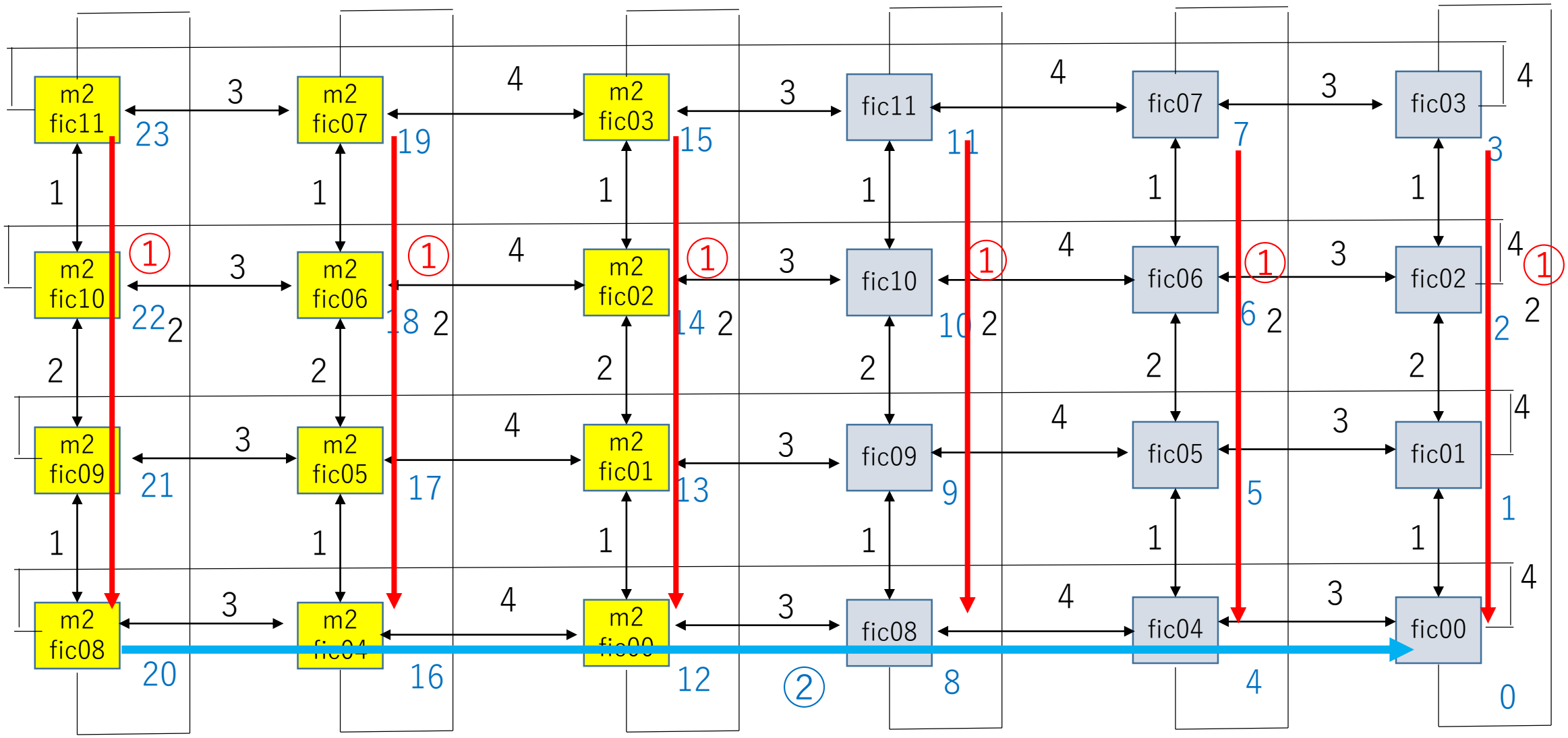
Ajima, et.al "The Tofu Interconnect," Hot Interconnect 2012

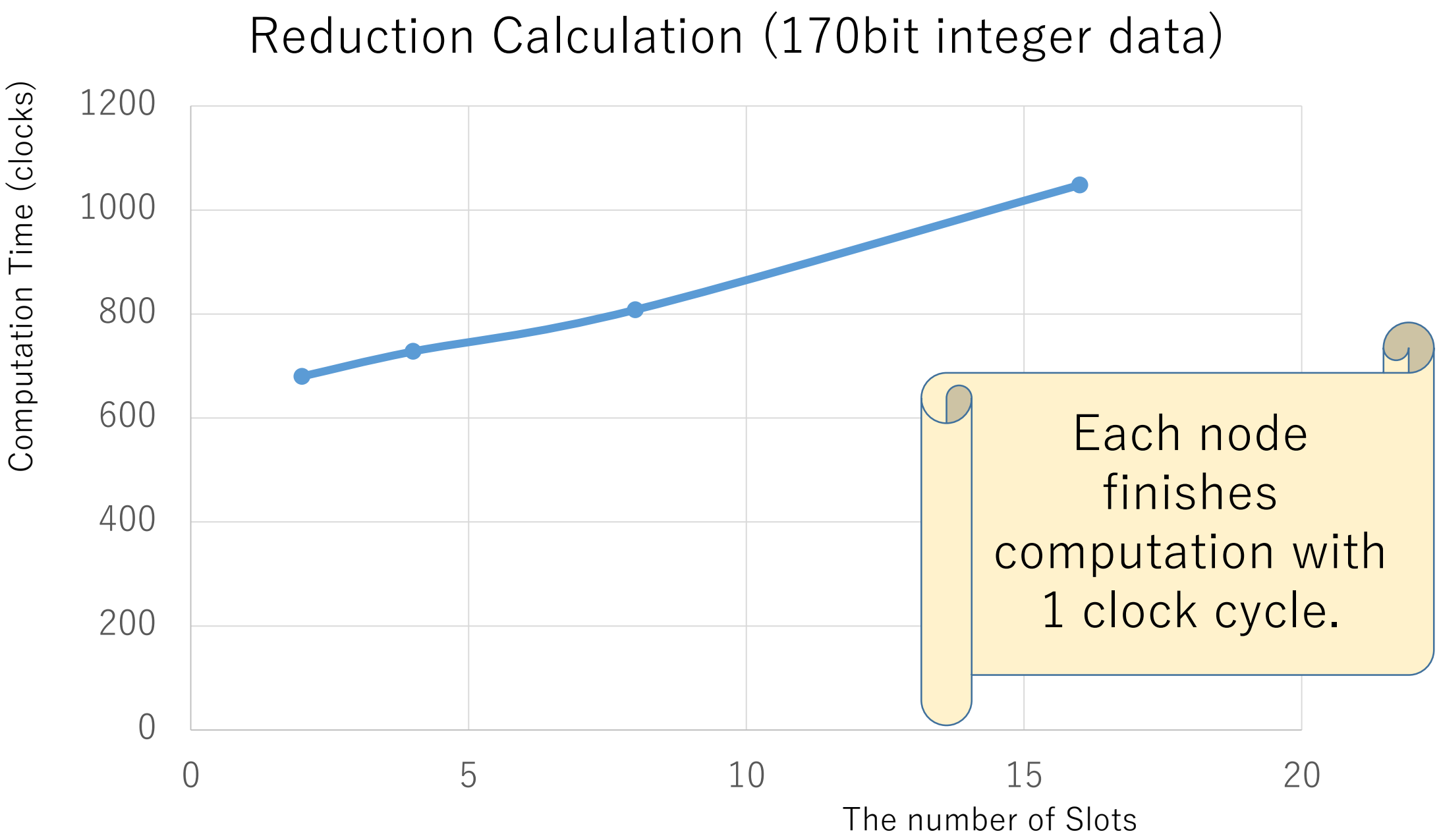
Broadcast can be done with the maximum hop



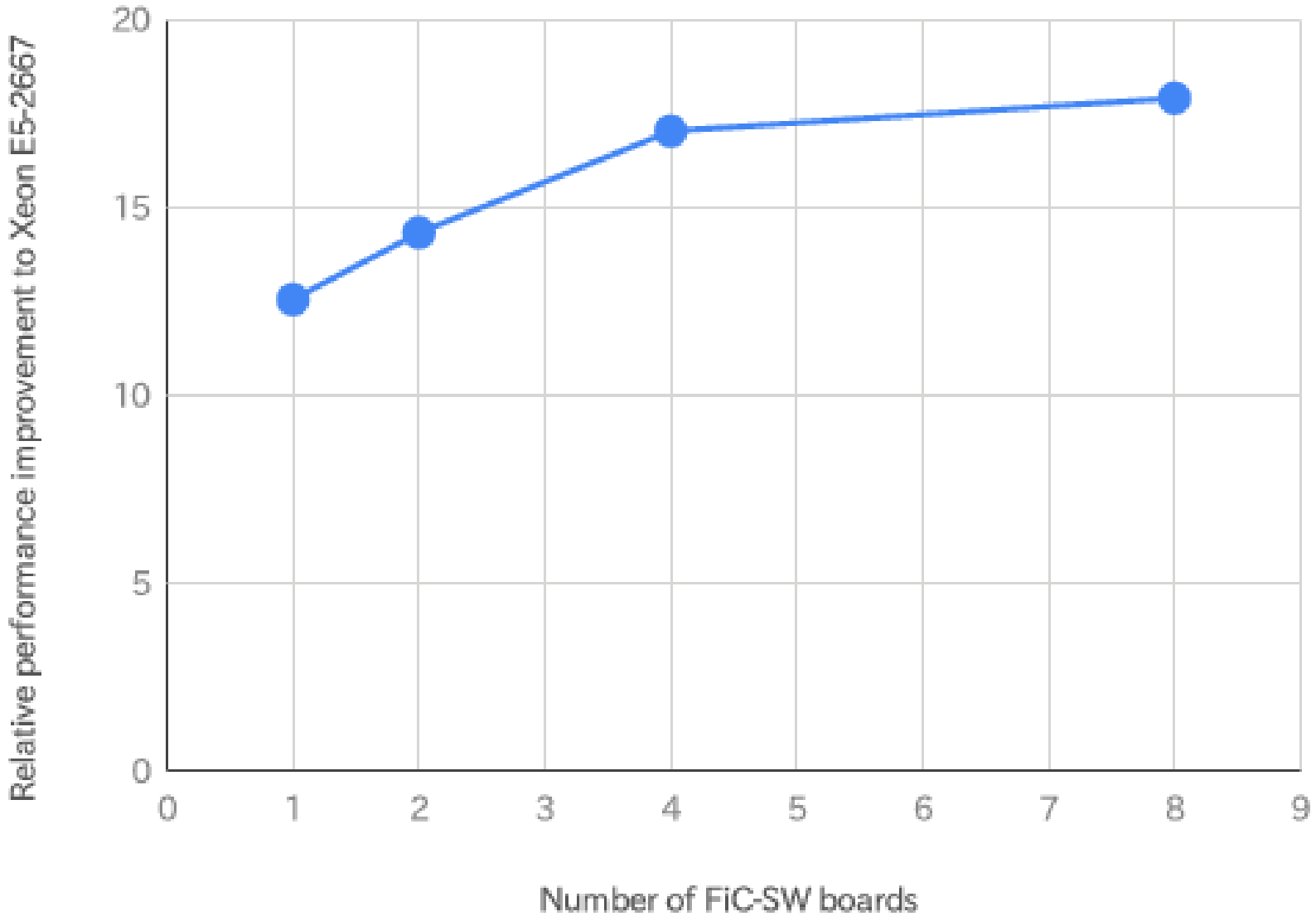

1-to-all and All-to-all
Latency (clocks)
500
400
300
200
100
0
0
5
10
15
20
25
30
Array Size
4x6
All-to-all
4x4
1-to-all
2x4
2x2
All-to-all broadcast can be done efficiently

Reduction Computation




Reduction Calculation (170bit integer data)
Computation Time (clocks)
1200
1000
800
600
400
200
0
0
5
10
15
20
The number of Slots
Each node finishes computation with 1 clock cycle.

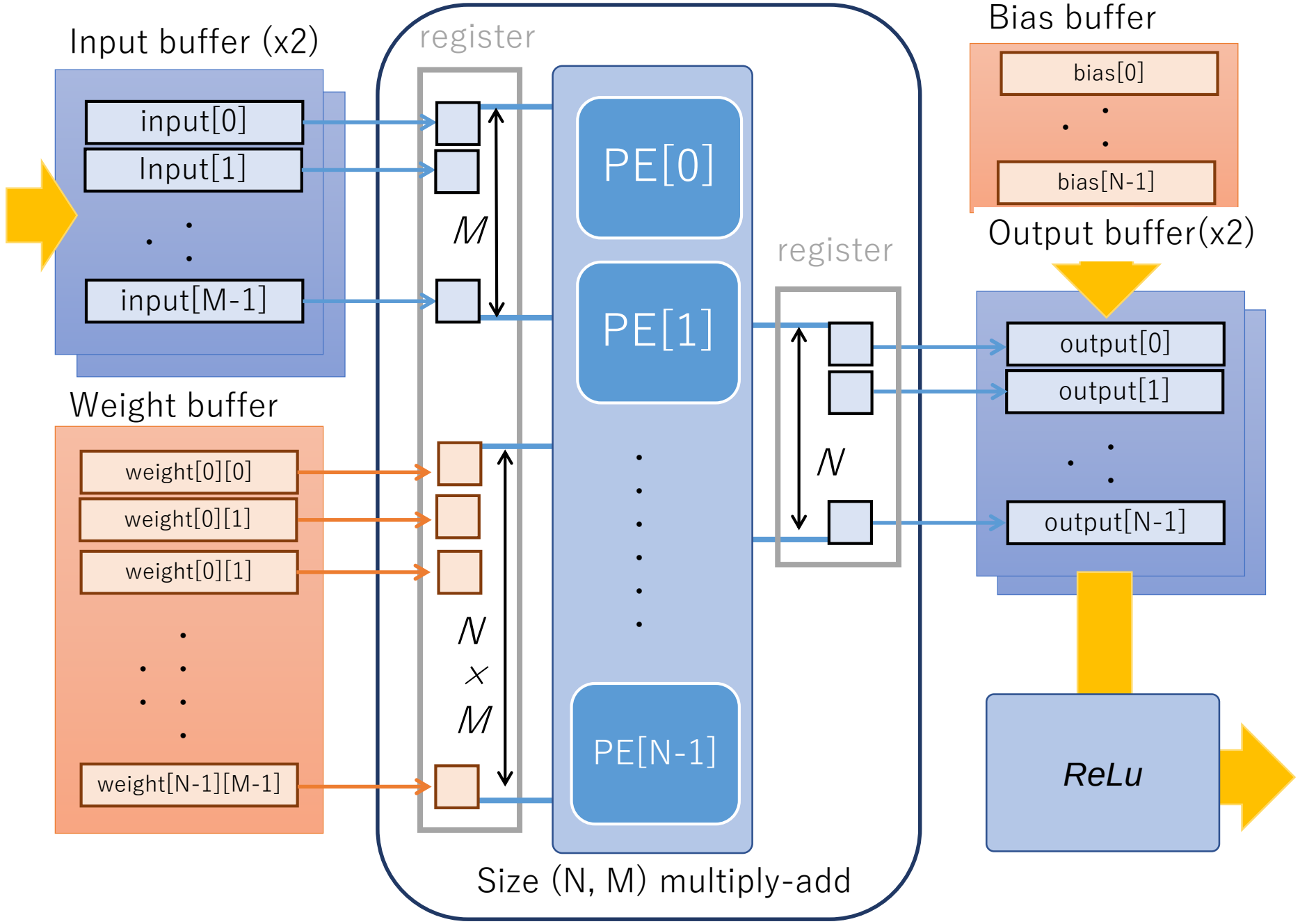
Communication centric example: CG method



17.8x performance improvement
to Xeon E5-2667 16 CPUs 2.9GHz

All-to-all data transfer needs
9000 clock cycles

Implementation Example (LeNet Fully Connected layer)



Fully connection layer of the Lenet

	Frequency（MH z ）	Power（W）	image/sec	GOPS	GOPS/W
1 board	100	17.89	23551	120.58	6.74
4 boards	100	71.56	94226	482.34	6.74

	Frequency（MHz）	Power（W）	GOPS	GOPS/W
FiC 4boards	100	71.56	482.34	6.74
Stratex-V [1]	120	25.8	136.5	5.29
KCU060　[2]	200	25.0	172.0	6.88

Higher performance is achieved with the similar power efficiency compared with a single FPGA system.

[1] N.Suda, V.Chandra, G.Dasika, et.al “Throughput-optimized open-based FPGA Accelerator for large-scale comvolutional neural networks,” FPGA2016.
[2] C.Zhang, Z.Fang, P.Zhao, P.Pan, J.Cong, “Caffeine: Towards uniformed representation and acceleration for deep convolutional neural networks,” ICCCAD2017.

Conclusion and future work

- A multi-FPGA system for MEC has been developed:
  - The management system has been used for a design contest in a class.
  - The latency and bandwidth can be guaranteed.
    - 34GB max. aggregated throughput with 32I channel
    - 1.2GB/sec 1-to-1 HLS throughput
    - 450nsec pass through time
    - 4.6 μ sec maximum HLS-to-HLS latency in the 24 nodes.
- Now on going…
  - We distributed small systems to other groups: TAT, Shibaura Tech, Tokai University and NEC.
  - Practical application is now under development:
    - Genome matching, Alexnet, Lesnet, RNA and distributed learning.