



A 21.9ns 15.7Gbps/mm² (128,15) BOSS FEC Decoder for 5G/6G URLLC Applications

Dongyun Kam¹, Sangbu Yun¹, Jeongwon Choe¹,
Zhengya Zhang², Namyoon Lee³, Youngjoo Lee¹



¹Pohang University of Science and Technology, Korea

²University of Michigan, MI

³Korea University, Korea



Outline

- **Introduction**
- **Backgrounds**
 - BOSS encoding & decoding algorithm
- **Proposed BOSS decoder**
- **Measurement Results**
- **Conclusion**

Introduction



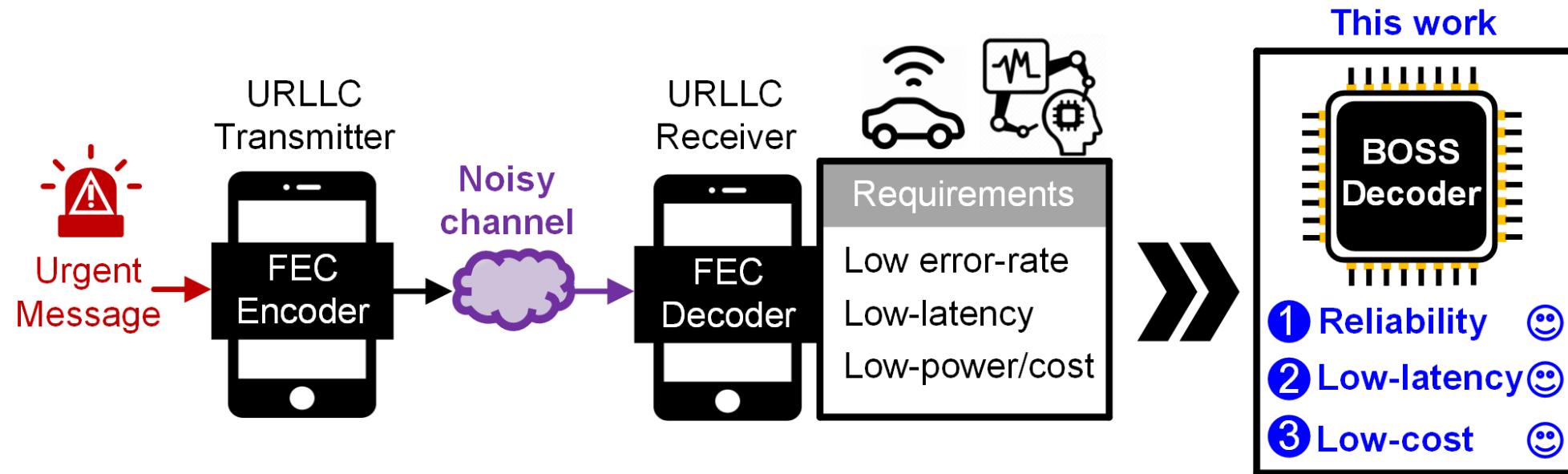
[Telesurgery]



[C-V2X]

- **5G/6G Ultra-Reliable Low-Latency Communications (**URLLC**)**
 - Ex1) Remote Healthcare
 - Ex2) C-V2X (Cellular Vehicle To Everything)

Introduction



■ Designing an URLLC receiver

- **Requirements** : 1) Strong reliability 2) Low-latency 3) Low-power/cost
- To achieve 1) : Forward Error Correction (FEC) encoding/decoding
- To achieve 1), 2) and 3) : A new short-length & low-rate FEC

Introduction

Performances at low transmit power ($E_b/N_0 < 5\text{dB}$)				
Decoder	VLSI'22	TVLSI'18	ISSCC'23	This work
FEC code	5G Polar	5G LDPC	5G Polar	BOSS
Code spec	(128,15)	(200,40)	(256,240)	(128,15)
Decoder	CA-SCL	Min-Sum	GRAND	BOSS
Error-rate	Low	High	Very high	Very low
Latency	High	Low	Very high	Very low
Cost	Low	High	Very low	Very low

Best solution
for URLLC

■ Block Orthogonal Sparse Superposition (**BOSS**) code

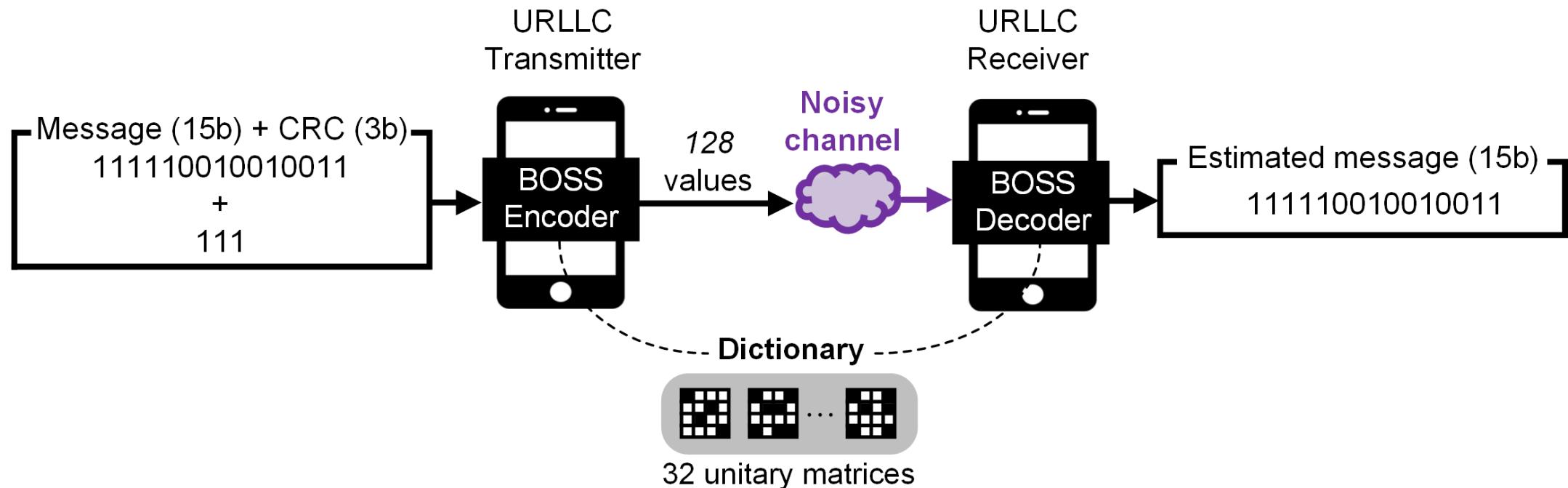
- BOSS decoder can meet three URLLC requirements*.

* D. Han et al., *IEEE TCOM*, Dec. 2023, pp.6884-6897.

Outline

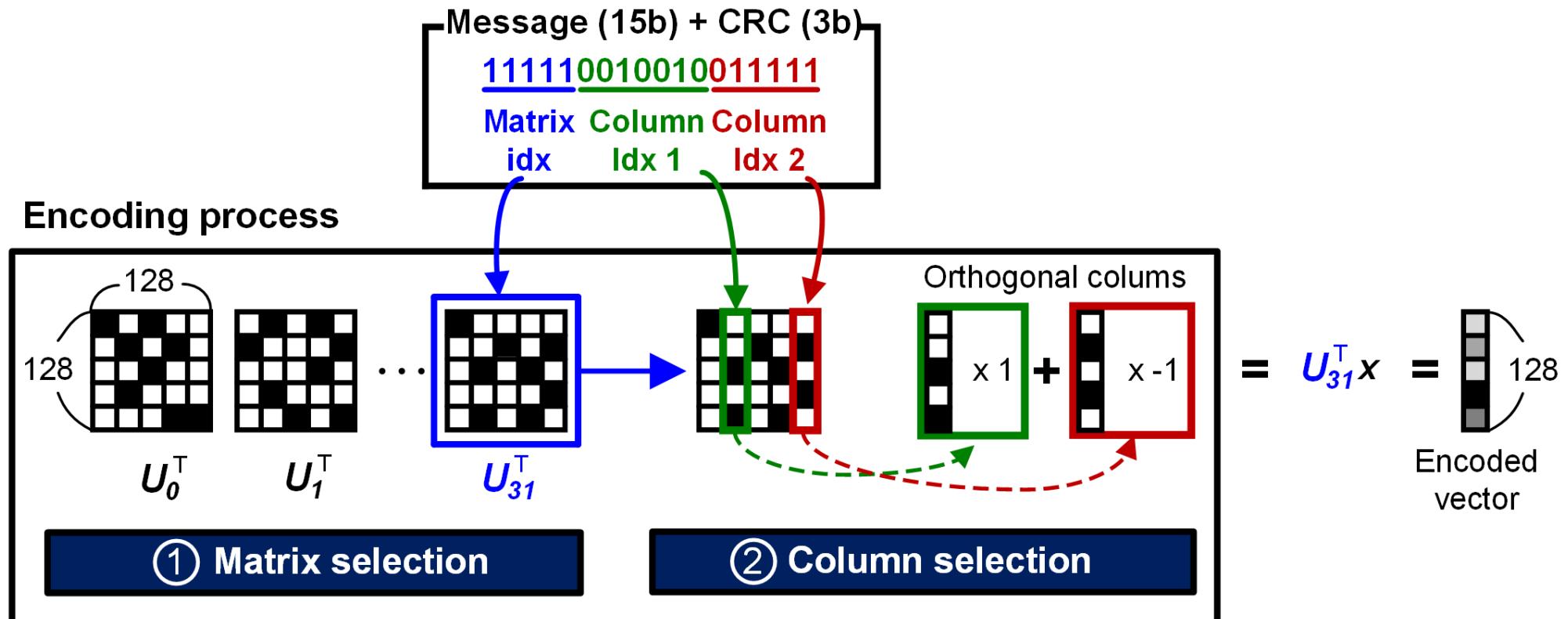
- Introduction
- Backgrounds
 - BOSS encoding & decoding algorithm
- Proposed BOSS decoder
- Measurement Results
- Conclusion

Communication system model



- **Transmission of a 15b message with 3b CRC**
 - The BOSS encoding & decoding utilize 32 128 x 128 unitary matrices.

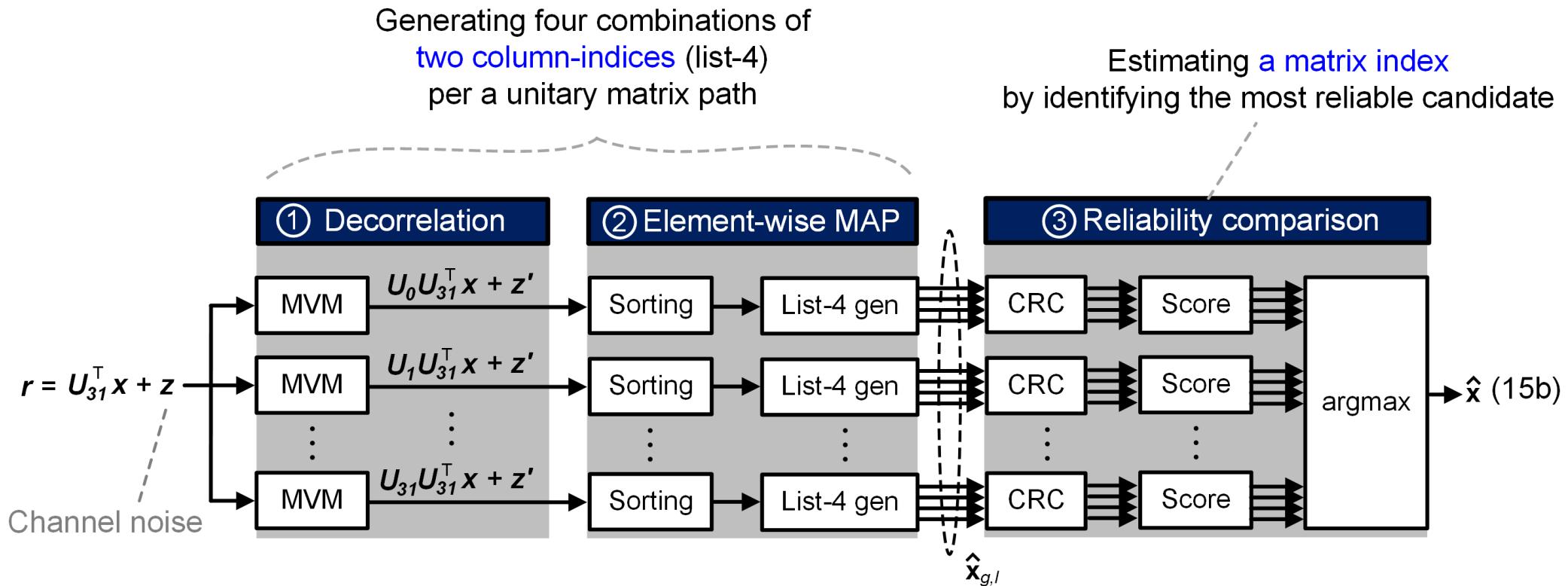
BOSS encoding



■ Index encoding

- Selecting a matrix and its two columns by using 18b as selection indices

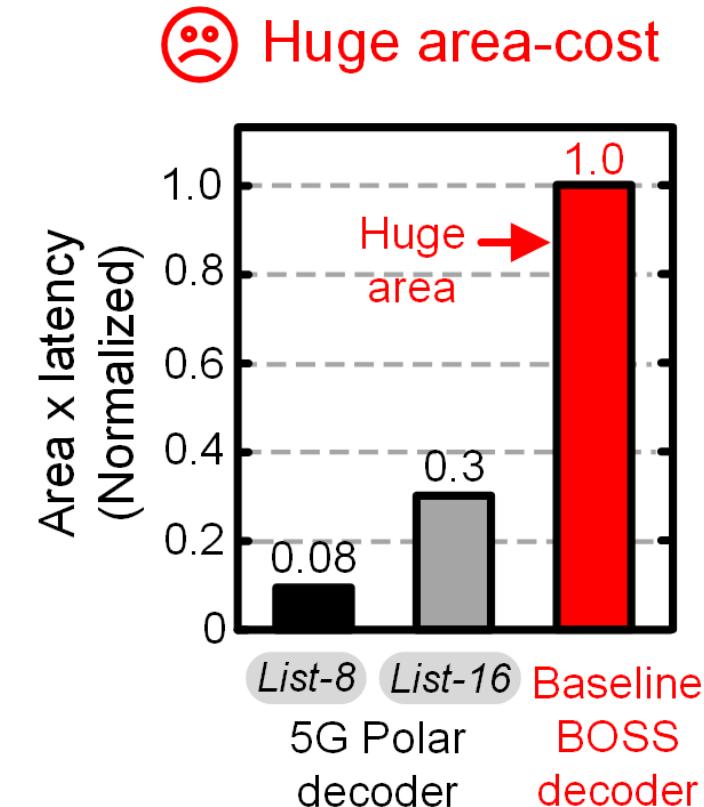
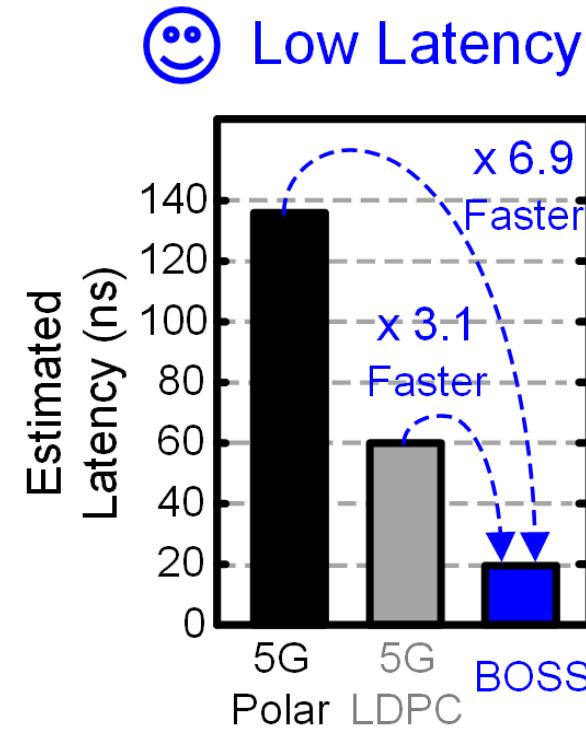
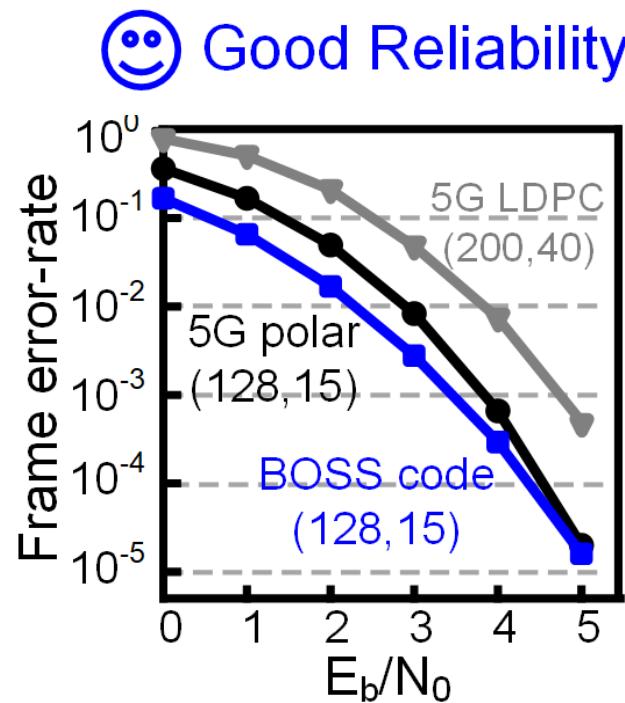
BOSS decoding



■ 3-stage decoding

- Estimating a matrix index and two column selection indices
- * MVM: Matrix-Vector Multiplication, MAP: Maximum a posteriori

Challenge of implementing a BOSS decoder

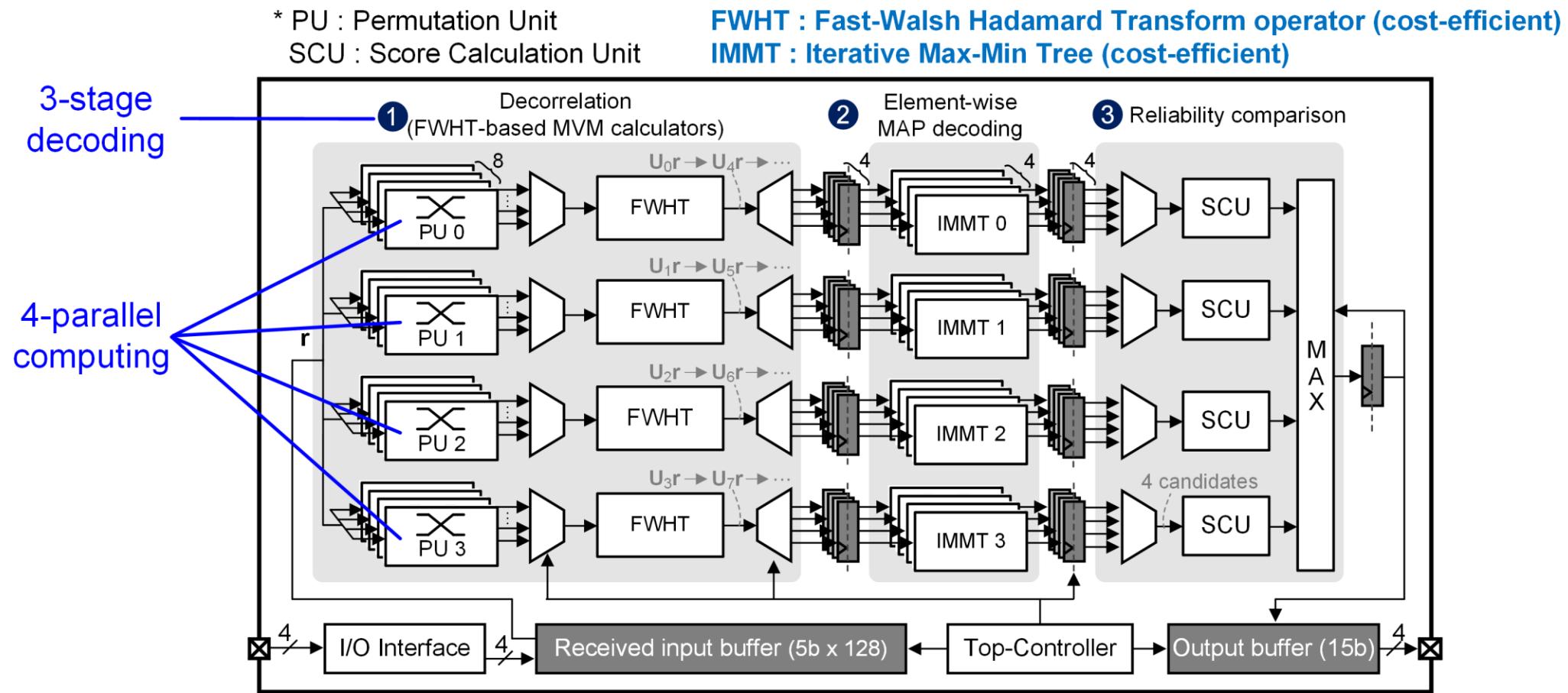


- A huge area-cost caused by matrix multipliers/sorters
- We introduce two cost-efficient circuits to reduce the area-cost.

Outline

- Introduction
- Backgrounds
 - BOSS encoding & decoding algorithm
- Proposed BOSS decoder
- Measurement Results
- Conclusion

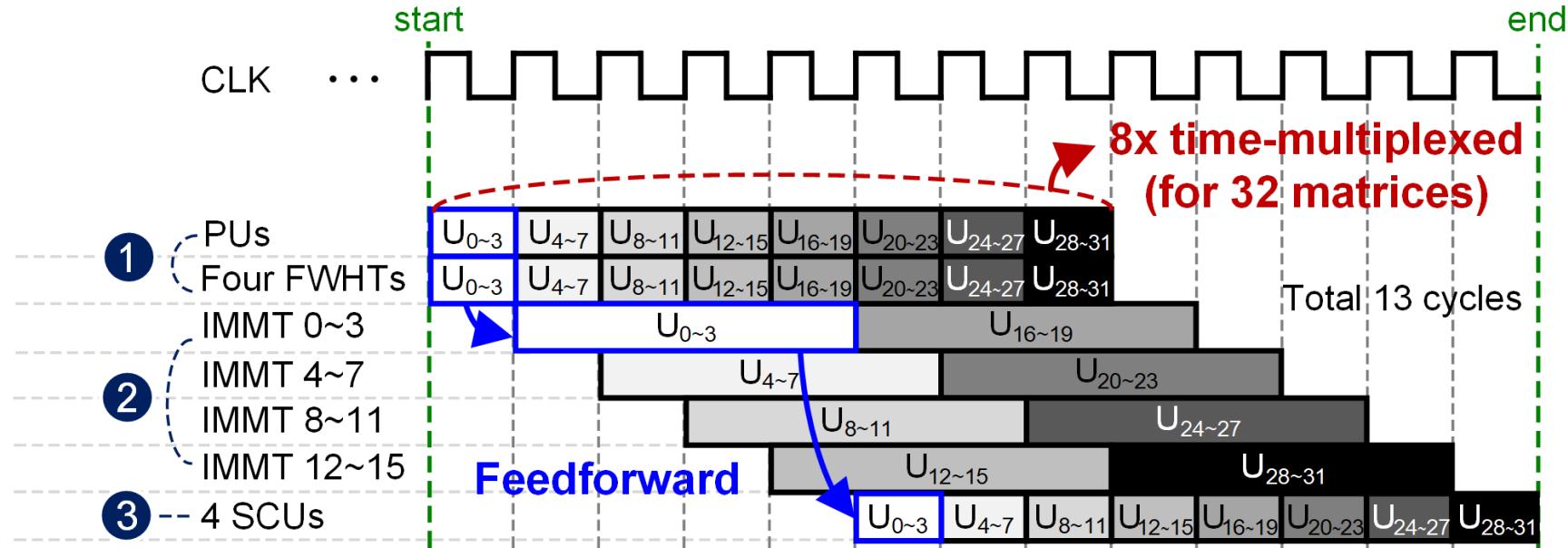
BOSS decoder: Overall architecture



■ 4-parallel architecture with 3 pipeline stages

- Two cost-efficient circuits : 1) FWHT-based MVM calculator, 2) IMMTs

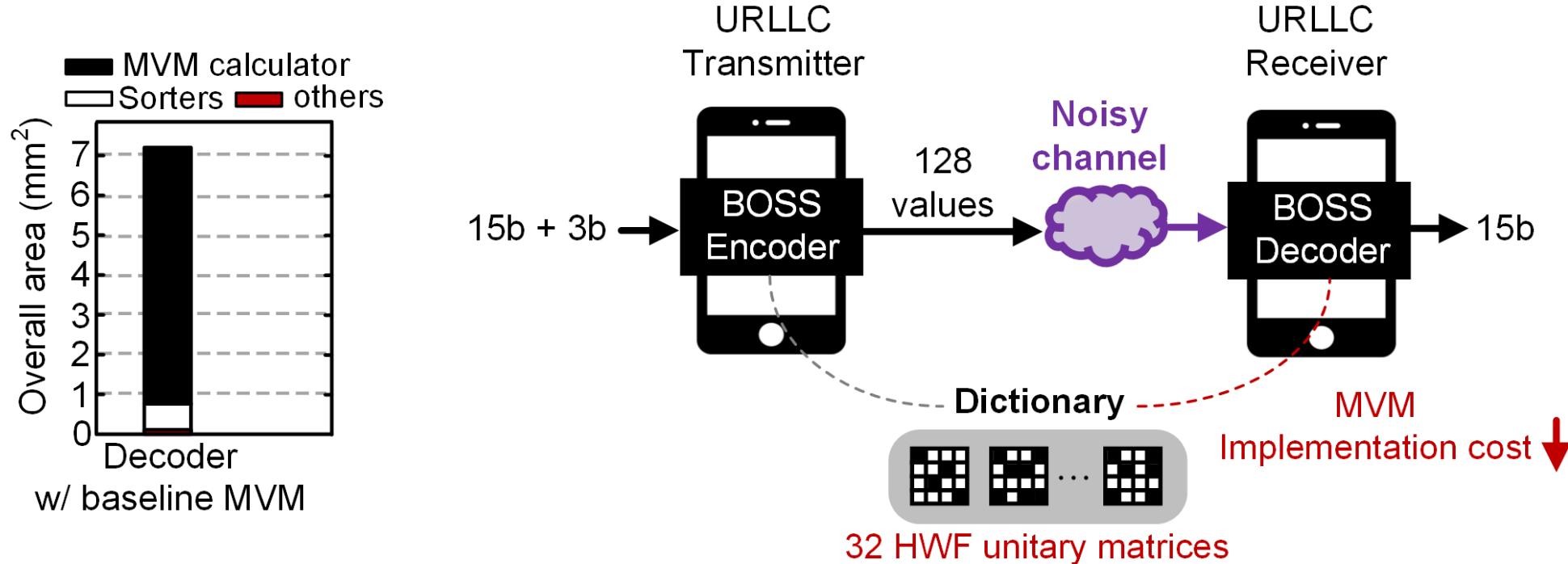
BOSS decoder: Processing flow



■ Processing flow

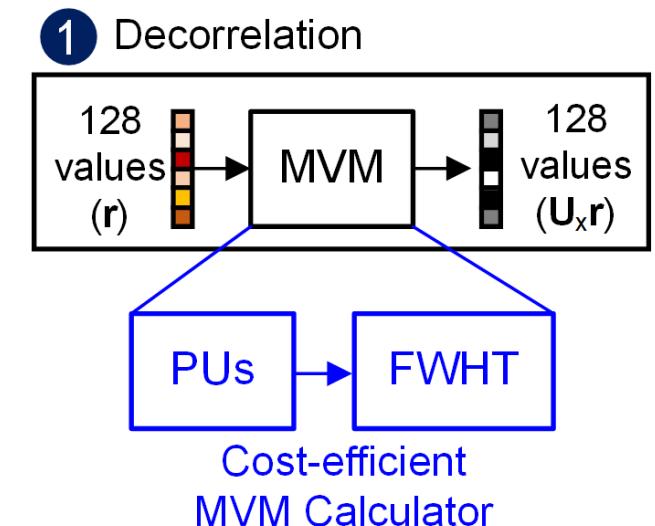
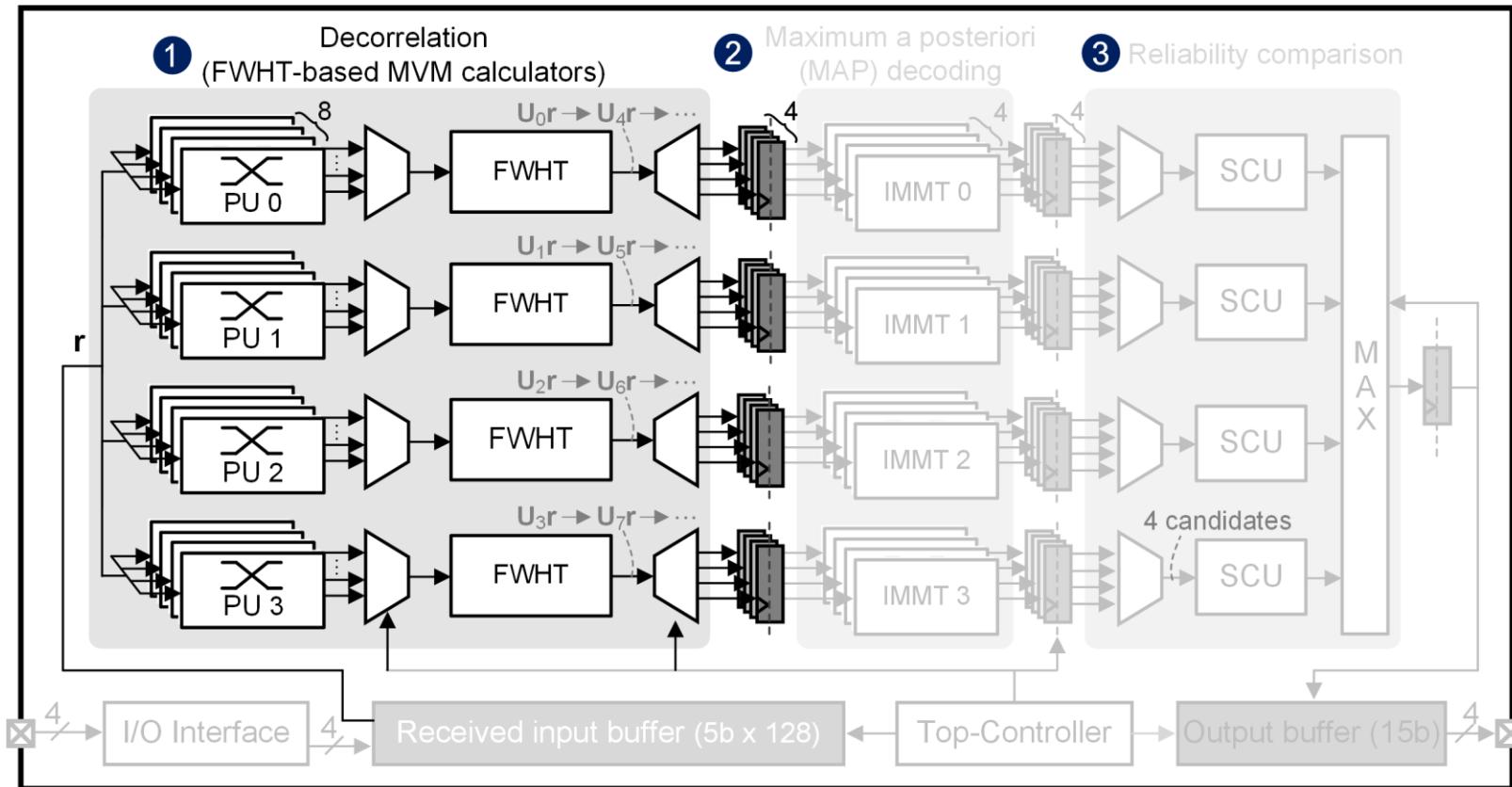
- It is time-multiplexed by 8x and feedforward.
- It takes 13 cycles to decode a (128, 15) BOSS codeword.

Cost-efficient circuit 1



- **Constructing hardware-friendly (HWF) unitary matrices**
 - Using HWF matrices reduces the implementation cost of BOSS decoder.

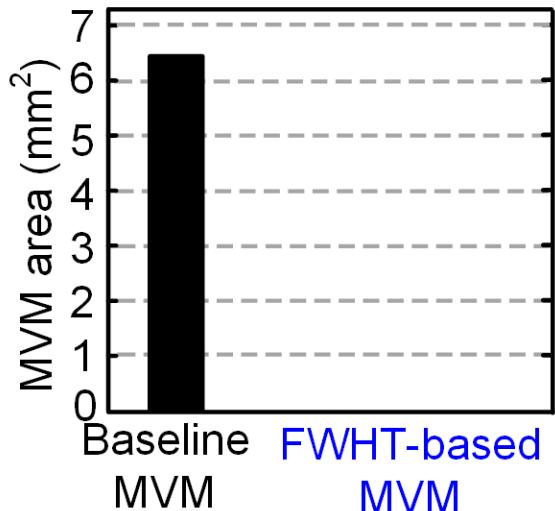
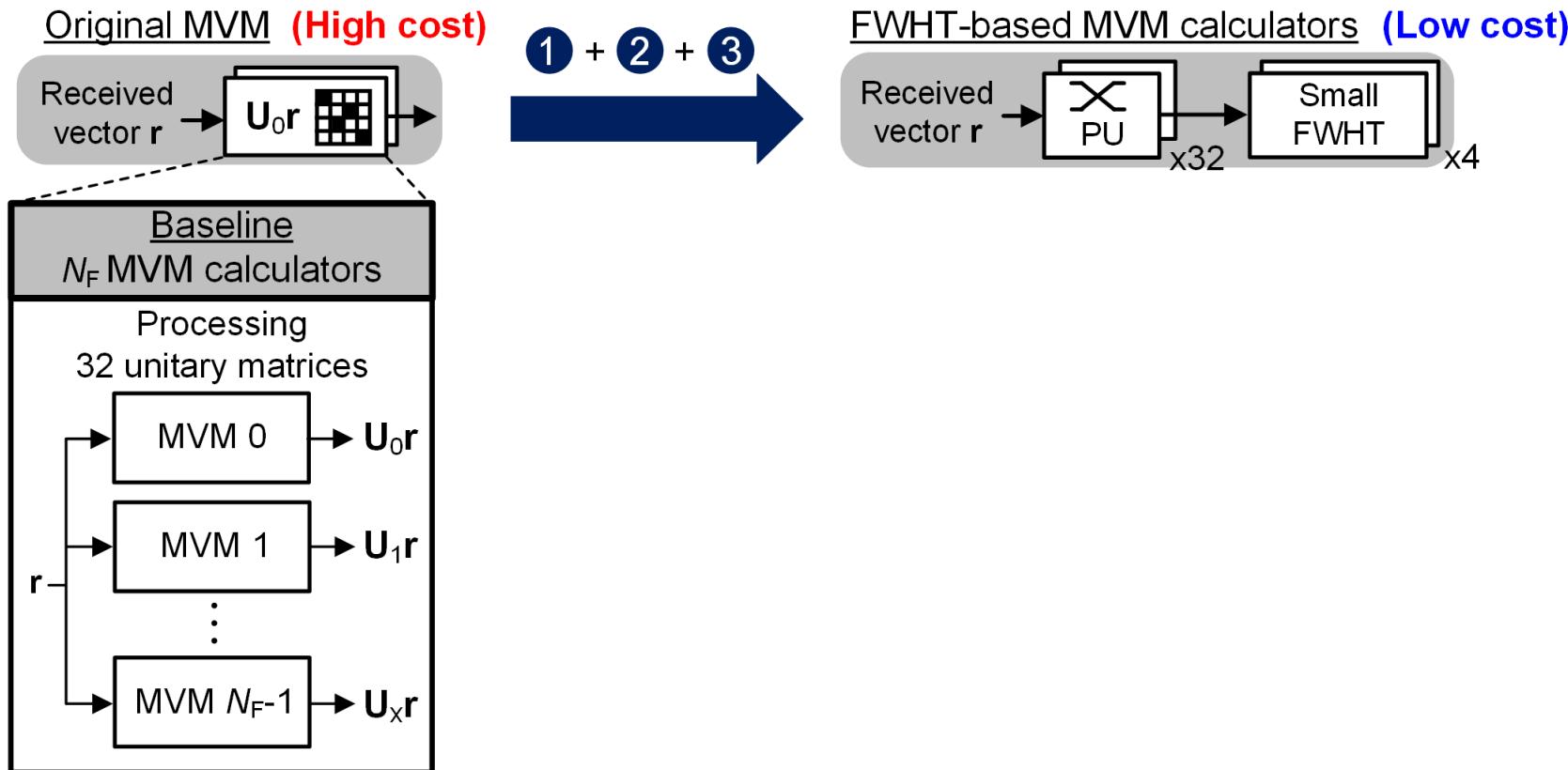
Cost-efficient circuit 1



■ Fast Walsh-Hadamard Transform-based MVM calculators

- consist of 32 PUs and 4 FWHT operators for 32 MVM tasks ($U_0r \sim U_{31}r$).

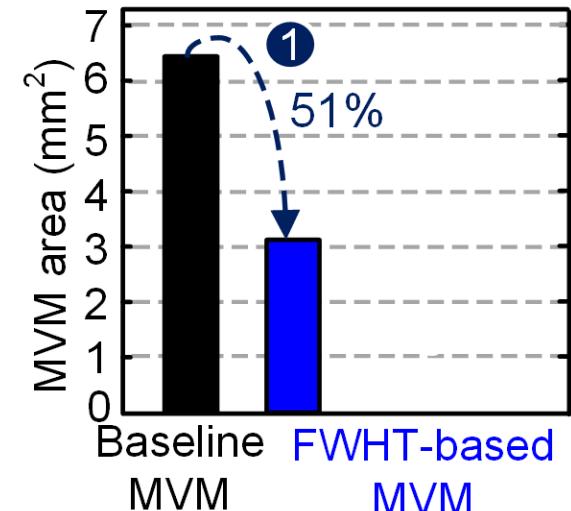
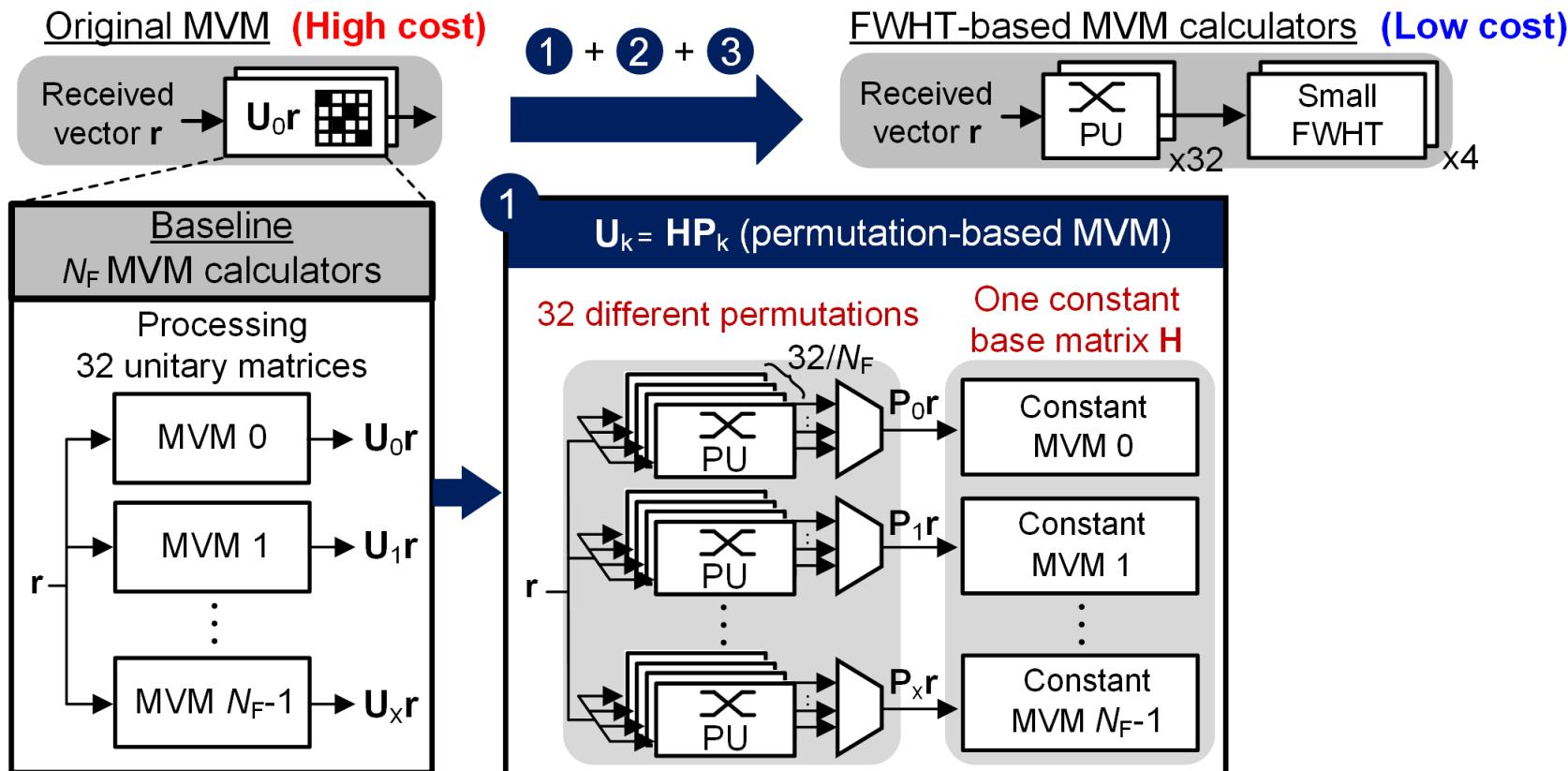
FWHT-based MVM Calculator



■ Baseline MVM Calculator

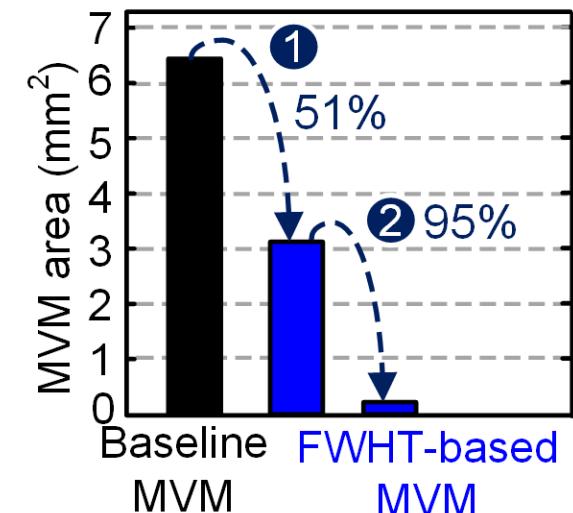
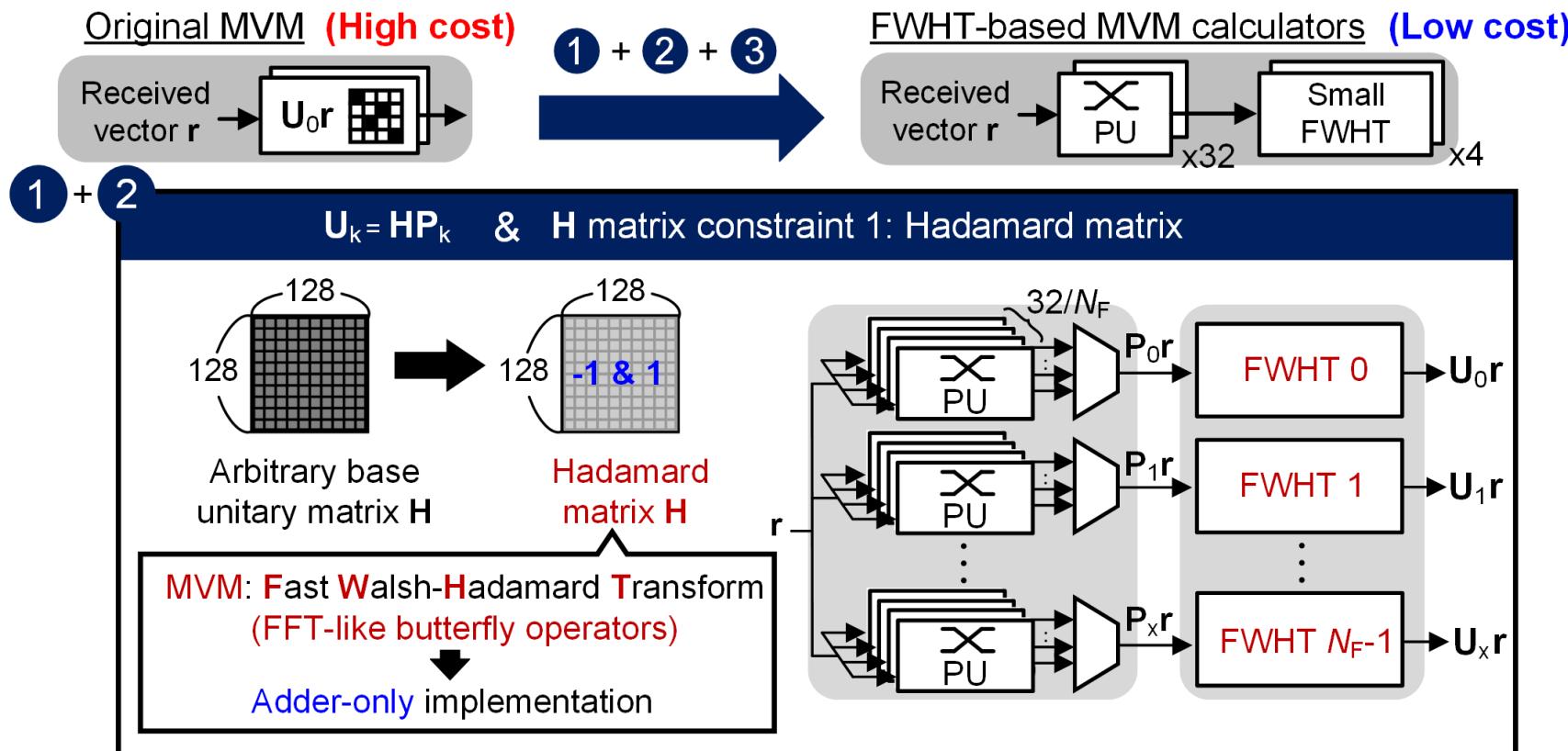
- N_F baseline MVM calculators support any matrix-vector multiplications.

FWHT-based MVM Calculator



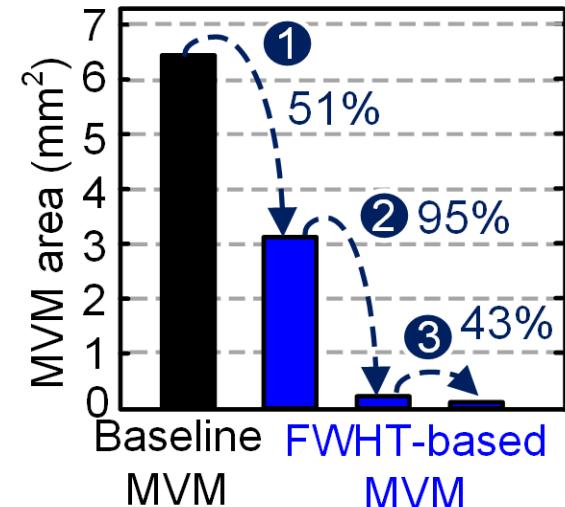
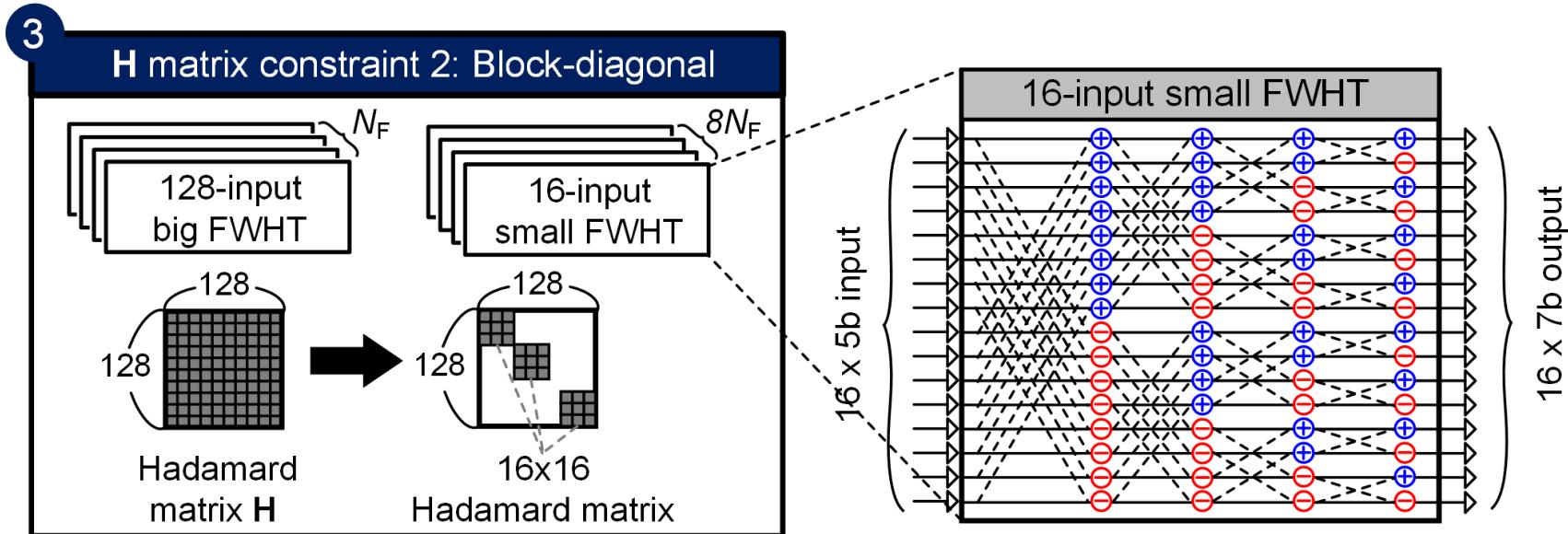
- **Method 1 : Row-permutation makes different unitary matrices**
 - A dictionary is constructed by a base unitary and 32 permutation matrices.

FWHT-based MVM Calculator



- Method 2 : Hadamard matrix is used as the base matrix.
 - A FWHT operator can be implemented only by adders.

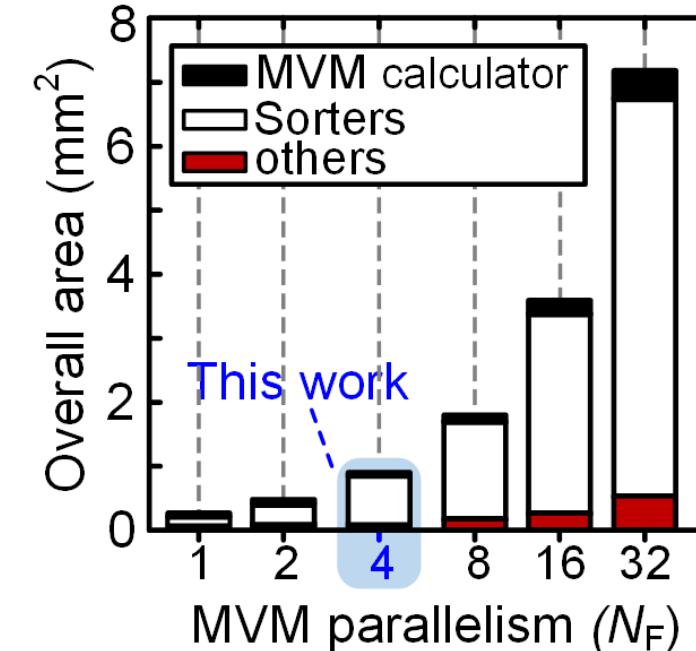
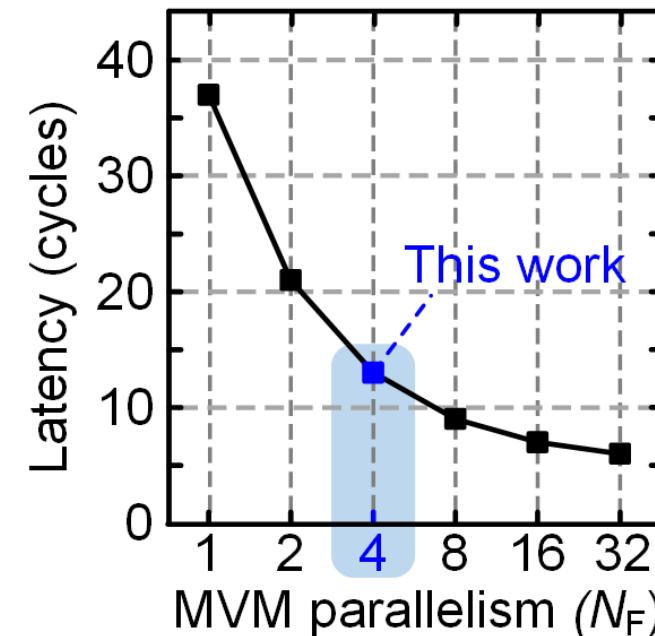
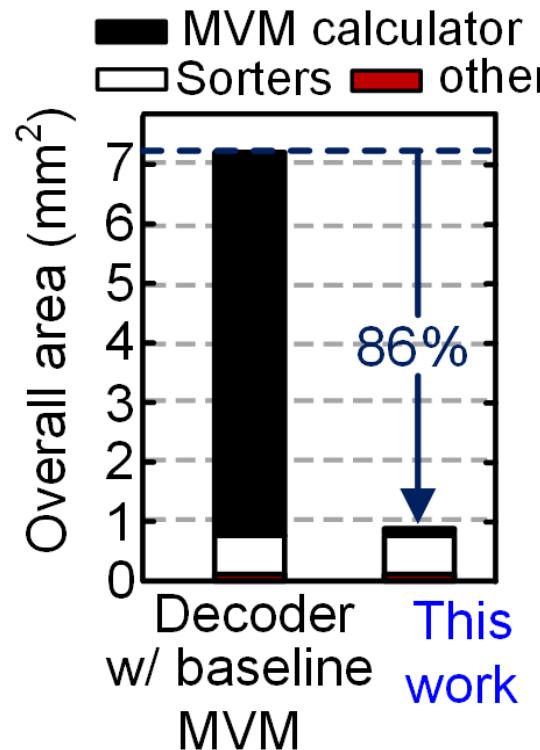
FWHT-based MVM Calculator



■ Method 3 : Block-diagonal approximation

- The block-diagonal Hadamard matrix reduces the size of FWHT operators.

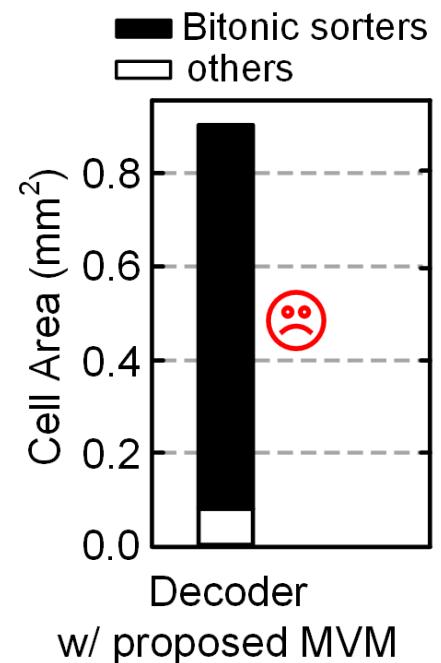
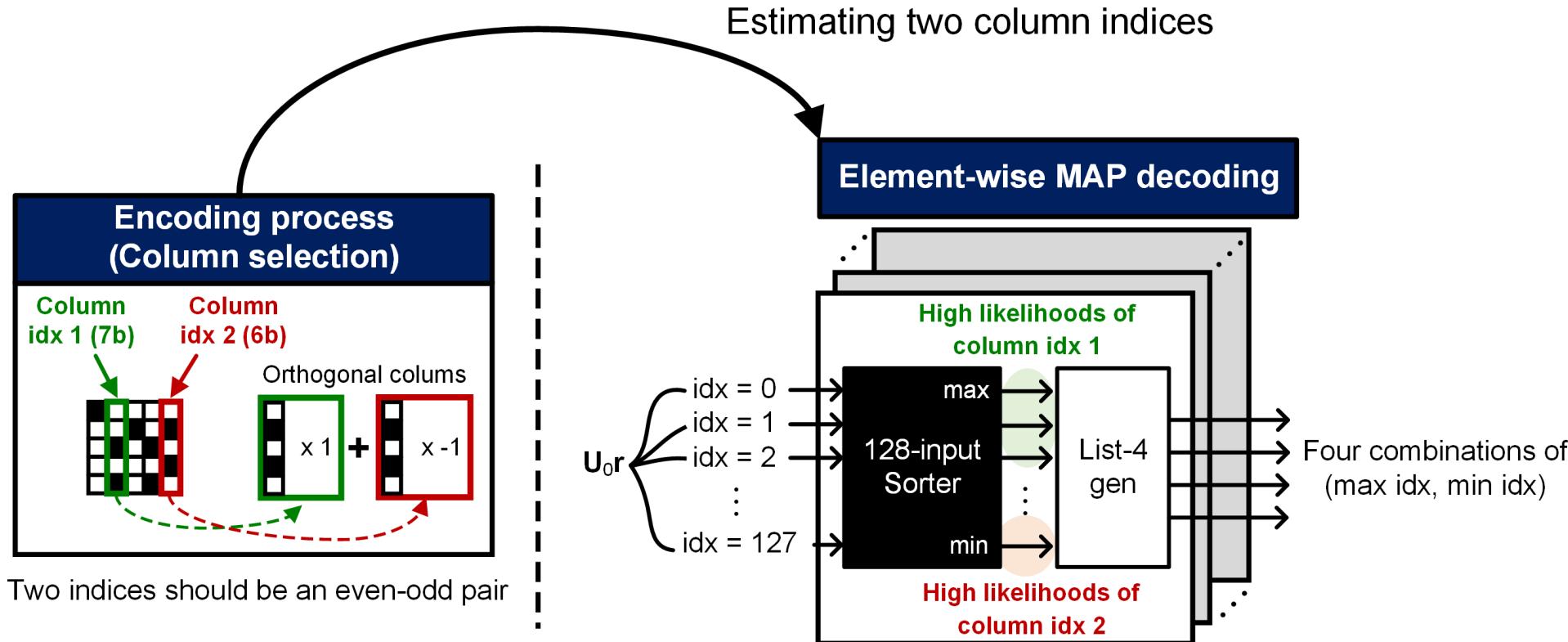
FWHT-based MVM Calculator



■ MVM parallelism

- 4-parallel MVM operators achieve good trade-off between latency and area.

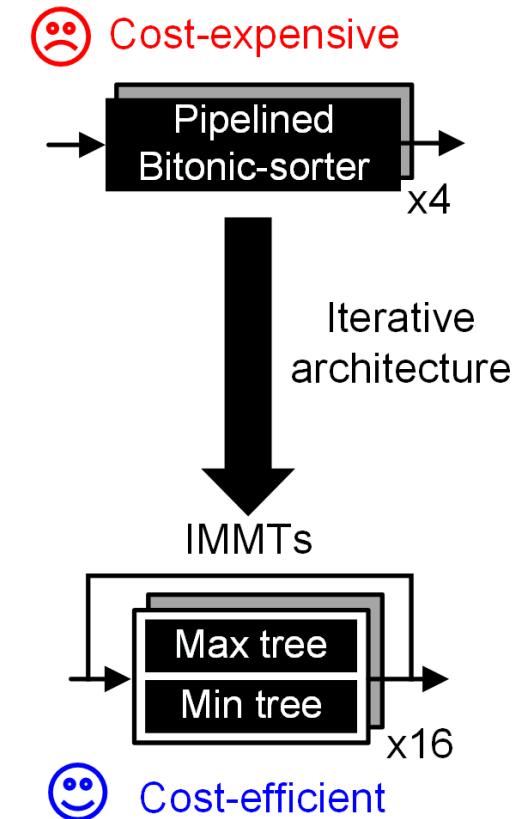
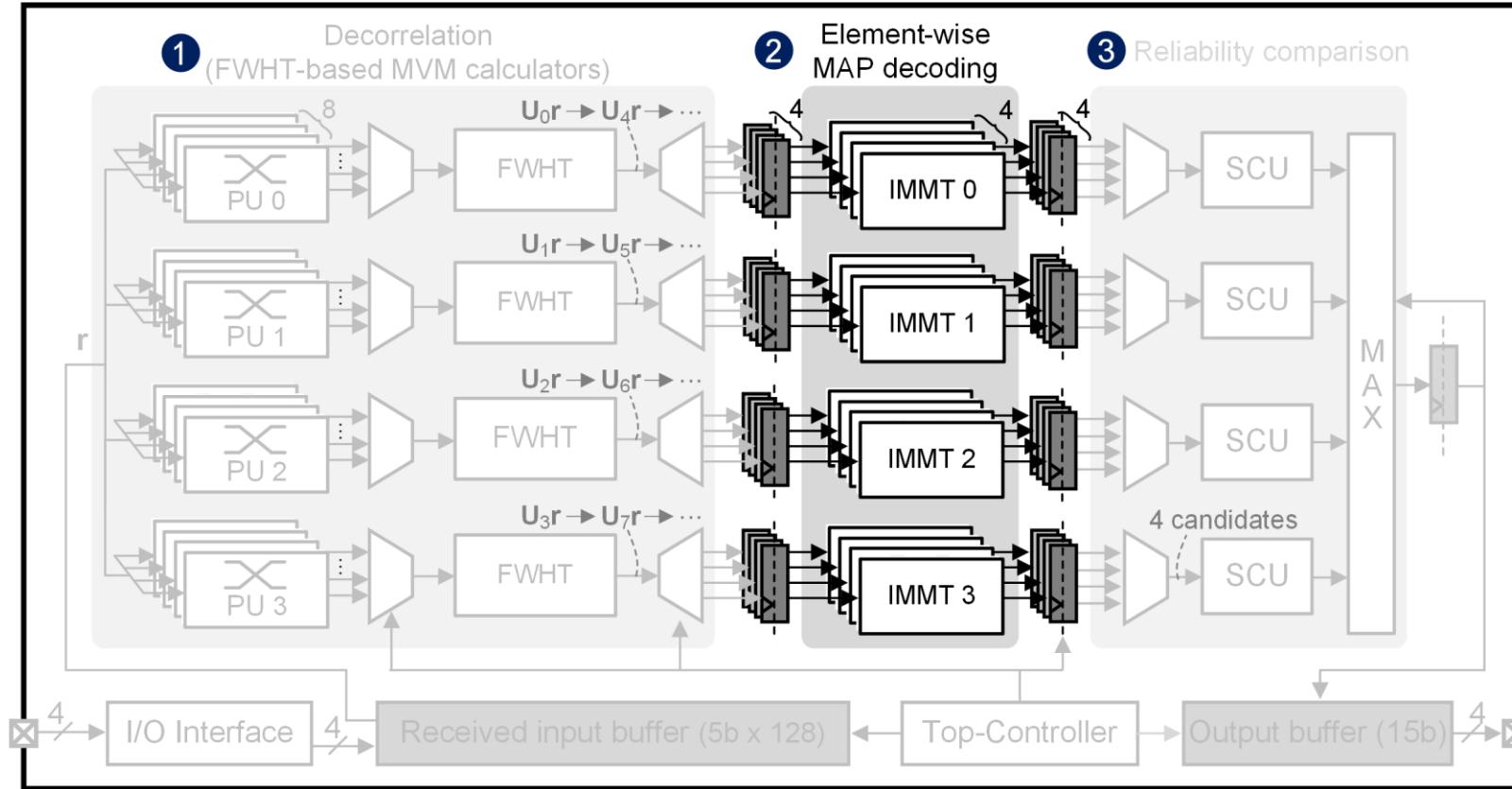
Cost-efficient circuit 2



■ Step 2 : Element-wise MAP decoding

- After optimizing MVM calculators, sorters occupy most of the total area.

Cost-efficient circuit 2

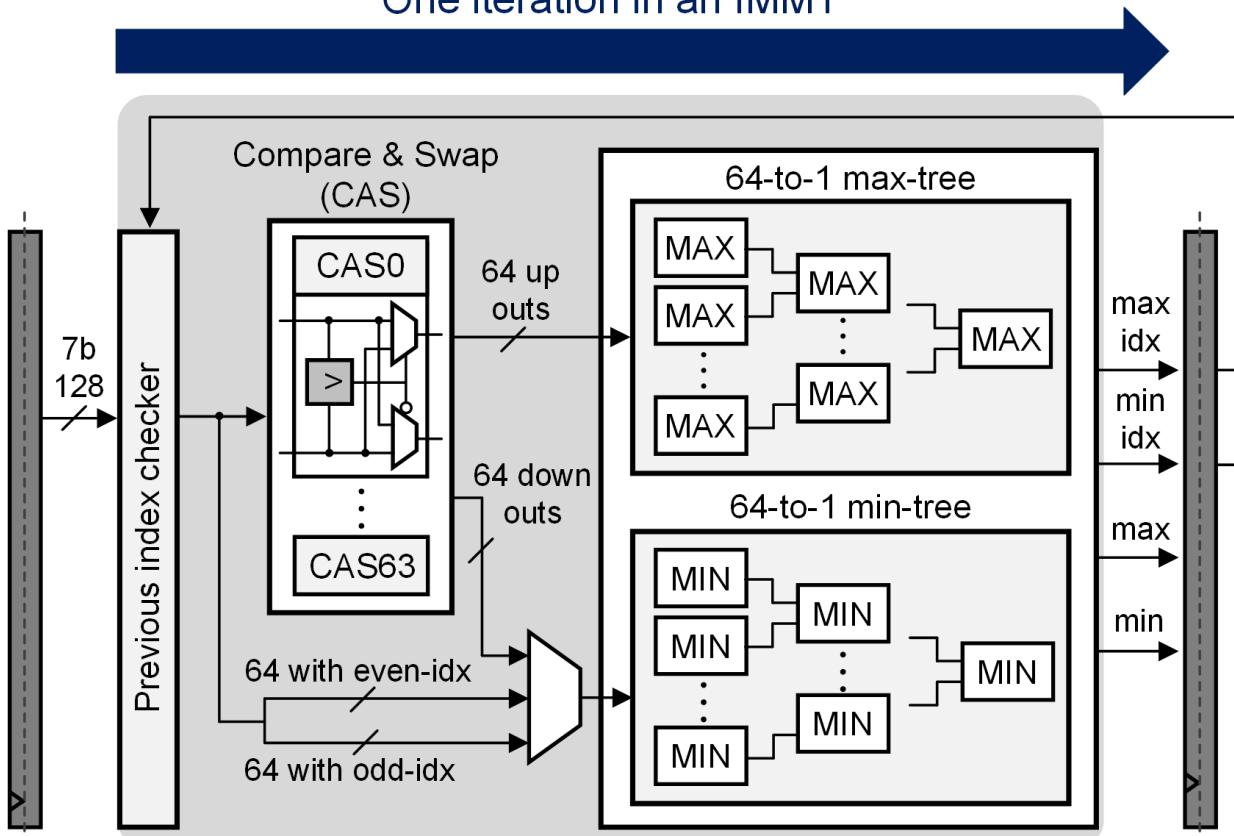


■ Iterative max-min tree (IMMT)

- An IMMT iteratively generates four combinations of two column indices.

Iterative Max-Min Tree

One iteration in an IMMT



IMMT Task :

Finding four (max idx, min idx) pairs in four cycles

Even-odd pair rule

Each pair (max idx, min idx) → (even, odd)
or
(odd, even)

Max-min search example

Cycle	max idx	min idx
1	even	even 0
2	odd	odd 0
3	x	odd 1
4	x	even 1

Four pairs

- (even, odd 0)
- (even, odd 1)
- (odd, even 0)
- (odd, even 1)

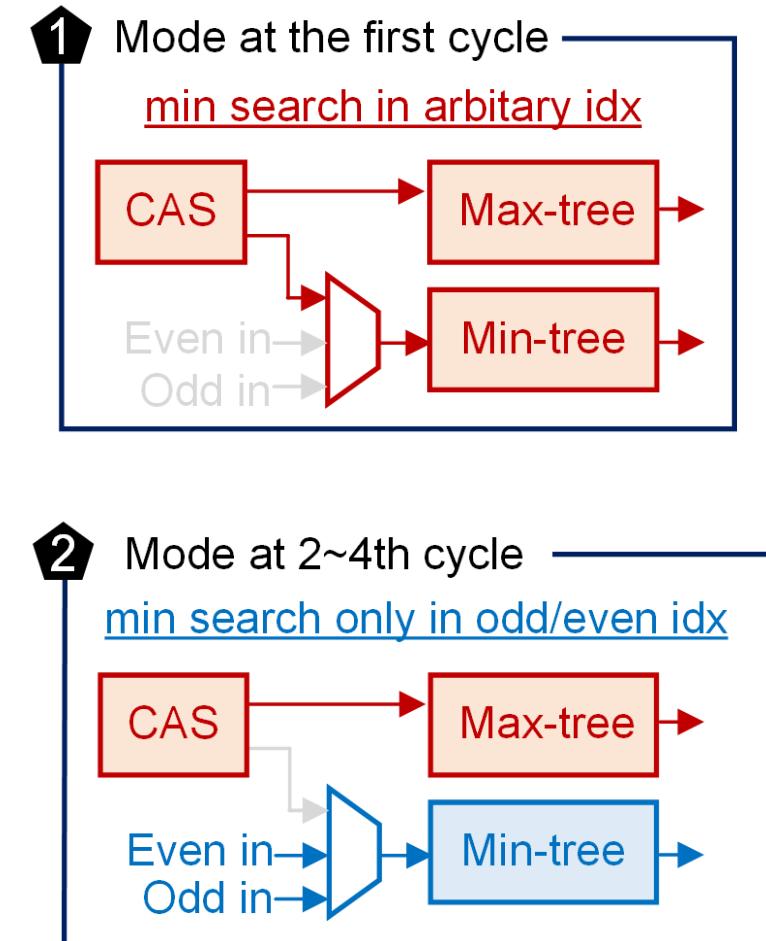
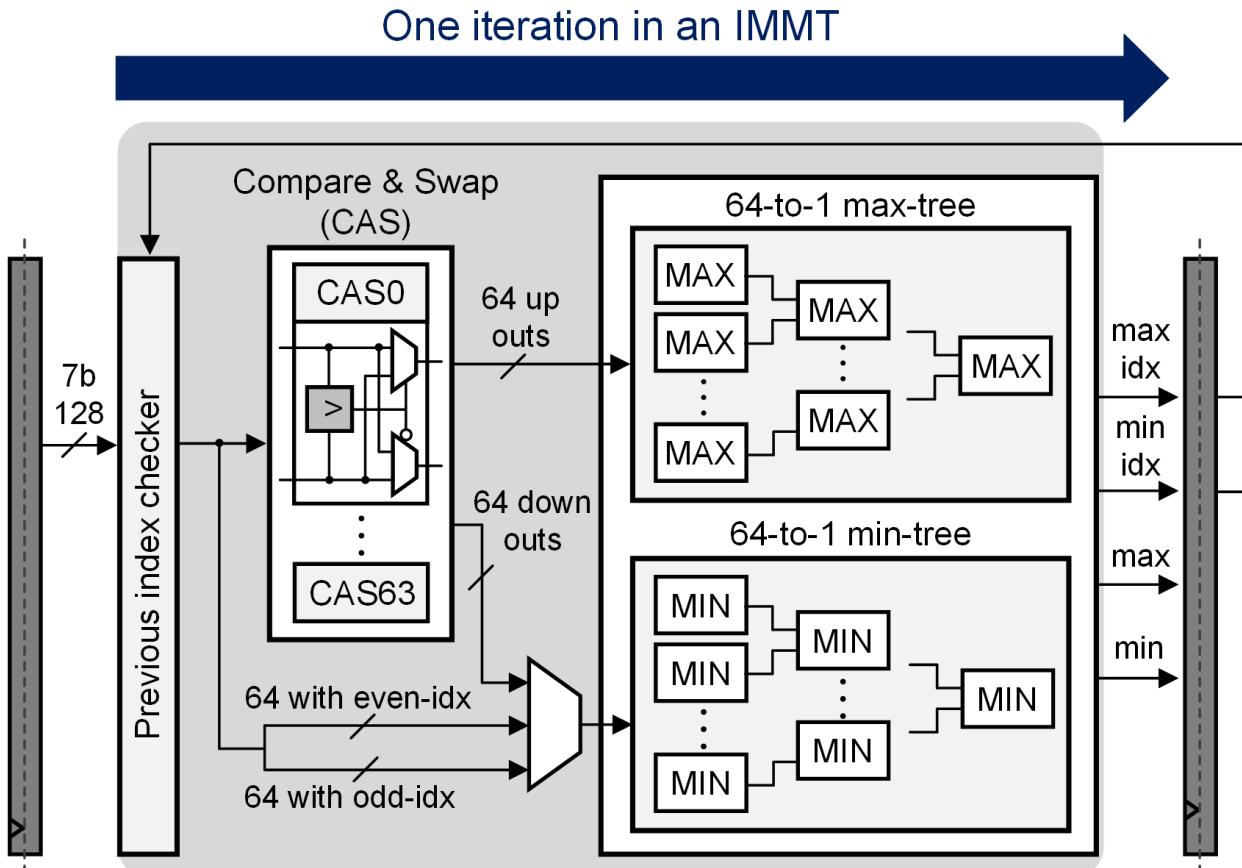
min-search

depending on the previous max-idx

■ Iterative computing architecture

- IMMT has a feedback path for the iterative searching for max-min pairs.

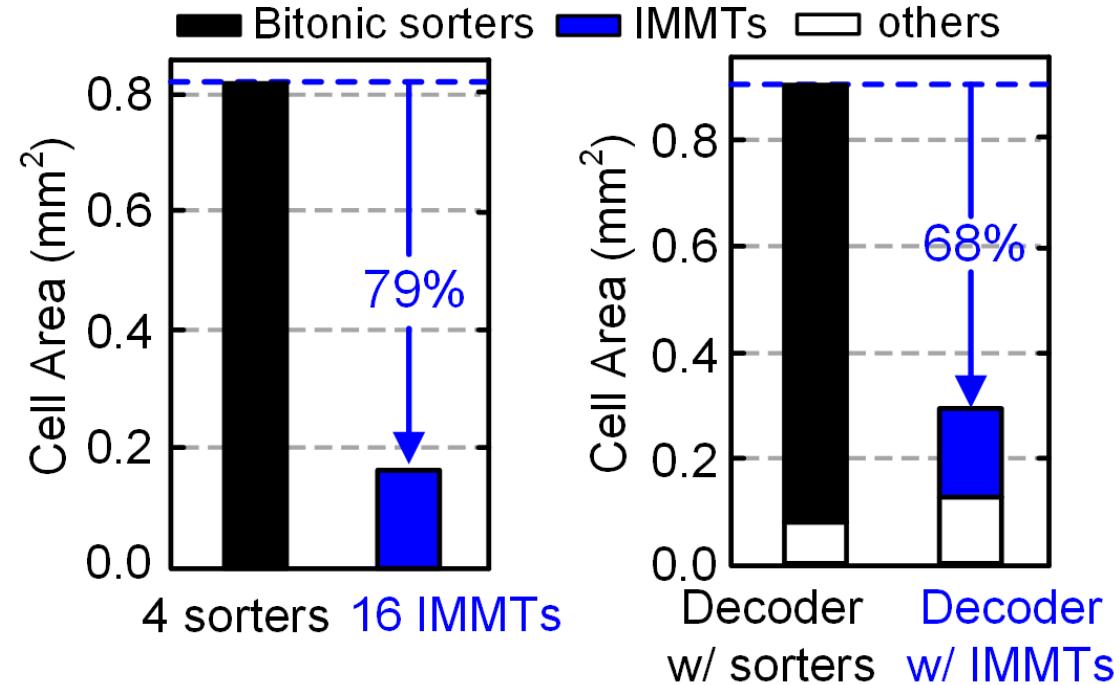
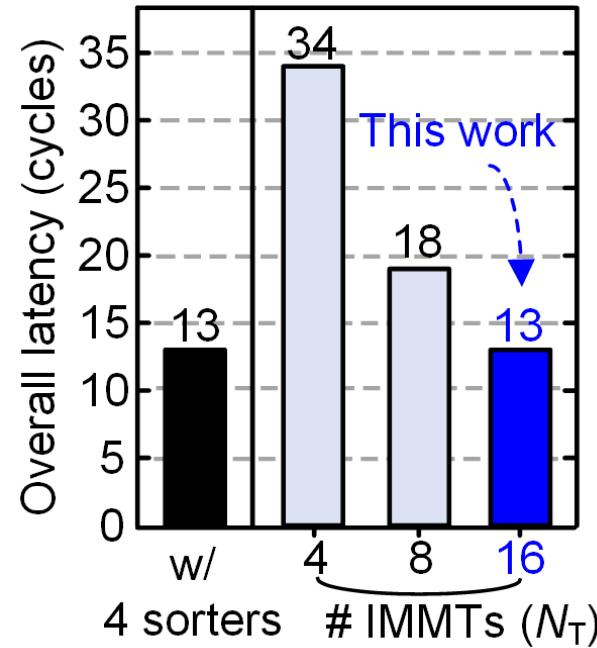
Iterative Max-Min Tree



■ Two modes of the min-tree

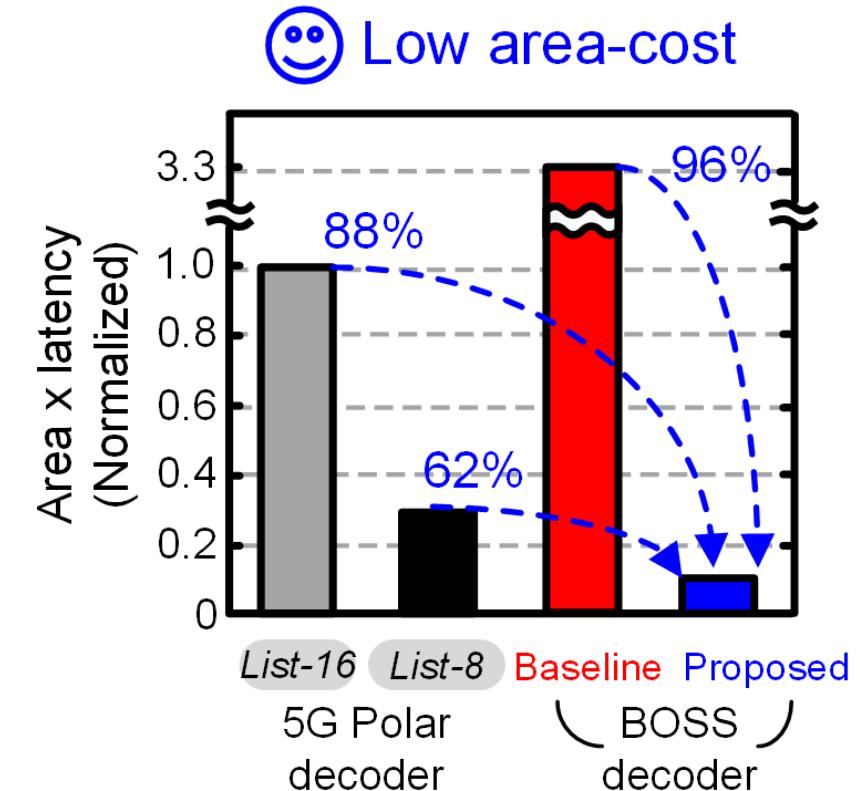
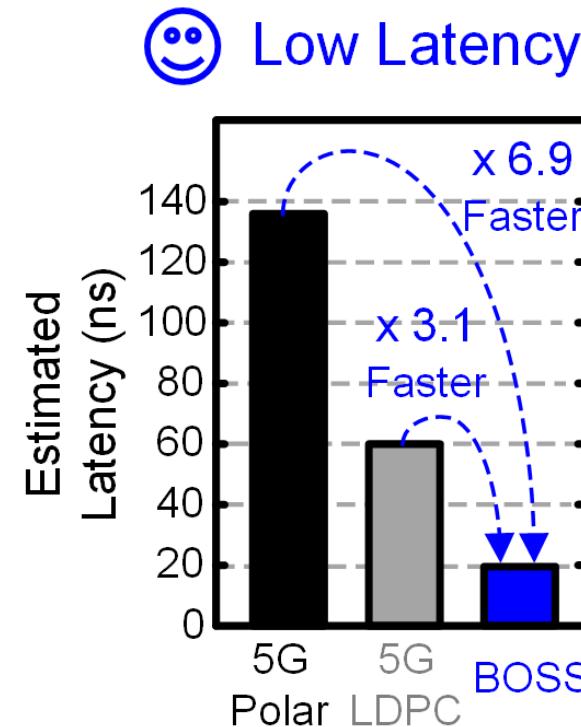
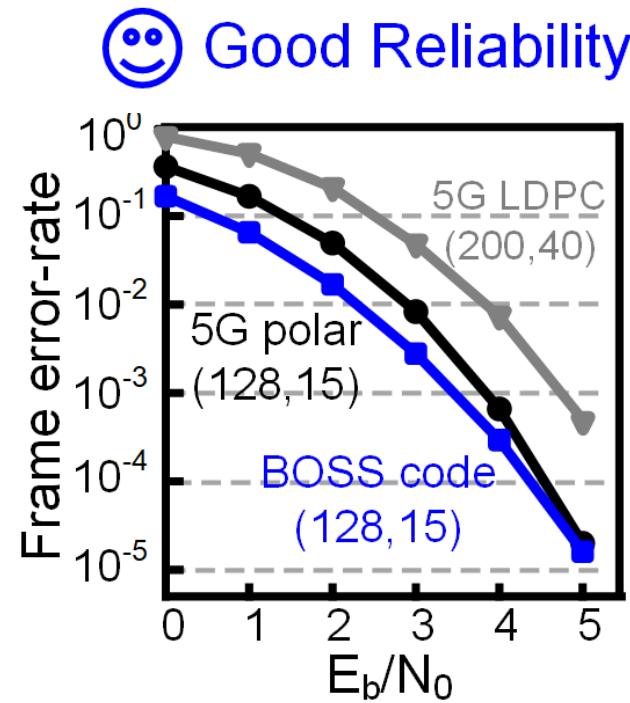
- Mode 1 is activated to search for a min value in arbitrary indices.
- Mode 2 is activated to search for a min value in even (or odd) indices.

Iterative Max-Min Tree



- Using 16 IMMTs achieve the same latency with a small area.

Proposed BOSS decoder

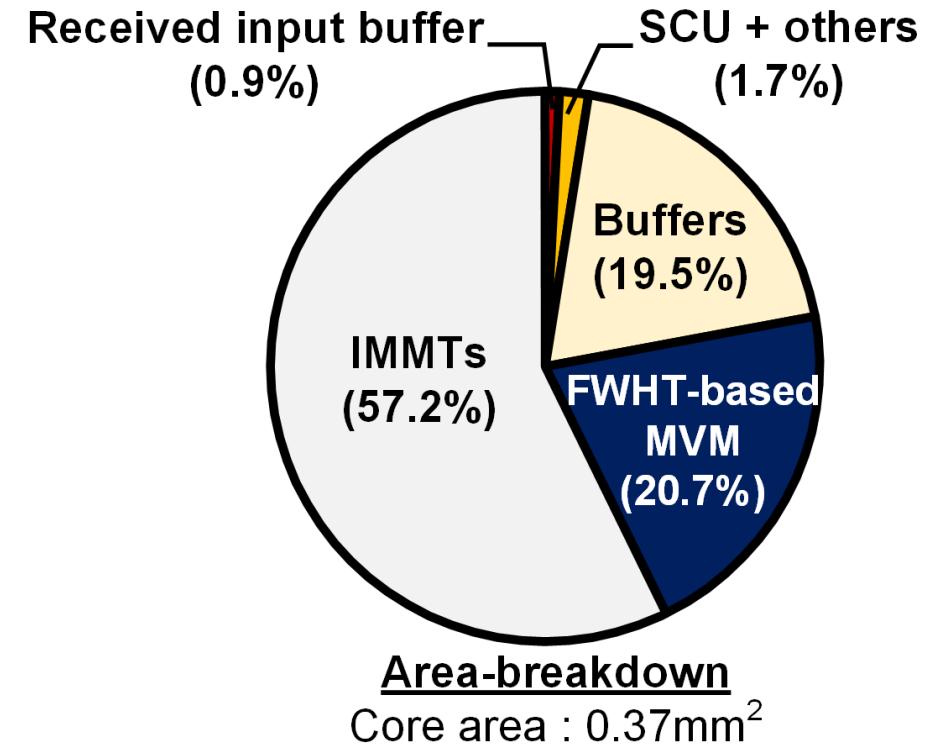
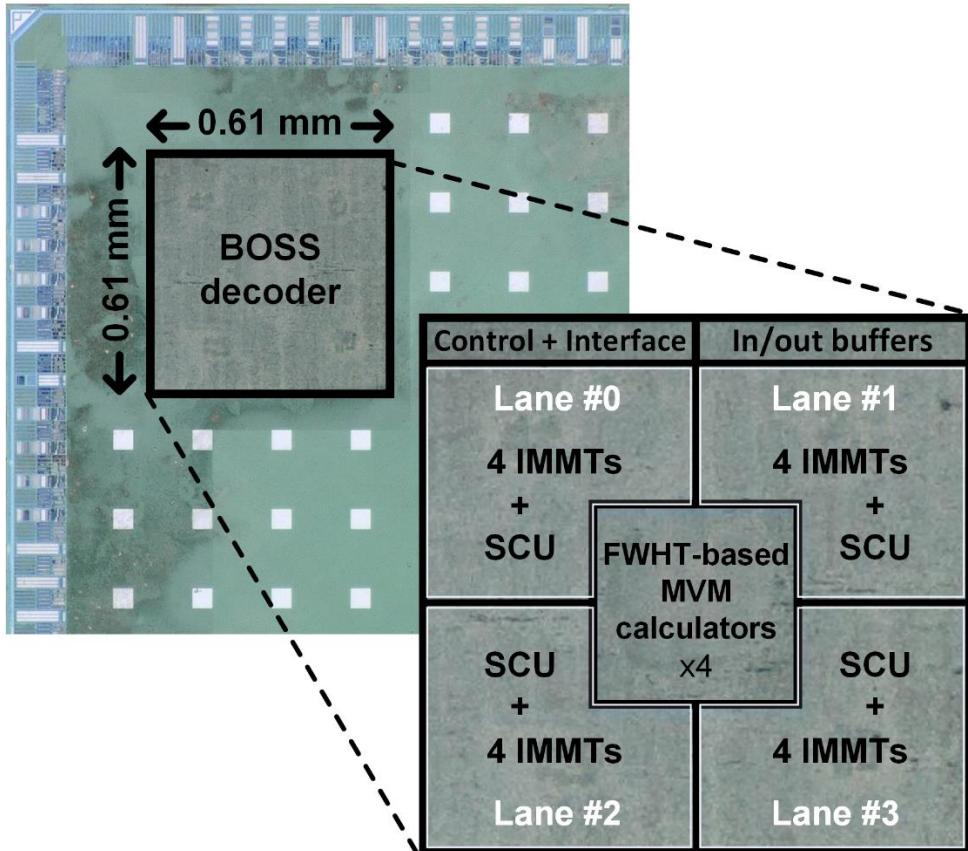


- Using two cost-efficient circuits (FWHT-based MVM, IMMT)
 - successfully achieves three URLLC requirements at the same time.

Outline

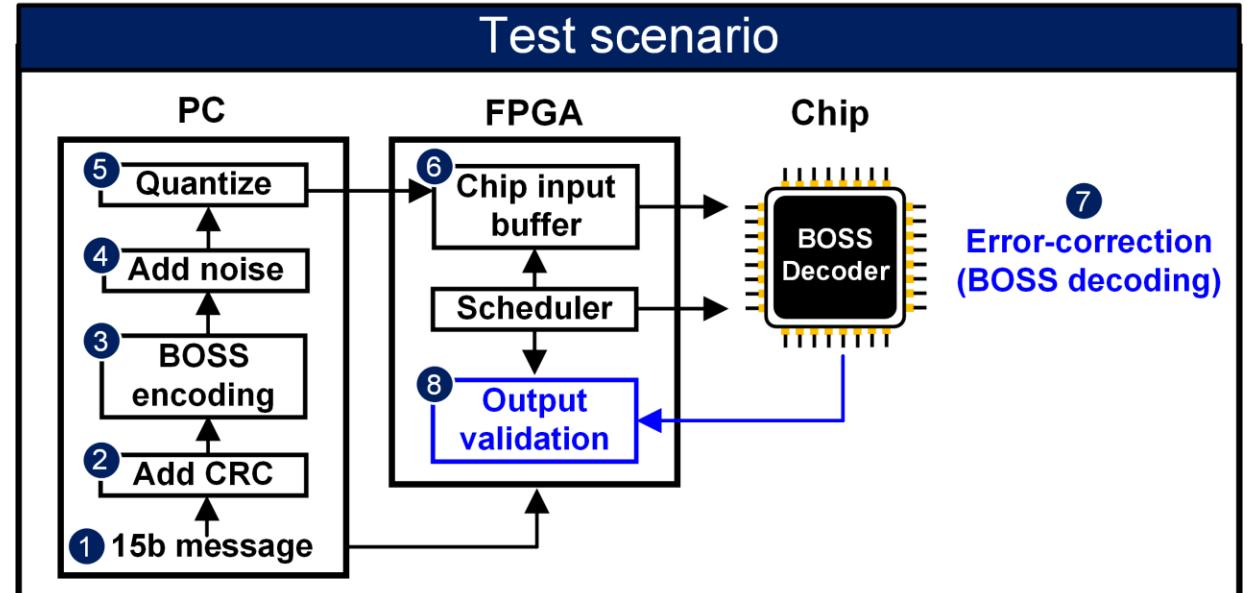
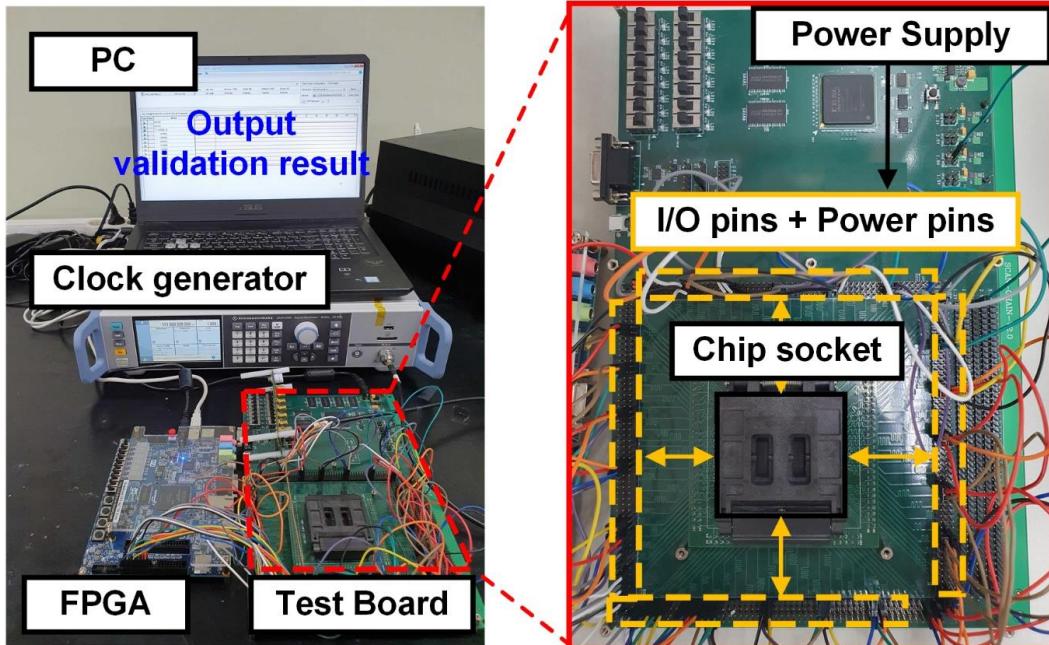
- Introduction
- Backgrounds
 - BOSS encoding & decoding algorithm
- Proposed BOSS decoder
- Measurement Results
- Conclusion

Chip Microphotograph



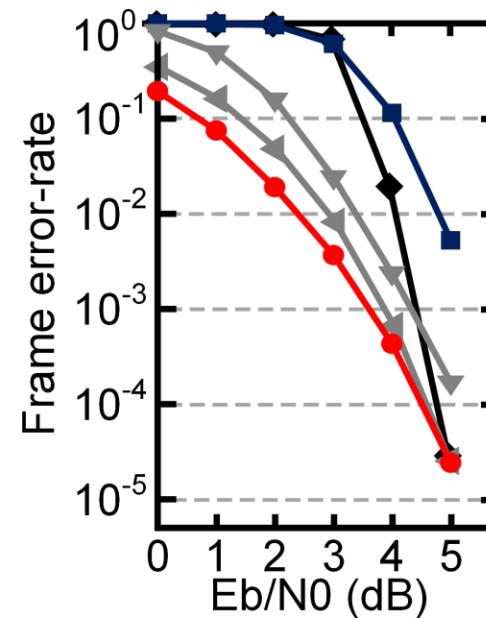
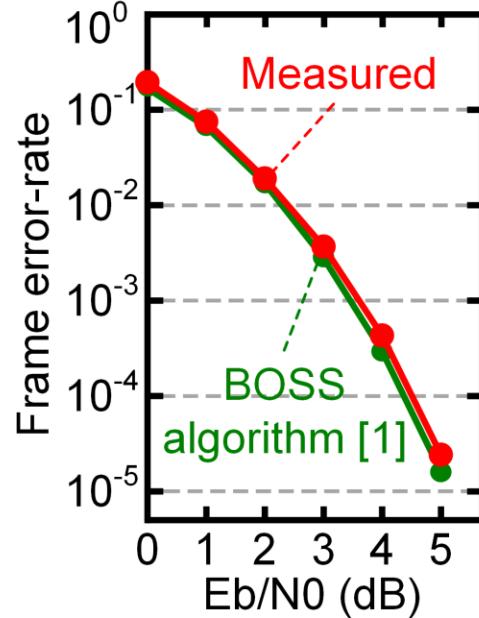
- The BOSS decoder was fabricated in 28nm CMOS

Chip Test Setup



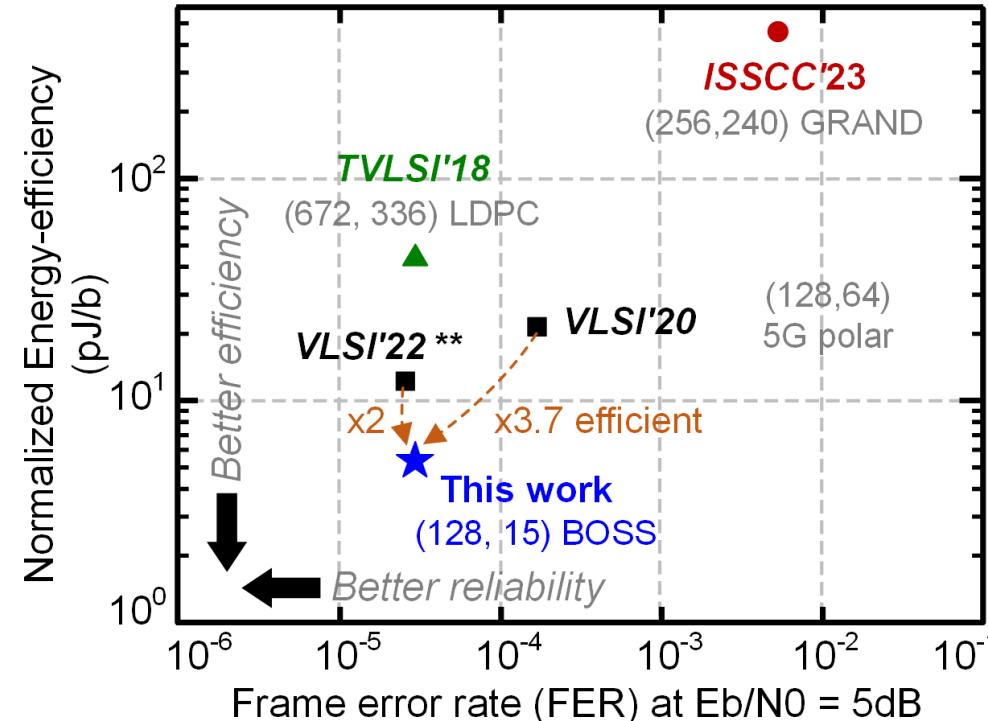
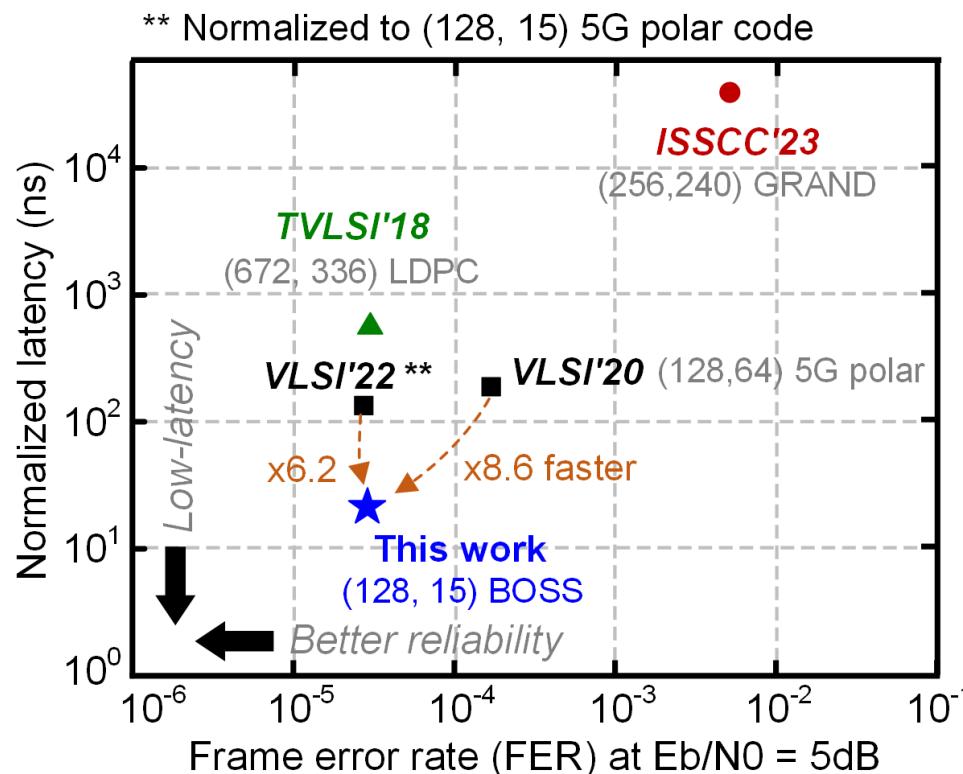
■ Our chip is tested on the FPGA-based verification scenario.

Error correction Performance (Reliability)



- The fabricated decoder achieves algorithm-level FERs.
 - The BOSS decoder outperforms recent FEC designs.
- * $Eb/N0$: Signal-to-noise power ratio per bit

Latency & Efficiency Comparison



- The BOSS decoder achieves the latency of 21.9ns at 0.95V.
- It consumes 33.3mW to decode a BOSS codeword.

Comparison Table

Strong reliability

Low area cost

Low latency

Good efficiency

	This work	ISSCC'23	VLSI'20			TVLSI'18	VLSI'22	JSSC'21
Technology	28nm	40nm	40nm			28nm	28nm	40nm
FEC code	BOSS code	GRAND + Polar	Polar code			LDPC	Polar code	Polar code
Decoding Algorithm	BOSS dec.	ORBDRAND	RNN-BP			MS	CA-SCL	CA-SCL
Code length	128	256	256	128	64	672	1024	1024
Eb/N0 @ FER=10 ⁻⁵	5.2	7	5.2	5.8	6.3	5.2	2.75	3.6
Core area (mm ²)	0.37	0.4	0.18			1.99	0.595	0.317
Supply Voltage (V)	0.95	1.0	0.9			0.9	1.05	0.9
Power (mW)	33.3	4.8	12.8			399	101.4	42.8
Constant latency	O	X	O			X	O	O
Normalized to 28nm, 0.95V [‡]								
Latency (ns)	21.9	41990*	42.9**	217	189	154	555	1100
Throughput (Gbps)	5.84	0.006*	6.14**	1.17	0.67	0.41	9.69	0.93
Area-efficiency (Gbps/mm ²)	15.78	0.031*	31.3**	13.3	7.6	4.7	3.41	1.56
Energy-efficiency (pJ/b)	5.7	497.4*	0.50**	12.1	21.1	34.3	45.90	89.64
* is estimated at Eb/N0 = 5 dB ‡ Norm. Latency= Latency * X ** is measured at Eb/N0 = 7 dB Norm. Area-effi. = Area-effi. / X ³ Norm. Throughput = Throughput / X † is measured for decoding a frame where X = 28nm / used tech. , Y = 0.95V / used voltage Norm. Energy-effi.= Energy-effi. * X ² Y ²								

* is estimated at Eb/N0 = 5 dB

** is measured at Eb/N0 = 7 dB

† is measured for decoding a frame

‡ Norm. Latency= Latency * X

Norm. Area-effi. = Area-effi. / X³

where X = 28nm / used tech. , Y = 0.95V / used voltage

Norm. Throughput = Throughput / X

Norm. Energy-effi.= Energy-effi. * X²Y²

Comparison Table

	This work	ISSCC'23		VLSI'20			TVLSI'18	VLSI'22	JSSC'21
Technology	28nm	40nm		40nm			28nm	28nm	40nm
FEC code	BOSS code	GRAND + Polar			Polar code		LDPC	Polar code	Polar code
Decoding Algorithm	BOSS dec.	ORBDRAND			RNN-BP		MS	CA-SCL	CA-SCL
Code length	128	256		256	128	64	672	1024	1024
Eb/N0 @ FER=10 ⁻⁵	5.2	7		5.2	5.8	6.3	5.2	2.75	3.6
Core area (mm ²)	0.37	The Lower Eb/N0, the stronger reliability					0.595	0.317	
Supply Voltage (V)	0.95						1.05	0.9	
Power (mW)	33.3	4.8		12.8		399	101.4	42.8	
Constant latency	O	X		O		X	O	O	
Normalized to 28nm, 0.95V [‡]									
Latency (ns)	21.9	41990*	42.9**	217	189	154	555	1100	1582
Throughput (Gbps)	5.84	0.006*	6.14**	1.17	0.67	0.41	9.69	0.93	0.65
The BOSS decoder is highly suitable for Ultra-Reliable Low-latency Communications.									
Energy-efficiency (pJ/b)	5.7	497.4*	0.50**	12.1	21.1	34.3	45.90	89.64	49.08

* is estimated at Eb/N0 = 5 dB

‡ Norm. Latency = Latency * X

Norm. Throughput = Throughput / X

** is measured at Eb/N0 = 7 dB

Norm. Area-effi. = Area-effi. / X³

Norm. Energy-effi.= Energy-effi. * X²Y²

† is measured for decoding a frame

where X = 28nm / used tech. , Y = 0.95V / used voltage

Comparison Table

	This work	ISSCC'23	VLSI'20			TVLSI'18	VLSI'22	JSSC'21
Technology	28nm	40nm	40nm			28nm	28nm	40nm
FEC code	BOSS code	GRAND + Polar	Polar code			LDPC	Polar code	Polar code
Decoding Algorithm	BOSS dec.	ORBDRAND	RNN-BP			MS	CA-SCL	CA-SCL
Code length	128	256	256	128	64	672	1024	1024
Eb/N0 @ FER=10 ⁻⁵	5.2	7	5.2	5.8	6.3	5.2	2.75	3.6
Core area (mm ²)	0.37	0.4	0.18			1.99	0.595	0.317
Supply Voltage (V)	0.95	1.0	0.95	0.95	0.95	1.25	1.25	0.9
Power (mW)	~100	~100	~100	~100	~100	~100	~100	42.8
Constant latency	~100ns	~100ns	~100ns	~100ns	~100ns	~100ns	~100ns	0
	Normalized to 28nm, 0.95V [†]							
Latency (ns)	~100	~100	~100	~100	~100	~100	~100	1582
Throughput (Gbps)	~100	~100	~100	~100	~100	~100	~100	0.65
Area-efficiency (Gbps/mm ²)	15.78	0.031 [*]	31.3 ^{**}	13.3	7.6	4.7	3.41	1.56
Energy-efficiency (pJ/b)	5.7	497.4 [*]	0.50 ^{**}	12.1	21.1	34.3	45.90	89.64

Due to two cost-efficient circuits,
the BOSS decoder is designed with a small area

It also achieves the good area-efficiency and energy-efficiency.

* is estimated at Eb/N0 = 5 dB

† Norm. Latency= Latency * X

Norm. Throughput = Throughput / X

** is measured at Eb/N0 = 7 dB

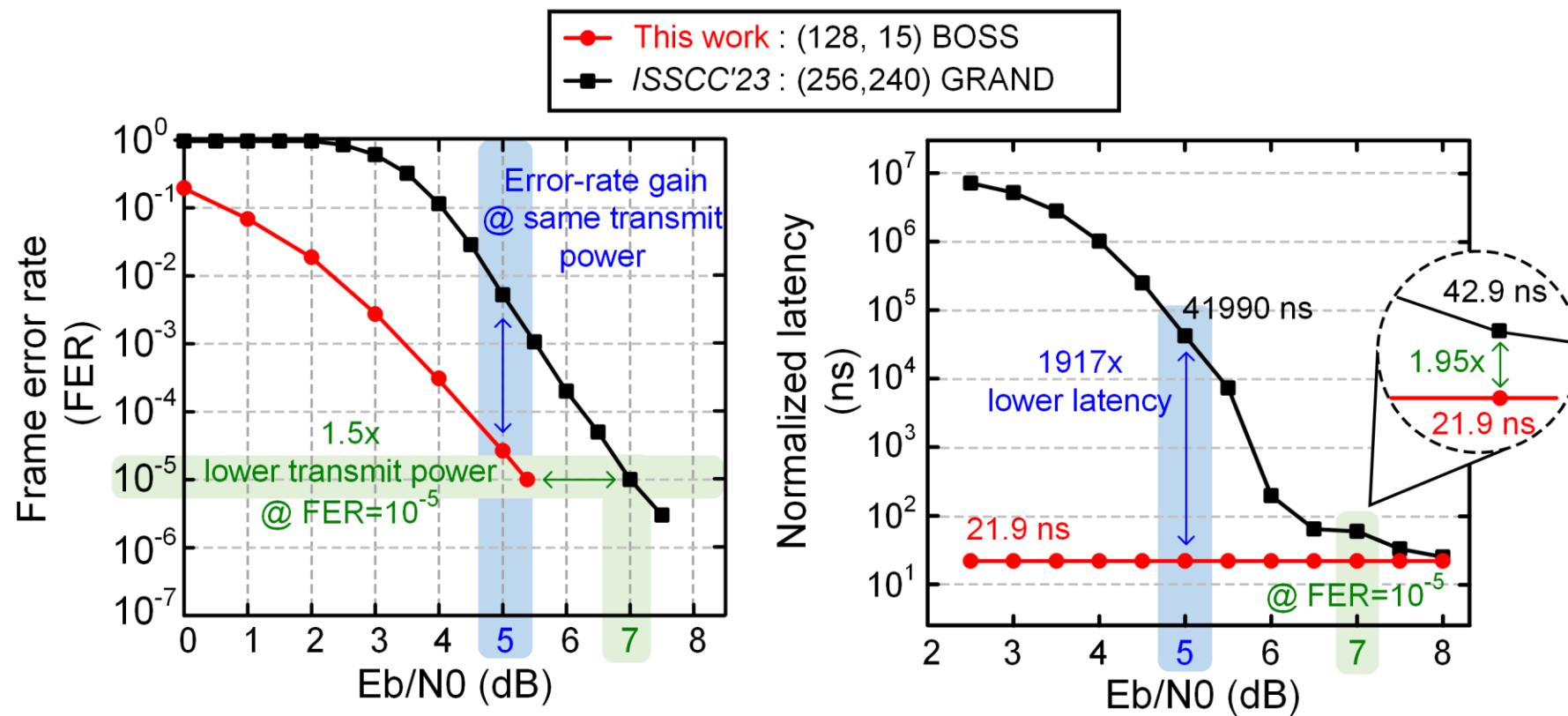
Norm. Area-effi. = Area-effi. / X³

Norm. Energy-effi.= Energy-effi. * X²Y²

† is measured for decoding a frame

where X = 28nm / used tech. , Y = 0.95V / used voltage

Case study : Comparison to GRAND decoder



- We evaluate the GRAND decoder at different $Eb/N0$.
 - The GRAND decoder has variable decoding latencies depending on $Eb/N0$.

Conclusion

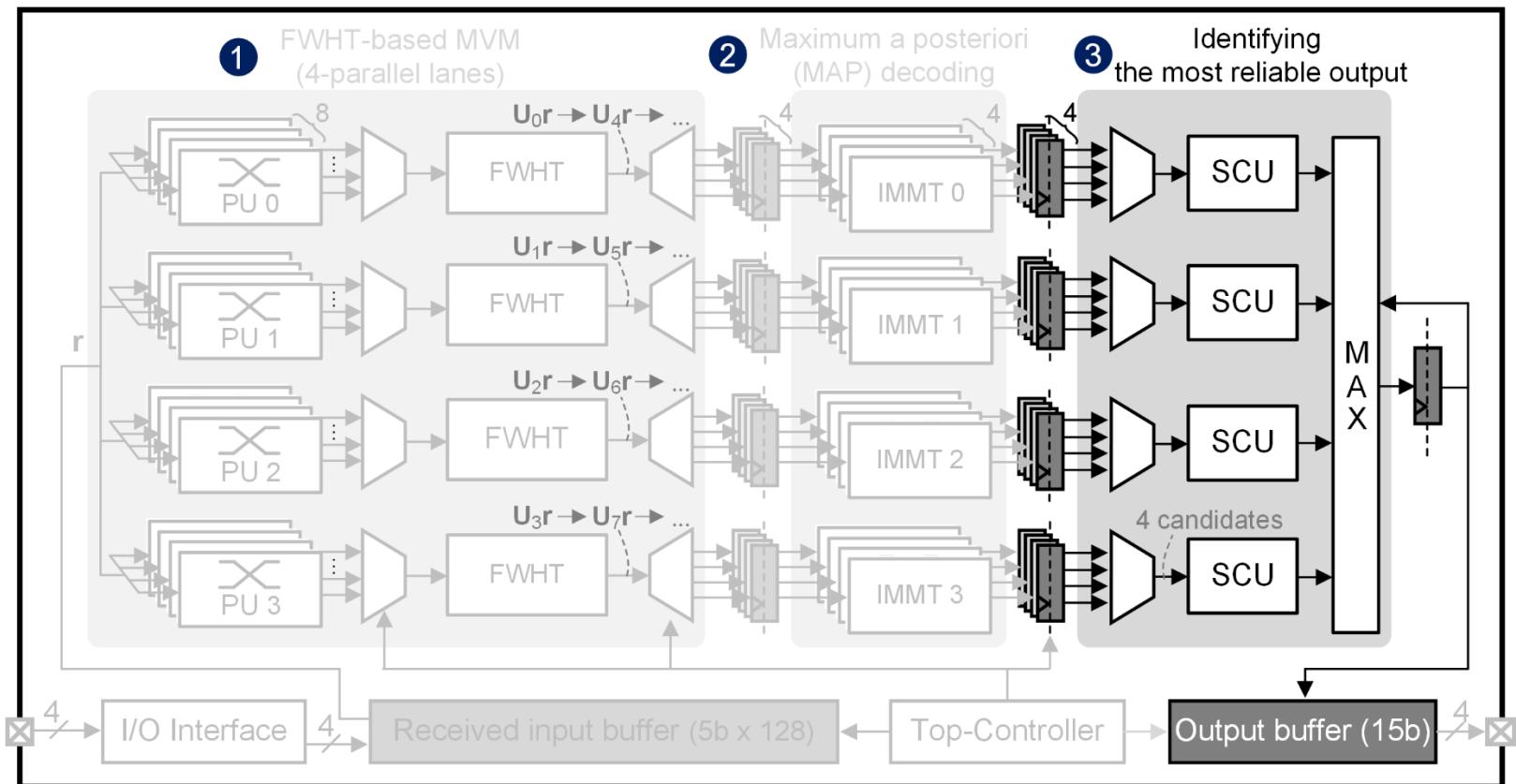
- **BOSS decoder for a 5G/6G URLLC FEC solution**
 - targets short-length and low-rate data transmissions for an **ultra-reliability**.
 - supports parallel & feedforward computations to achieve a **low-latency**.
 - includes **cost-efficient** circuits: FWHT-based MVM calculators and IMMTs.

- **Implementation of the BOSS decoder in 28nm CMOS**
 - achieves a low-latency of 21.9ns (6.2x lower than 5G polar decoder) and an area-efficiency of 15.78Gbps/mm² (2x better than 5G polar decoder).
 - successfully meets all the requirements for 5G/6G URLLC scenarios.

Thank you

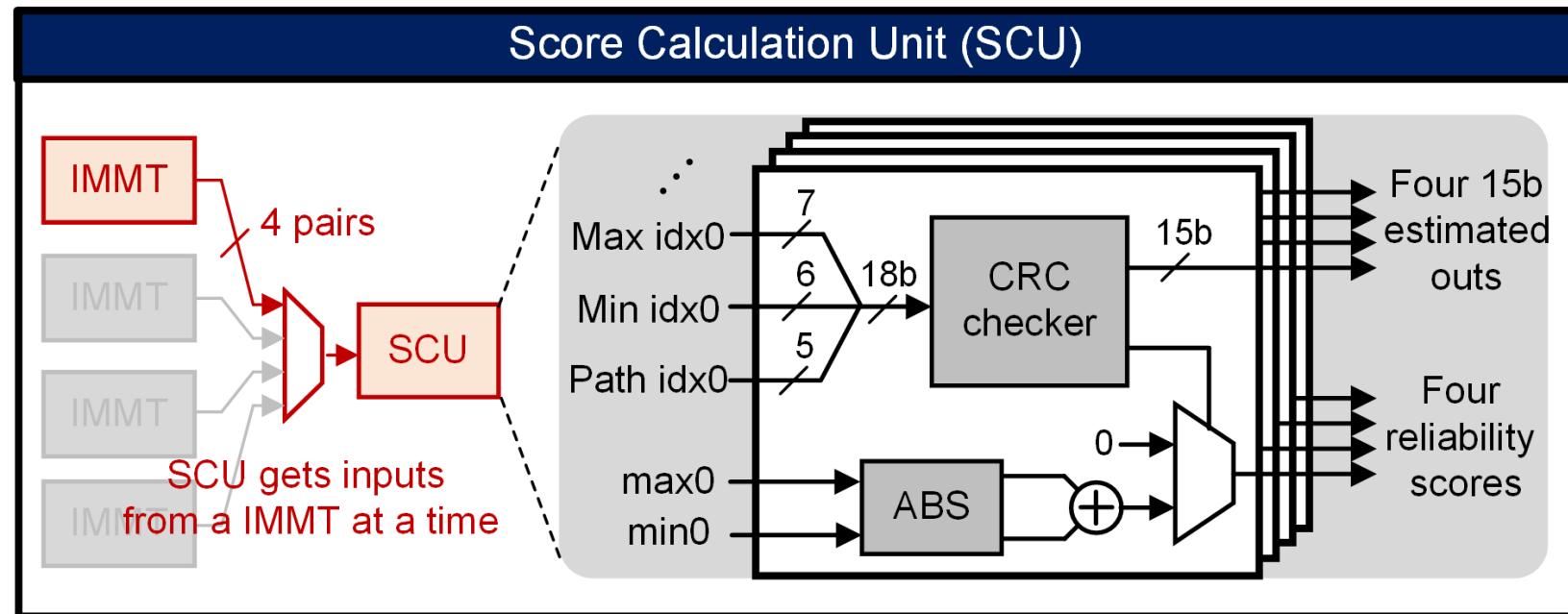
#2.8 A 21.9ns 15.7Gbps/mm² (128,15)
BOSS FEC Decoder for 5G/6G URLLC Applications

Appendix : Score calculation unit (SCU) (1/2)



■ Step 3 : Identifying the most reliable output

Appendix : Score calculation unit (SCU) (2/2)



■ Score Calculation Unit (SCU)

- Identifying the most reliable output among 128 candidates