Logic Synthesis and Automation for Memristive Memory Processing Unit

Shahar Kvatinsky

Viterbi Faculty of Electrical Engineering

Technion – Israel Institute of Technology

EPFL Workshop on Logic Synthesis and Emerging Technologies

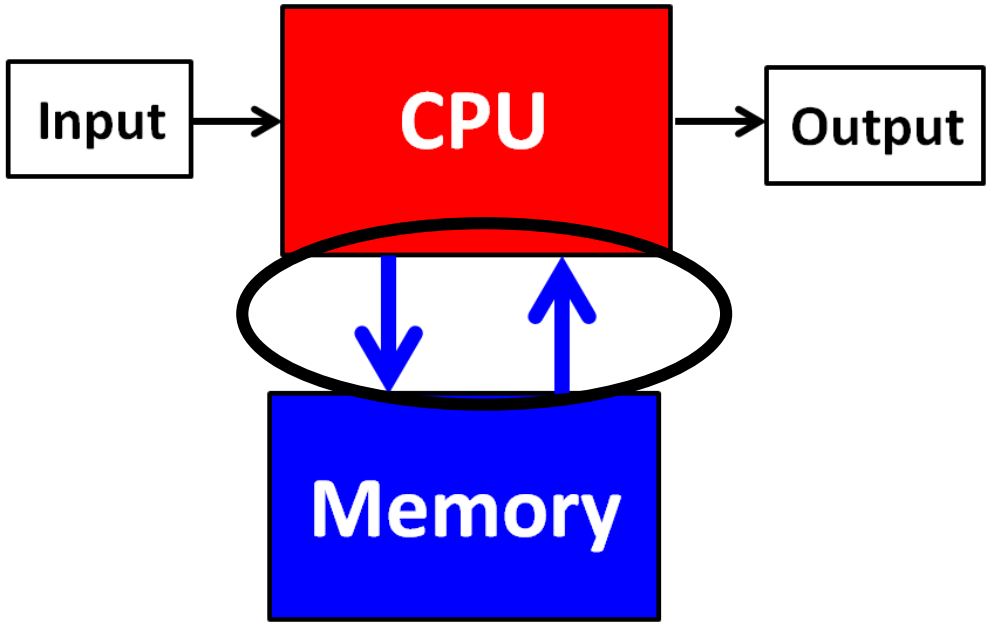
September 2017

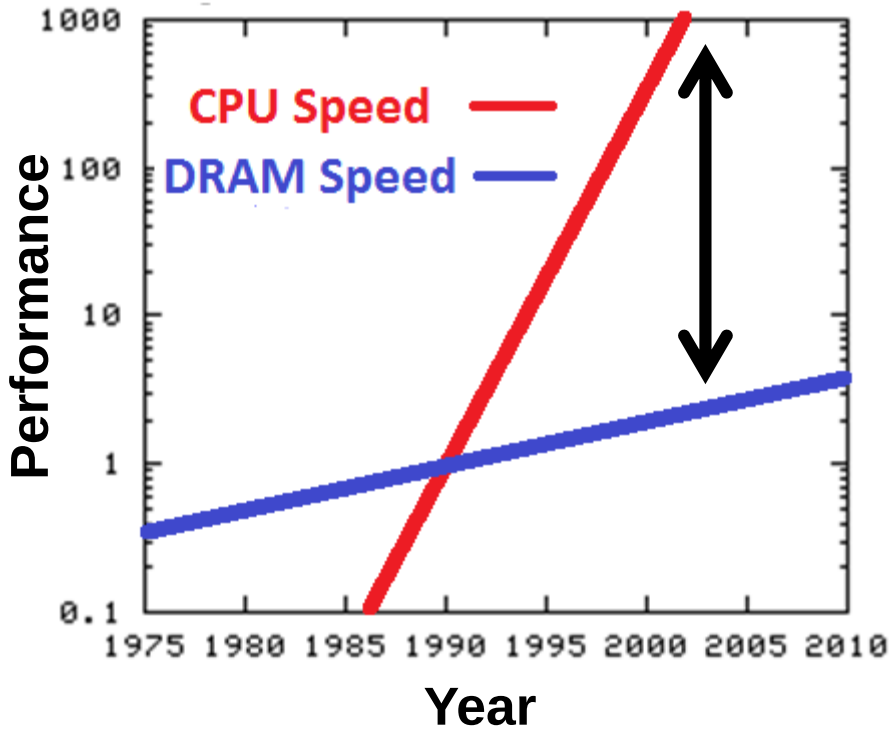
ASIC^2

ARCHITECTURES
SYSTEMS
INTELLIGENT COMPUTING
INTEGRATED CIRCUITS

The External Memory Wall Problem

von Neumann (Architecture) Bottleneck





A bottleneck for both throughput and power!

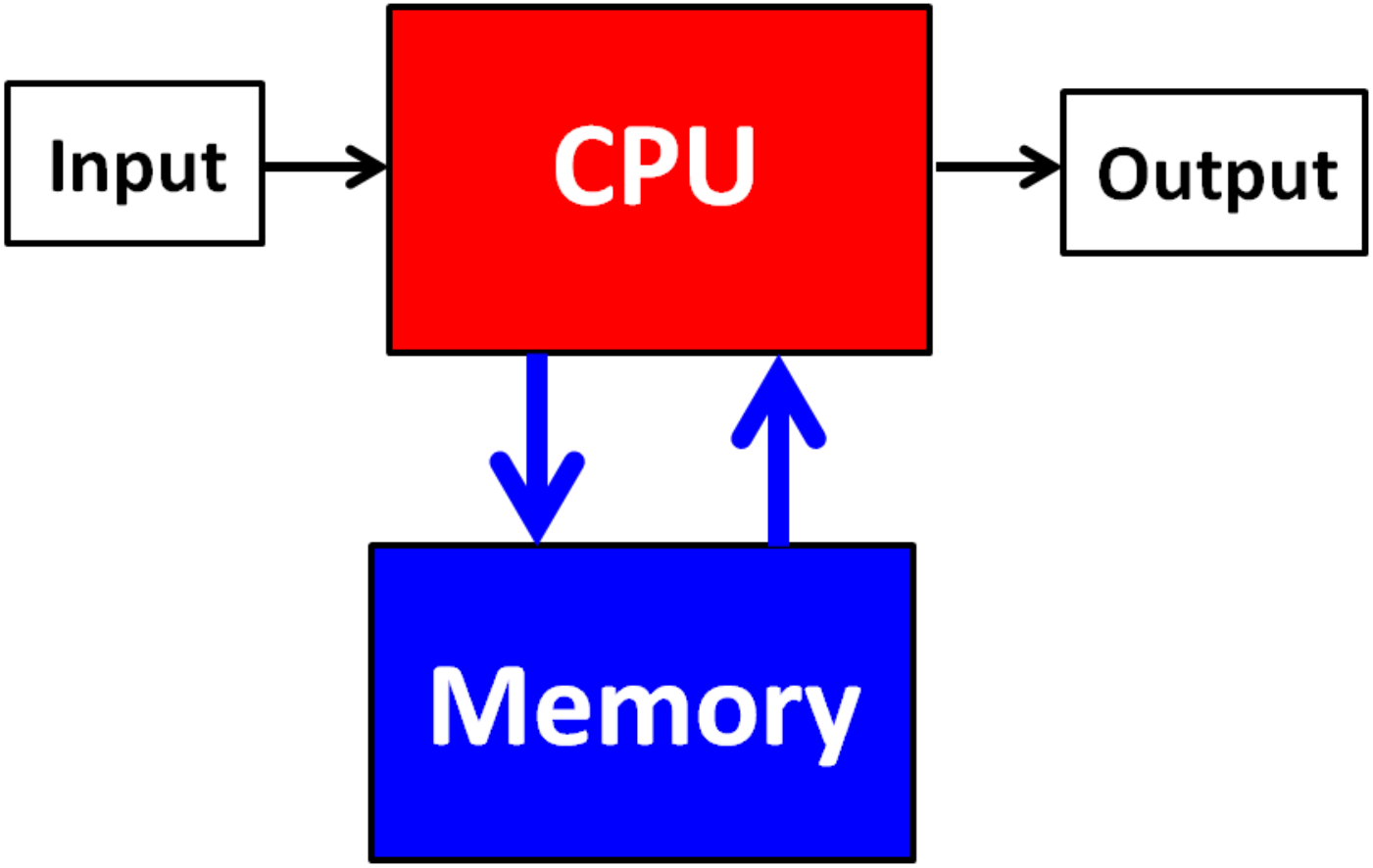
And an Energy Bottleneck

Operation (16-bit operand)	**Energy/Op** (45 nm)	**Cost** (vs. Add)
Add operation	0.18 pJ	**1X**
Load from on-chip SRAM	11 pJ	61X
Send to off-chip DRAM	**640 pJ**	**3,556X**

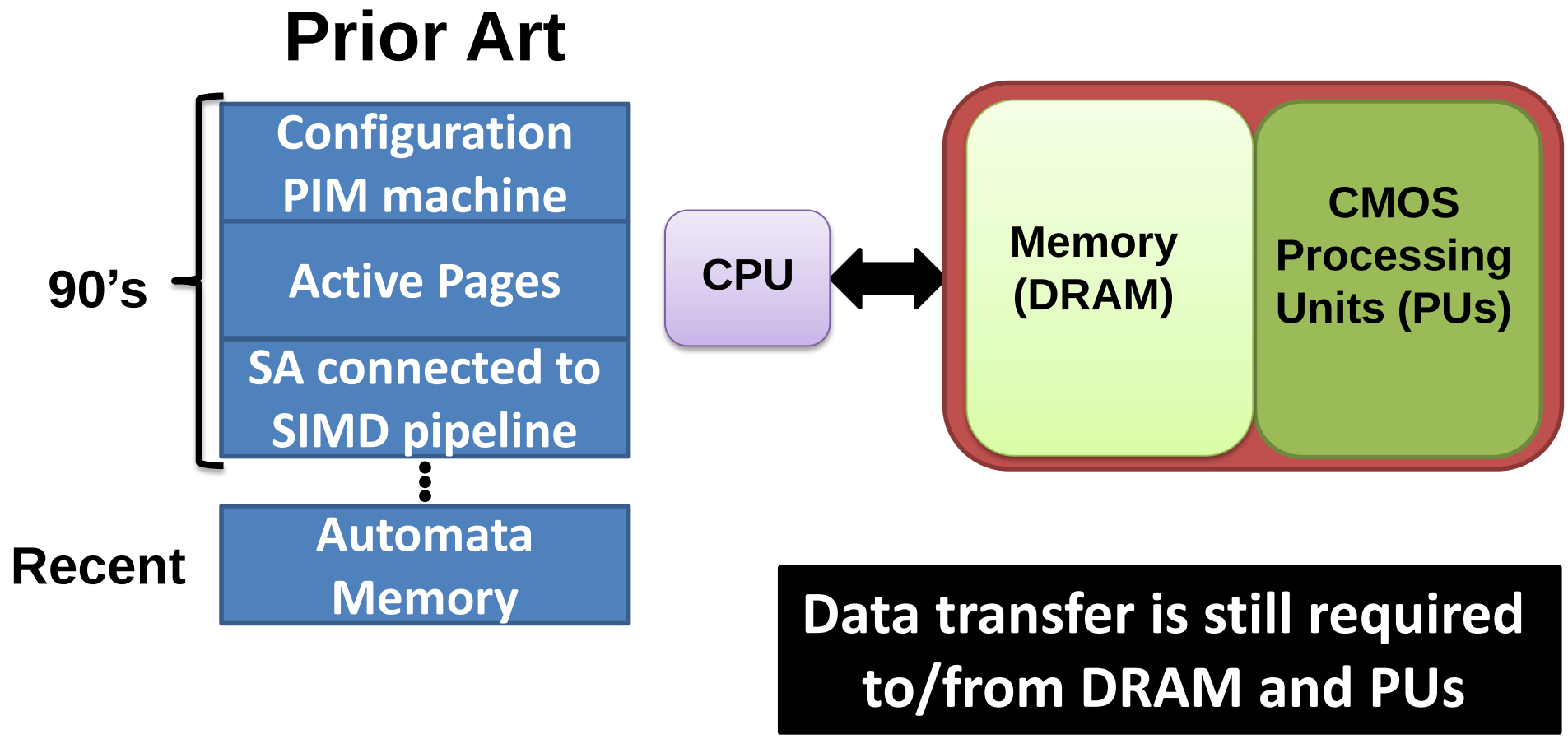
A. Pedram, S. Galal, S. Richardson, S. Kvatinsky, and M. Horowitz, "Dark Memory and Accelerator-Rich System Optimization in the Dark Silicon Era," IEEE Design & Test, April 2017

Processing "In-Memory" (PIM)

Reducing Data Movement



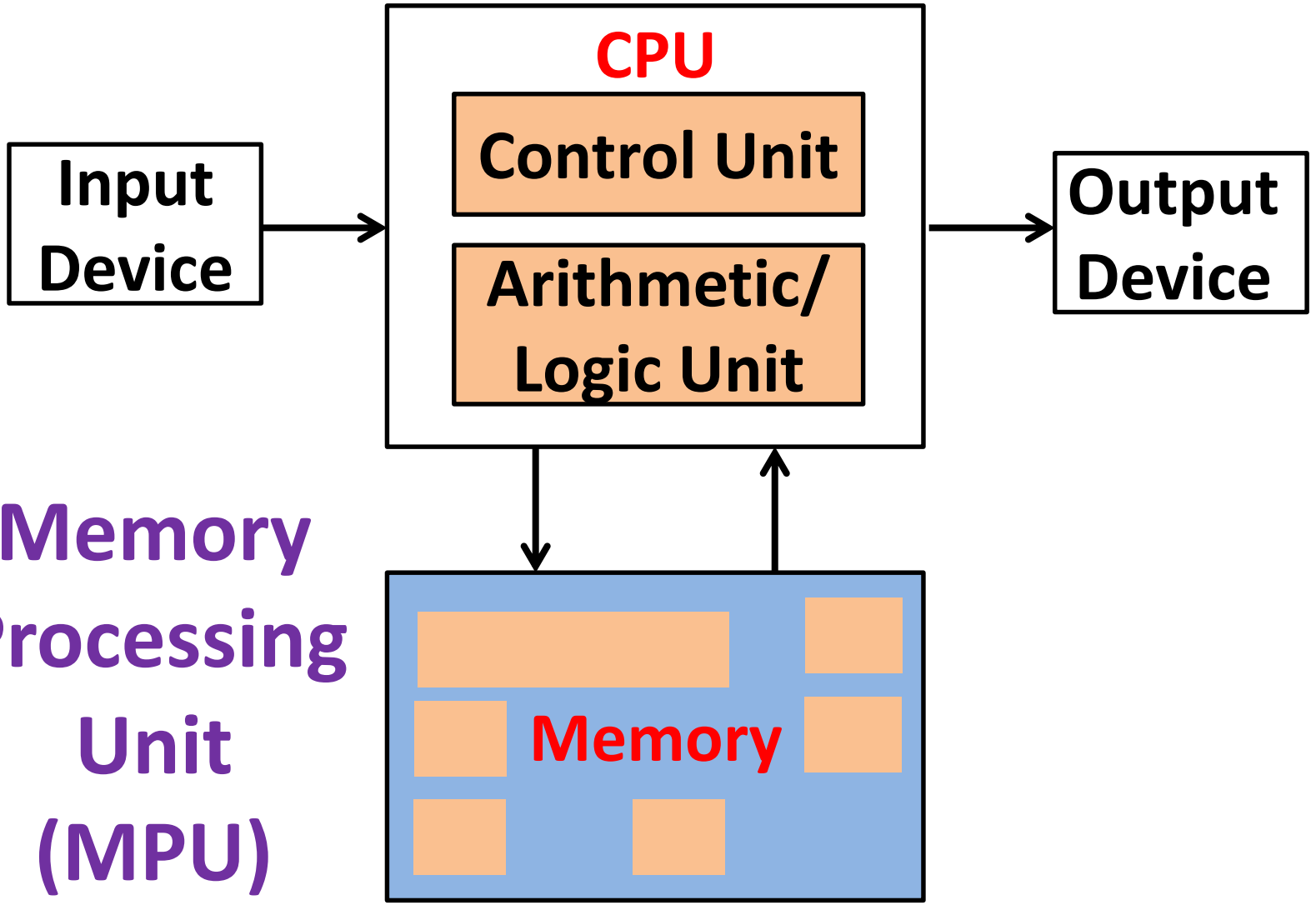
Processing "In-Memory" (PIM)
Reducing Data Movement



M. Gokhale *et al.*, "Processing in memory: the Terasys massively parallel PIM array," *Computer*, 1995
M. Oskin *et al.*, "Active pages: A computation model for intelligent memory," *Comput. Archit. News*, 1998
D. Elliott *et al.*, "Computational ram: Implementing processors in memory," *IEEE Des. Test*, 1999
P. Dlugosch *et al.*, "An Efficient and Scalable Semiconductor Architecture for Parallel Automata Processing," *IEEE TPDS*, 2014

Real Computing within the Memory

Beyond von Neumann Architecture

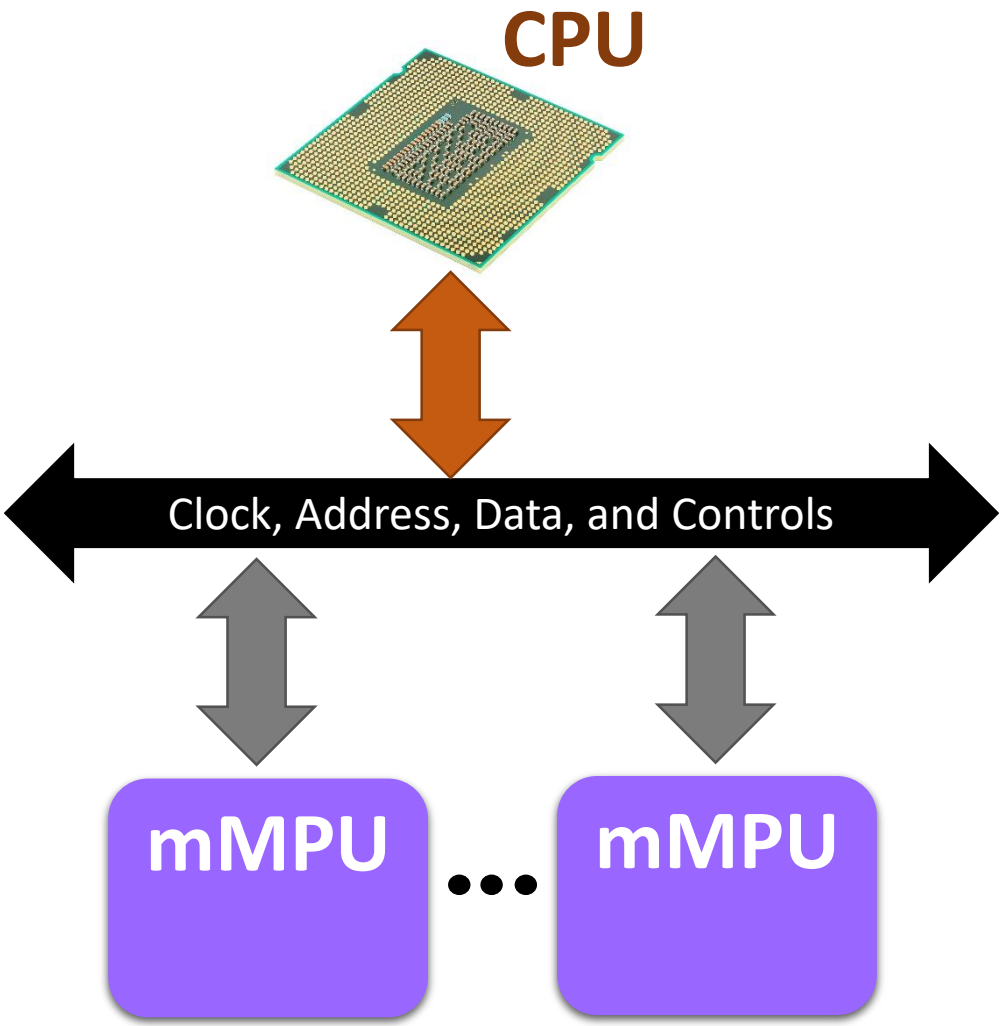


Memory Processing Unit (MPU)

mMPU: Solving the von Neumann Bottleneck

Moving from DRAM to memristive memory

mMPU: performing computation *USING* the memristive memory cells



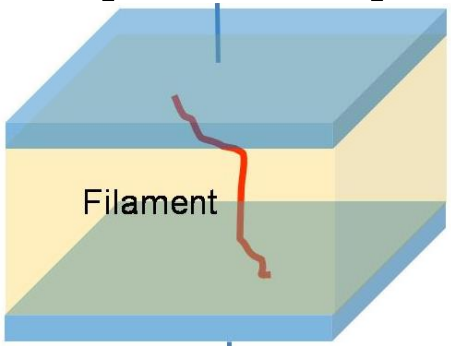
Agenda

- The need for non-von Neumann architectures
- **Memristive technologies**
- Memristive MPU (mMPU) architecture
- mMPU logic synthesis and automation
- Summary

Memristors

Emerging Nonvolatile Memory Technologies

Resistive RAM (RRAM)



SanDisk SONY

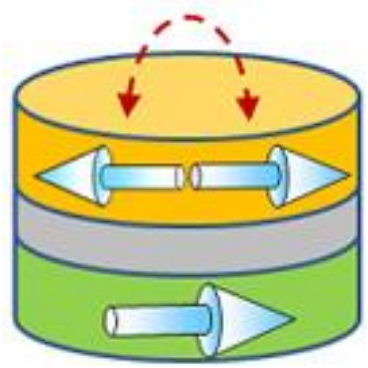
hp SAMSUNG

winbond

Panasonic

SK hynix TOSHIBA Crossbar

STT MRAM



EVERSPIN TECHNOLOGIES

HITACHI

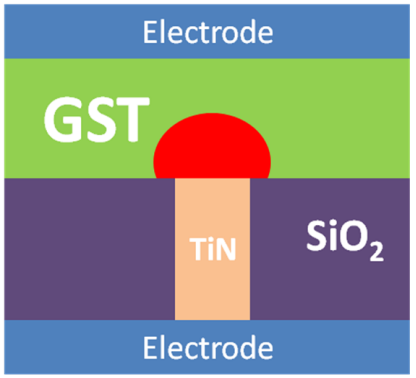
CROCUS Technology Blossoming future

TOSHIBA

QUALCOMM

SAMSUNG

Phase Change Memory (PCM)



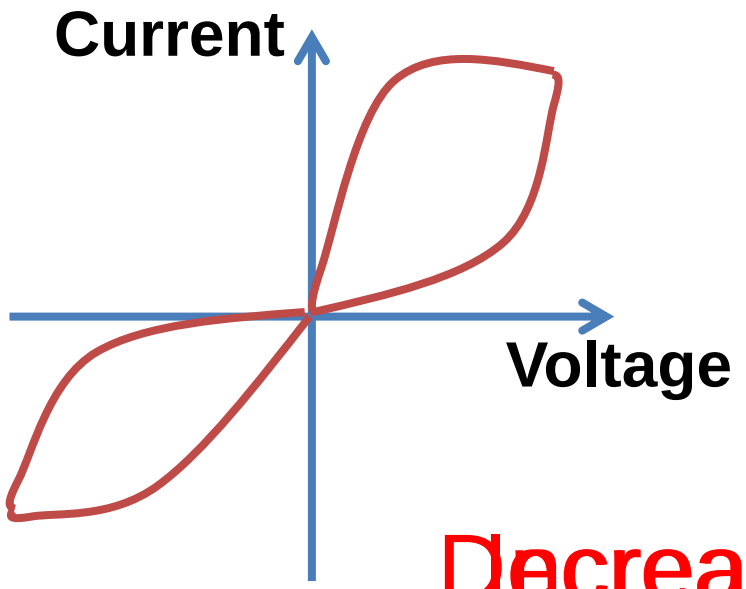
SAMSUNG intel

IBM

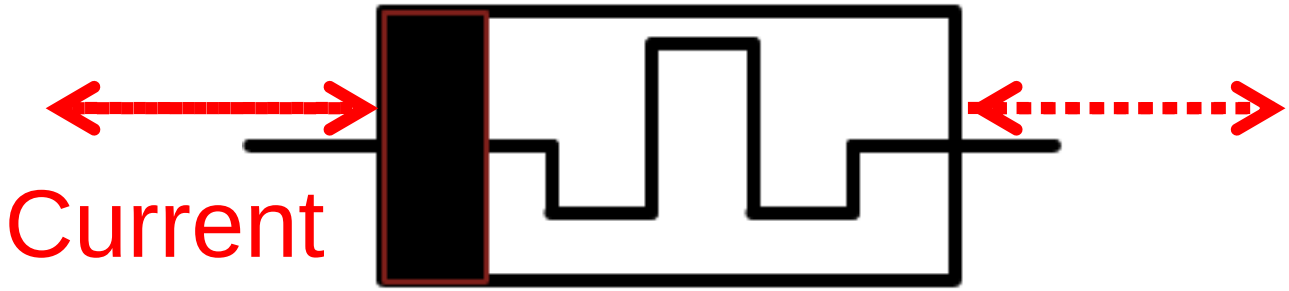
Micron SK hynix

Memristor – Memory Resistor

Resistor with Varying Resistance



Decrease resistance



Agenda

- The need for non-von Neumann architectures
- Memristive technologies
- **Memristive MPU (mMPU) architecture**
- mMPU logic synthesis and automation
- Summary

MAGIC – Memristor Aided LoGIC

Example of MAGIC NOR

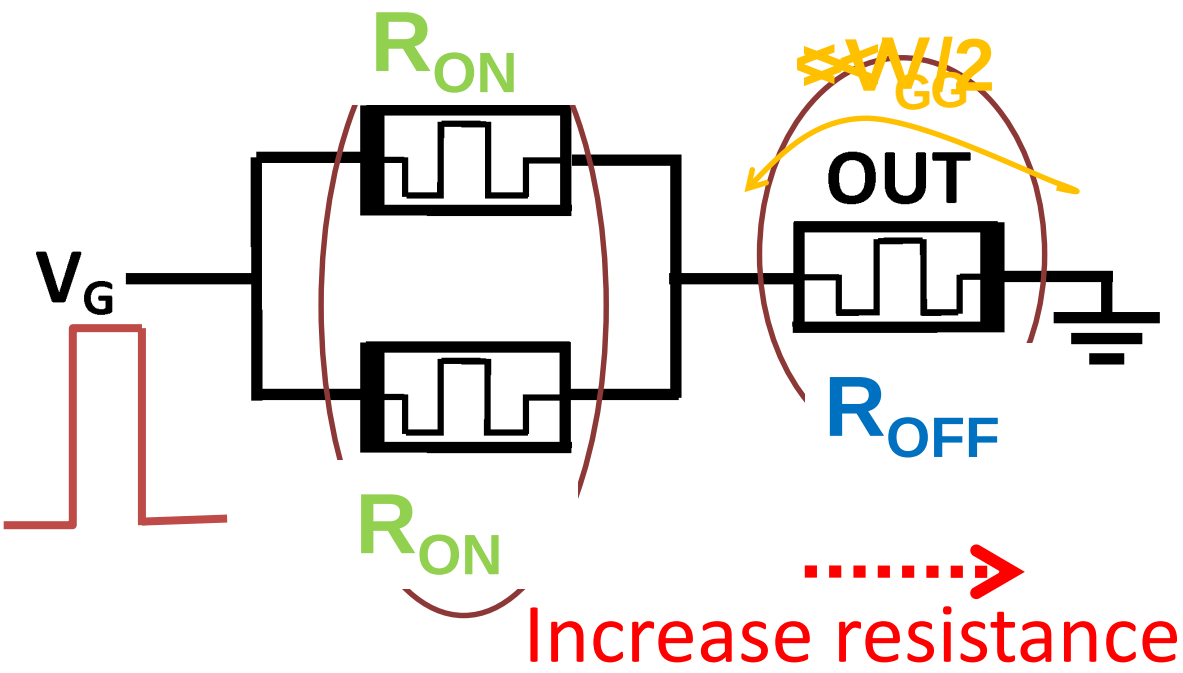
Initialize OUT to R_{ON}

$R_{OFF} >> R_{ON}$

R_{ON} =Logic '1'

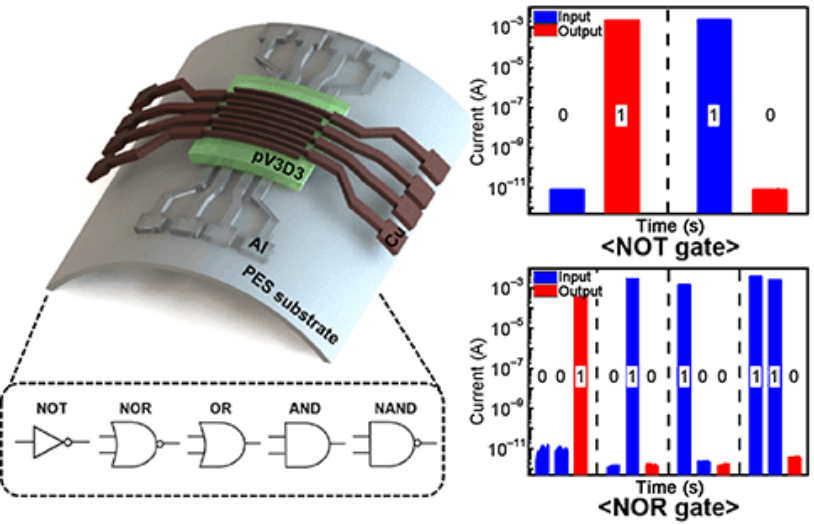
R_{OFF}=Logic '0'

IN_1	IN_2	*NOR*
0	0	1
0	1	0
1	0	0
1	1	0

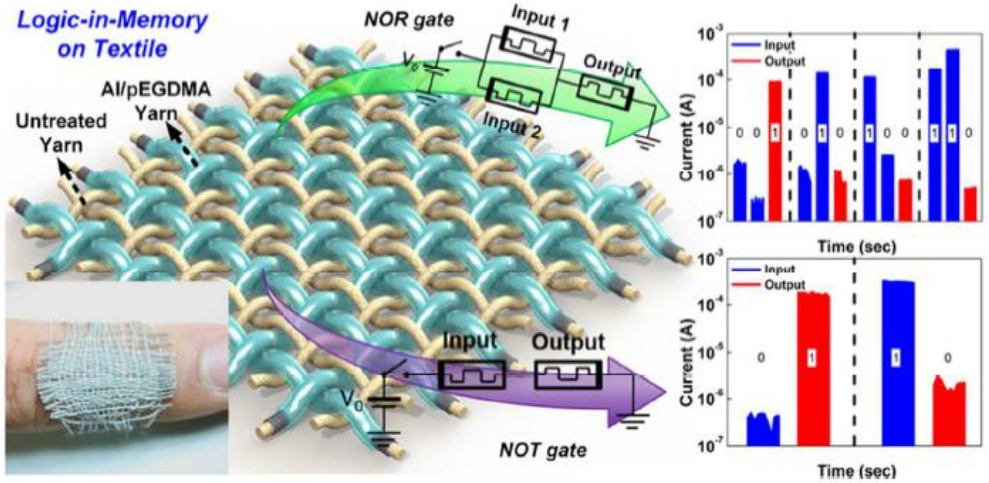


S. Kvatinsky, D. Belousov, S. Liman, G. Satat, N. Wald, E. G. Friedman, A. Kolodny, and U. C. Weiser, "MAGIC – Memristor Aided LoGIC," IEEE TCAS II, Nov. 2014

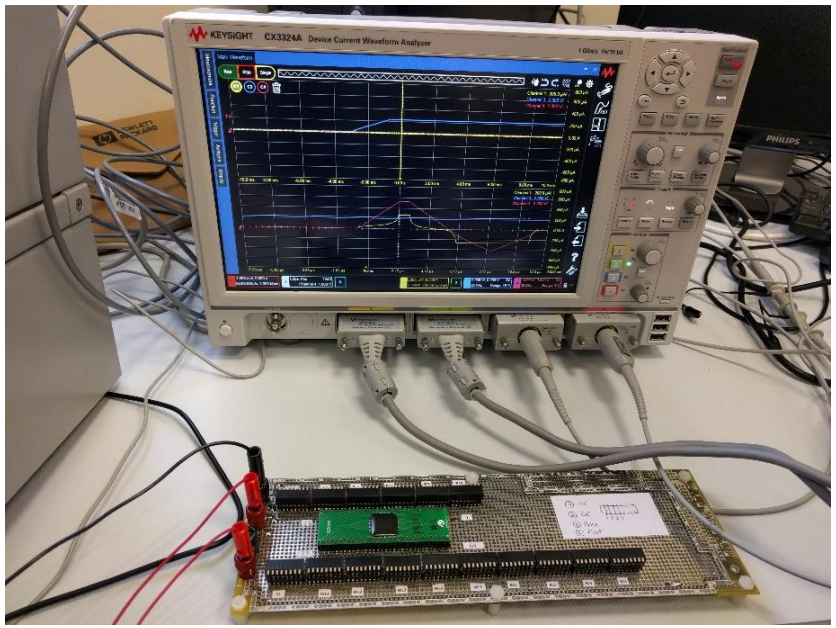
Real MAGIC



Jung *et al.*, *Nano Research*, July 2017



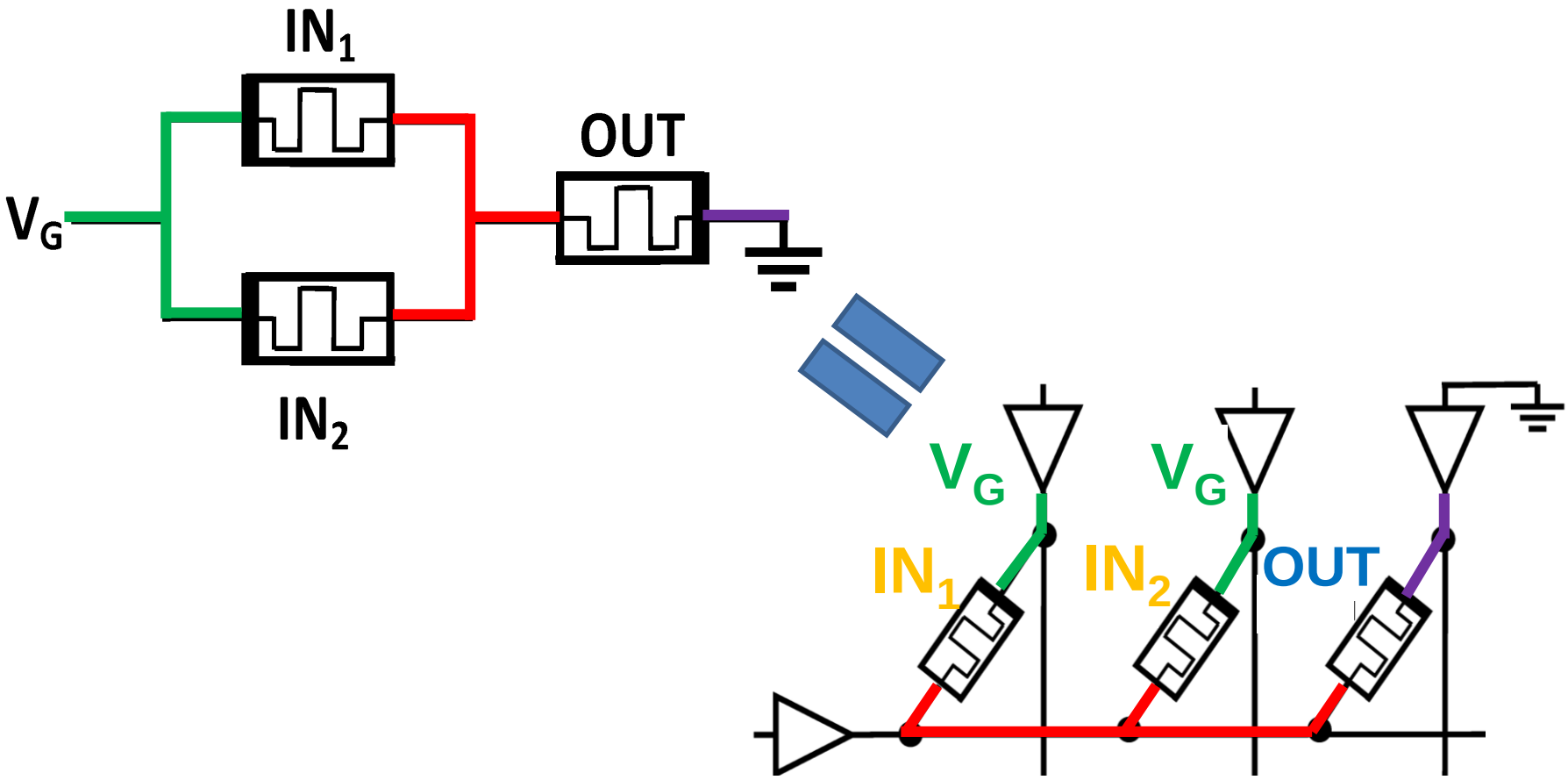
Bae *et al.*, *Nano Letters* (accepted)



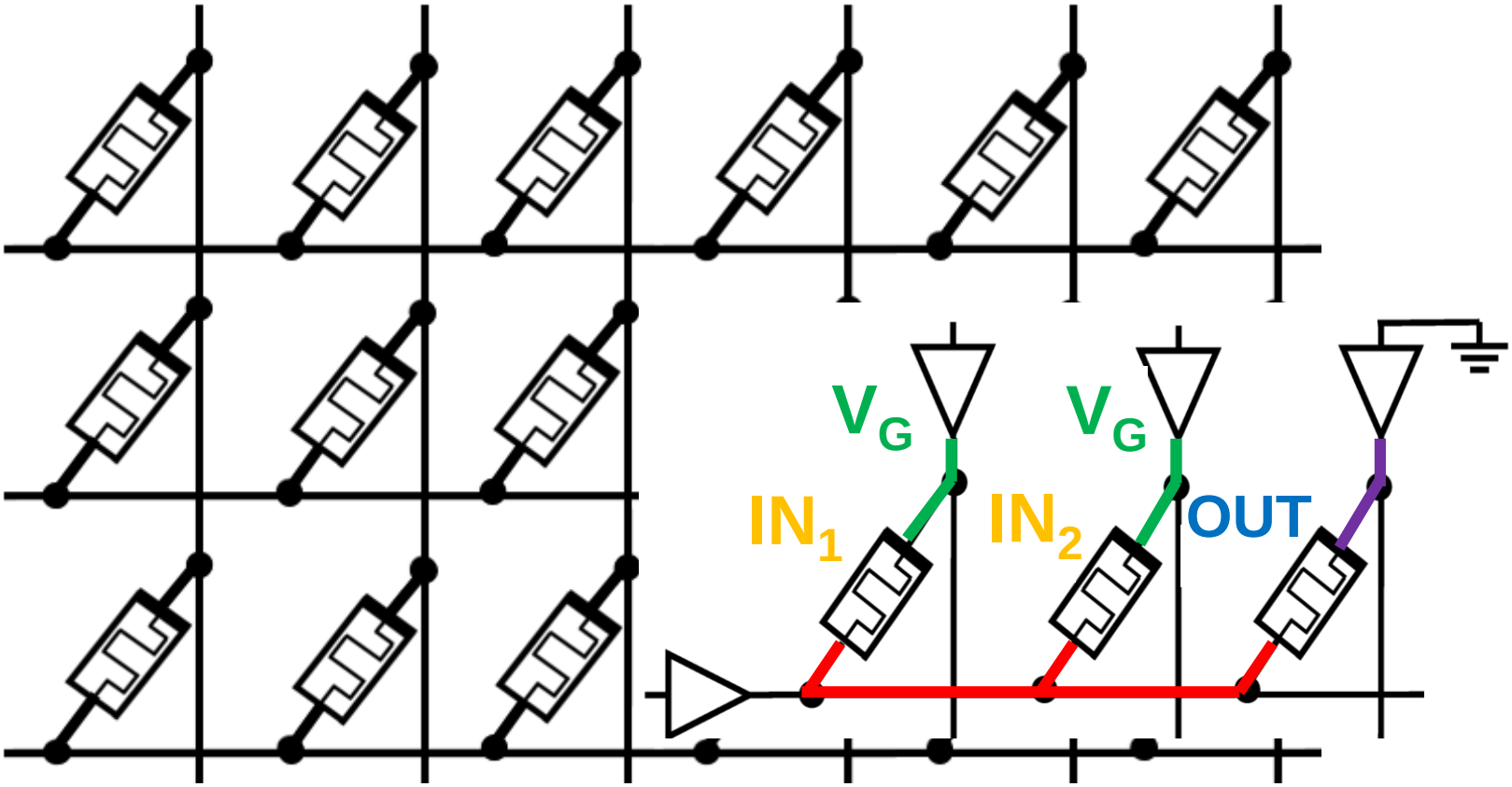
ASIC² winbond

Our lab (HfOx based)

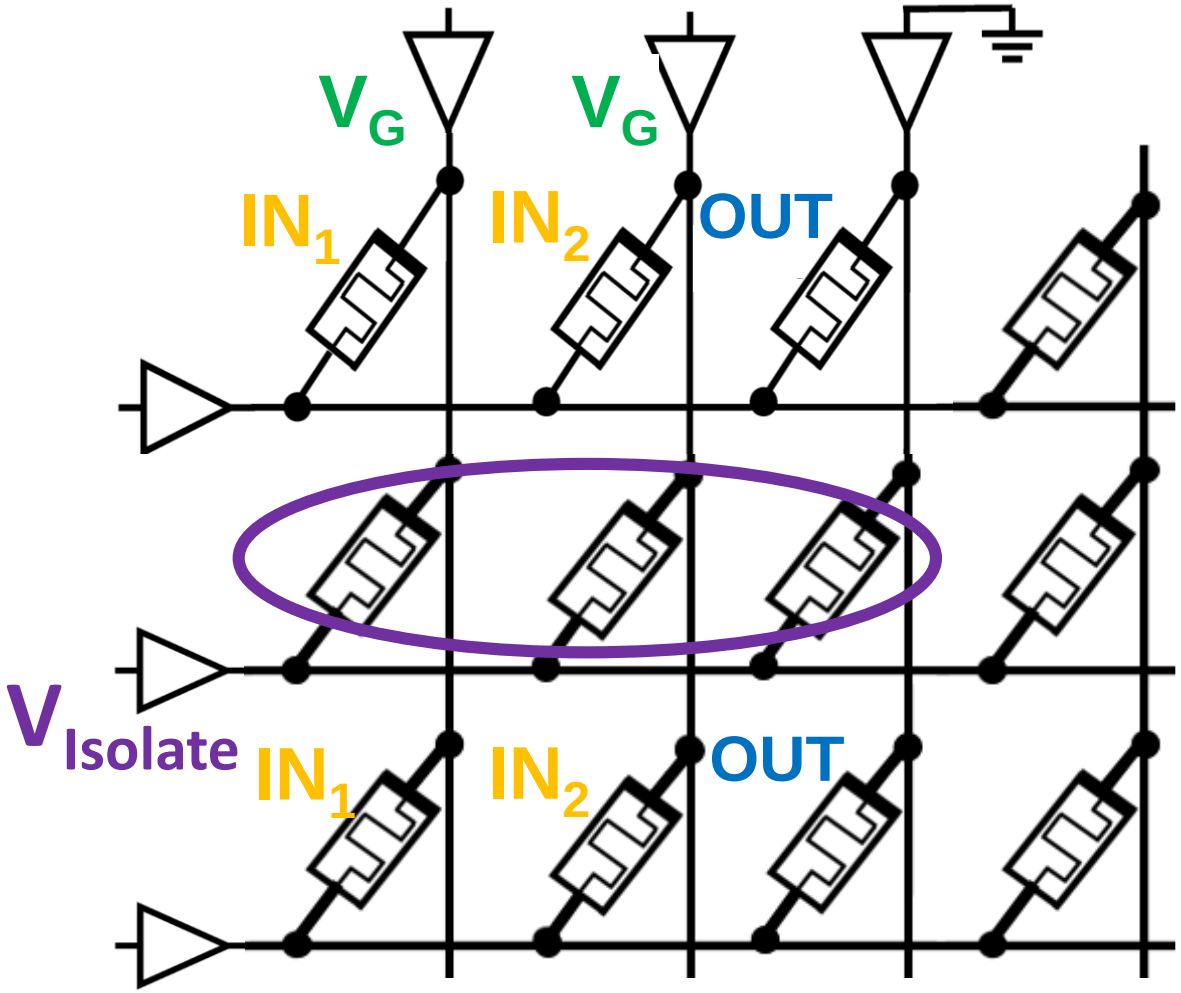
MAGIC NOR in a Crossbar



MAGIC NOR in a Crossbar



MAGIC NOR in a Memristive Memory



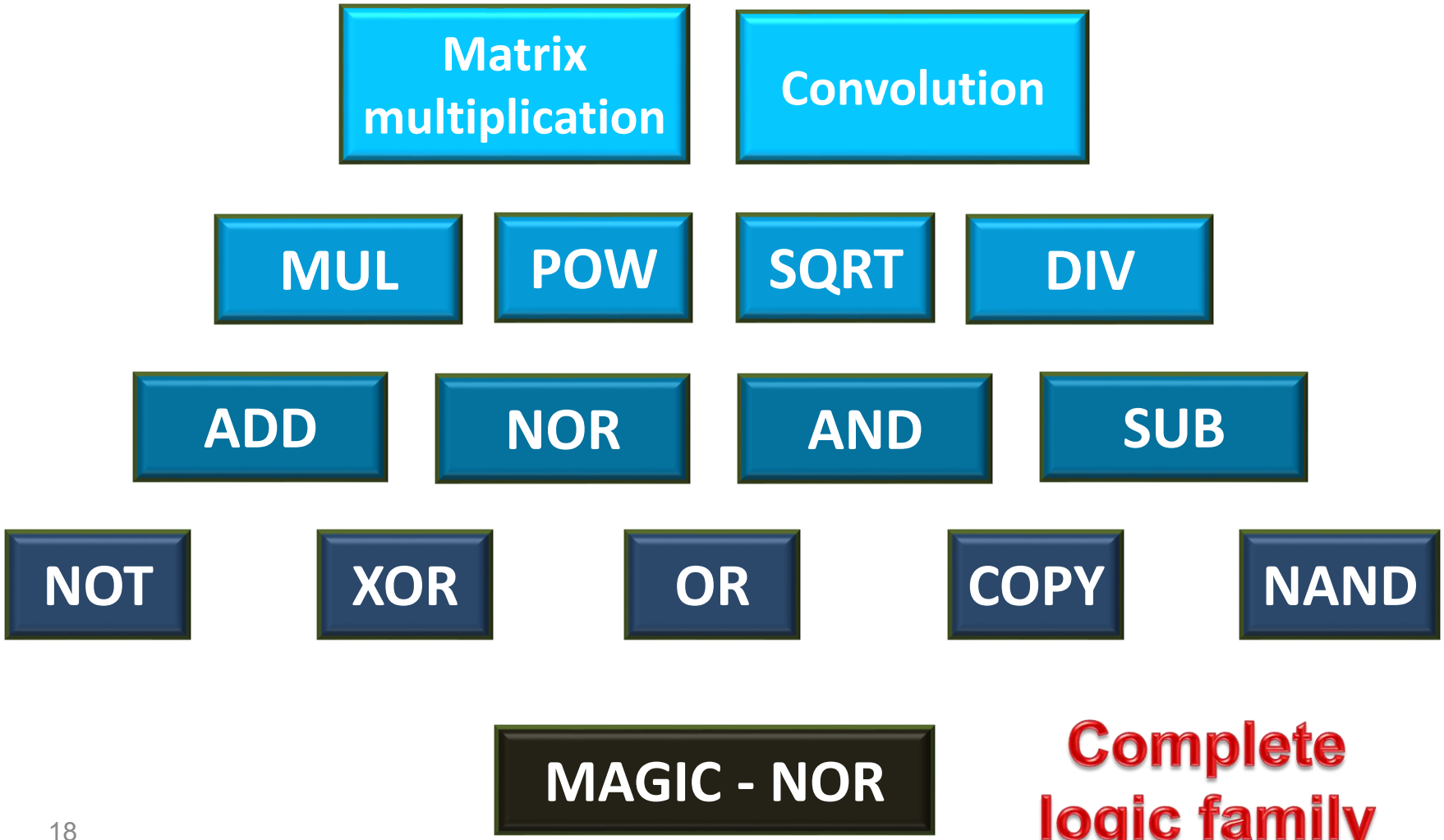
Parallelism

↓

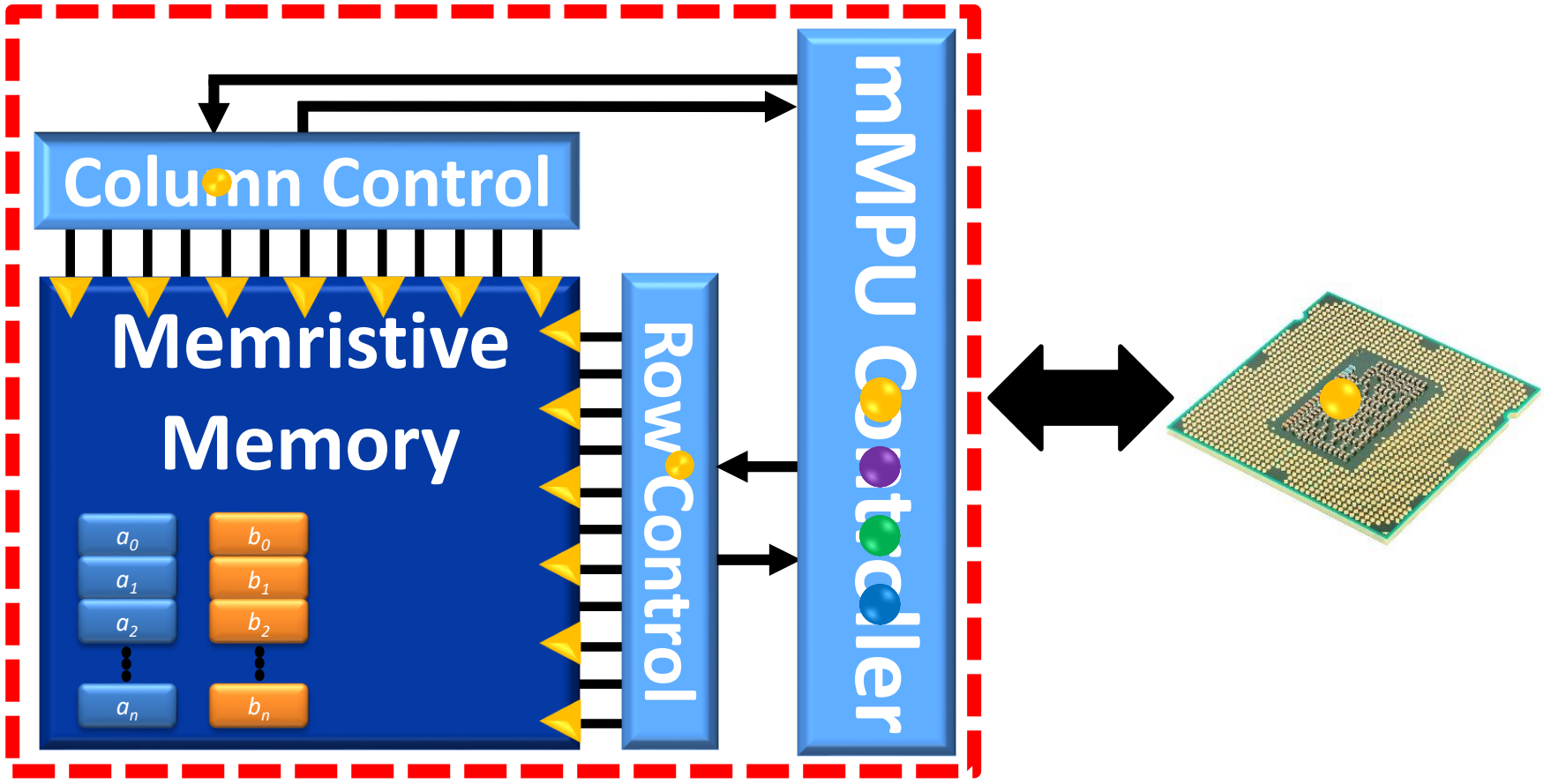
SIMD

N. Talati, S. Gupta, P. Mane, and S. Kvatinsky, "Logic Design within Memristive Memories Using MAGIC," IEEE Transactions on Nanotechnology, July 2016

Hierarchy of Logical Functions

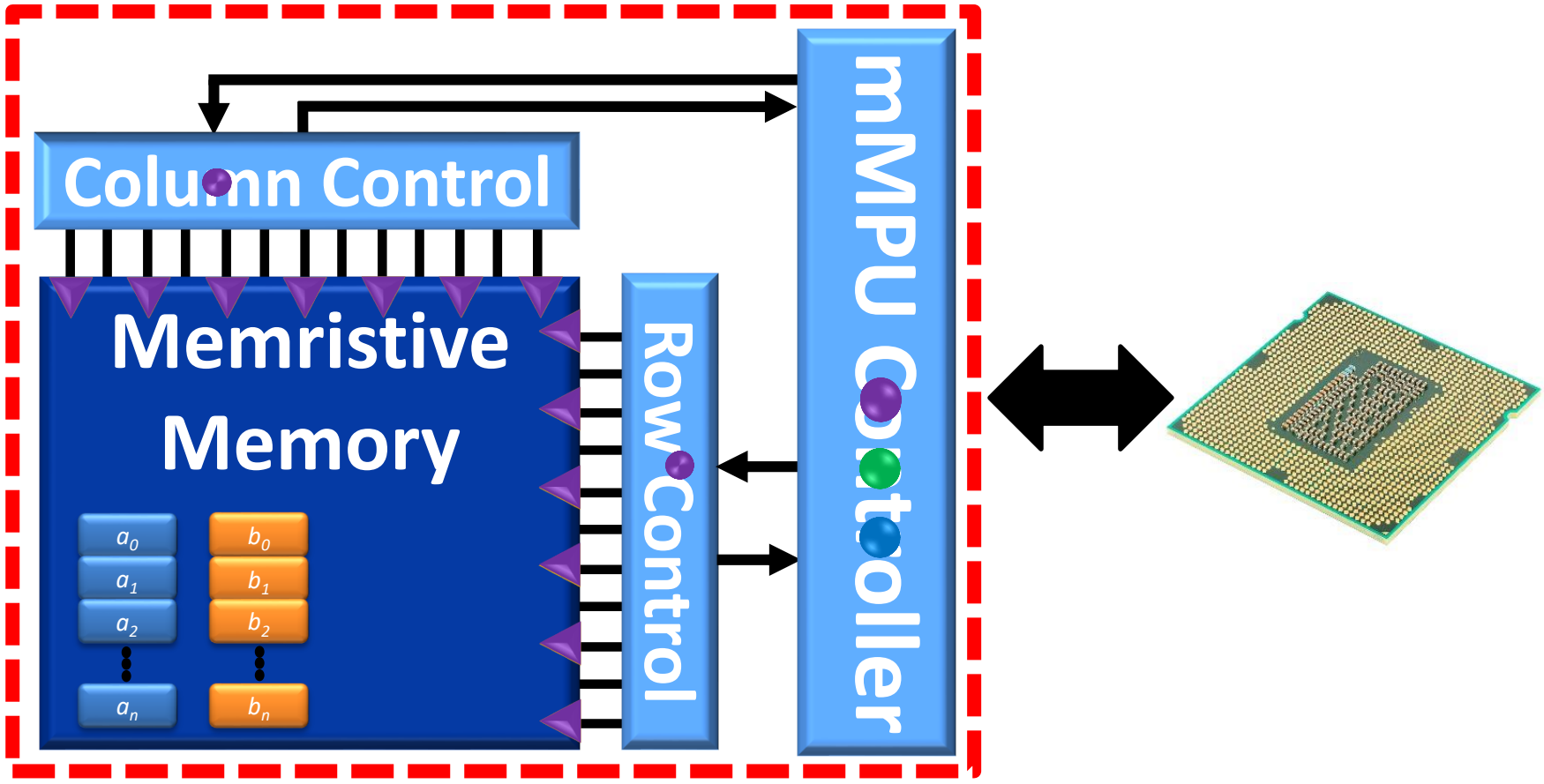


mMPU µArchitecture



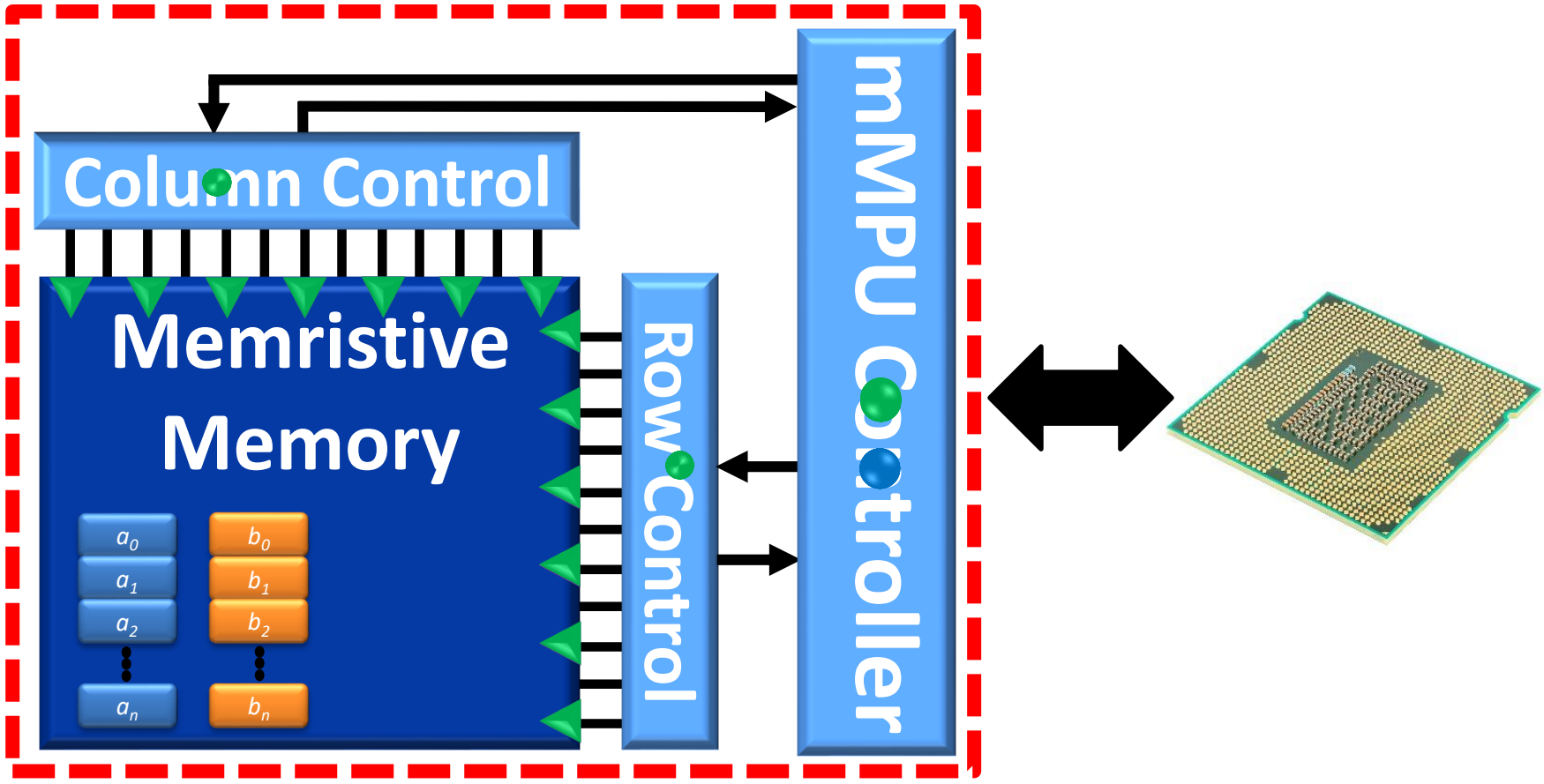
R. Ben-Hur and S. Kvatinsky, "Memory Processing Unit for In-Memory Processing," NANOARCH 2016

mMPU µArchitecture



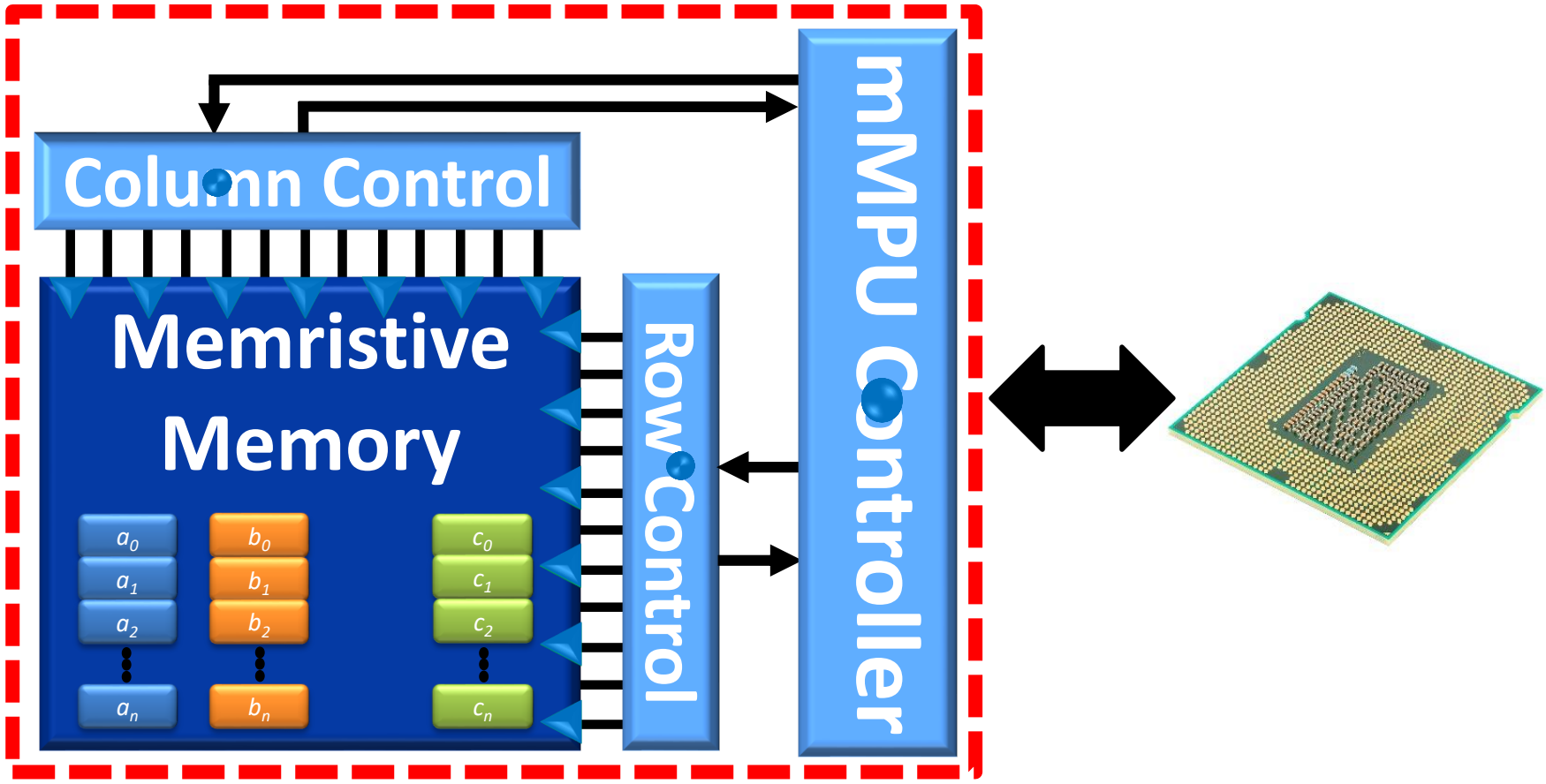
R. Ben-Hur and S. Kvatinsky, "Memory Processing Unit for In-Memory Processing," NANOARCH 2016

mMPU µArchitecture



R. Ben-Hur and S. Kvatinsky, "Memory Processing Unit for In-Memory Processing," NANOARCH 2016

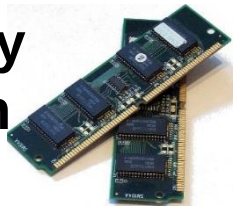
mMPU µArchitecture



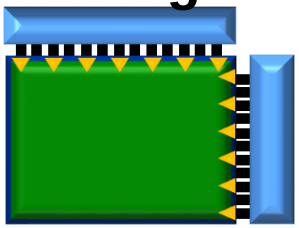
R. Ben-Hur and S. Kvatinsky, "Memory Processing Unit for In-Memory Processing," NANOARCH 2016

Issues Involved in mMPU Architecture

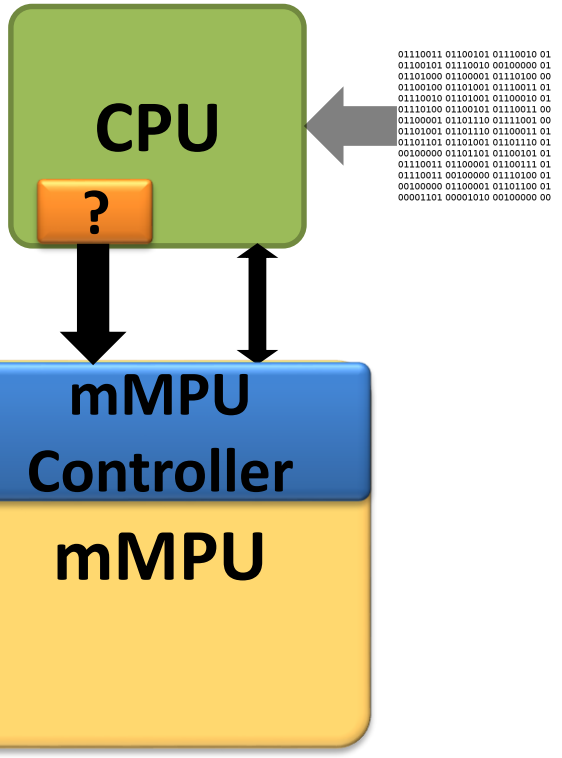
Memory Design



Periphery Design



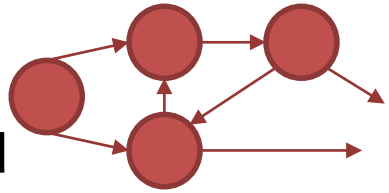
Real-PIM-System



Programming Model

Software

mMPU Controller Design and Optimization



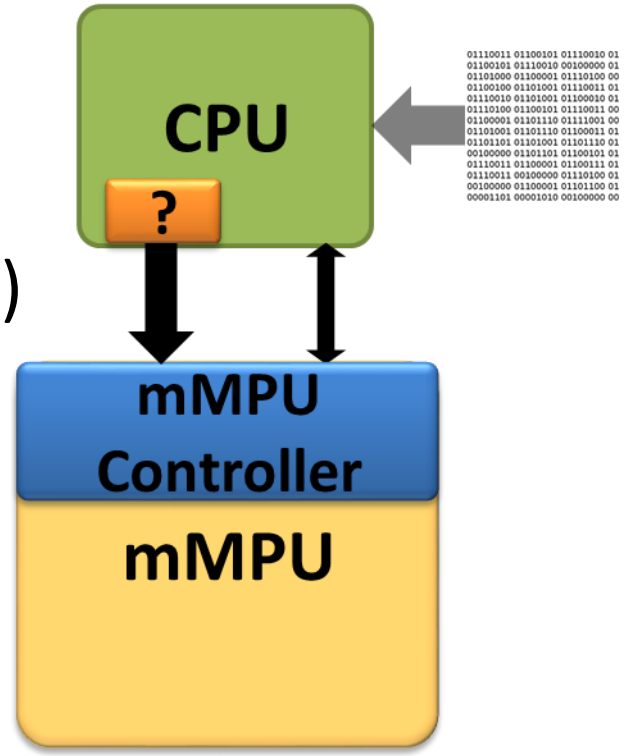
Applications

Agenda

- The need for non-von Neumann architectures
- Memristive technologies
- Memristive MPU (mMPU) architecture
- **mMPU logic synthesis and automation**
- Summary

mMPU Controller Design and Automation

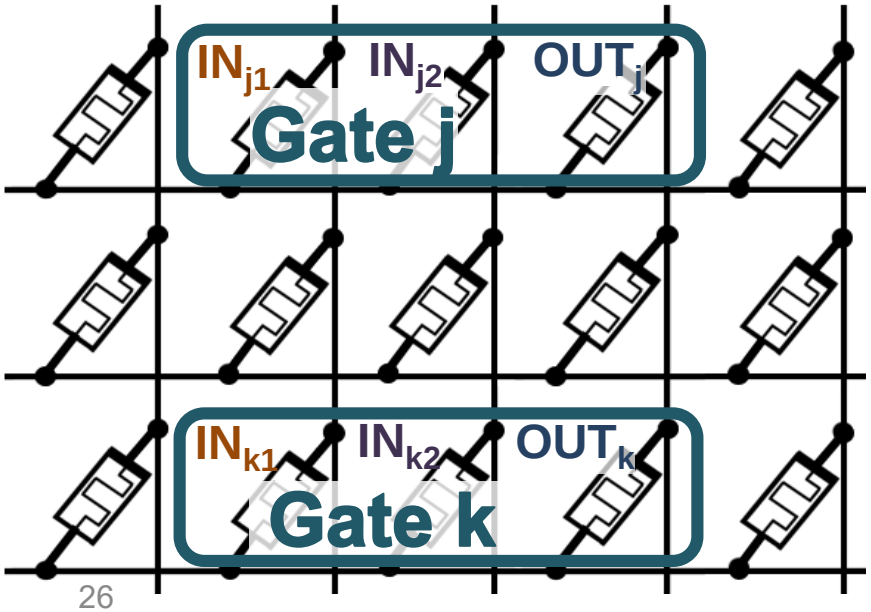
- Supports regular memory operations
- Optimized logic flow:
  - Parallelism (in-array and banks)
  - Cost function (latency, area, energy)
- Real-time memory mapping



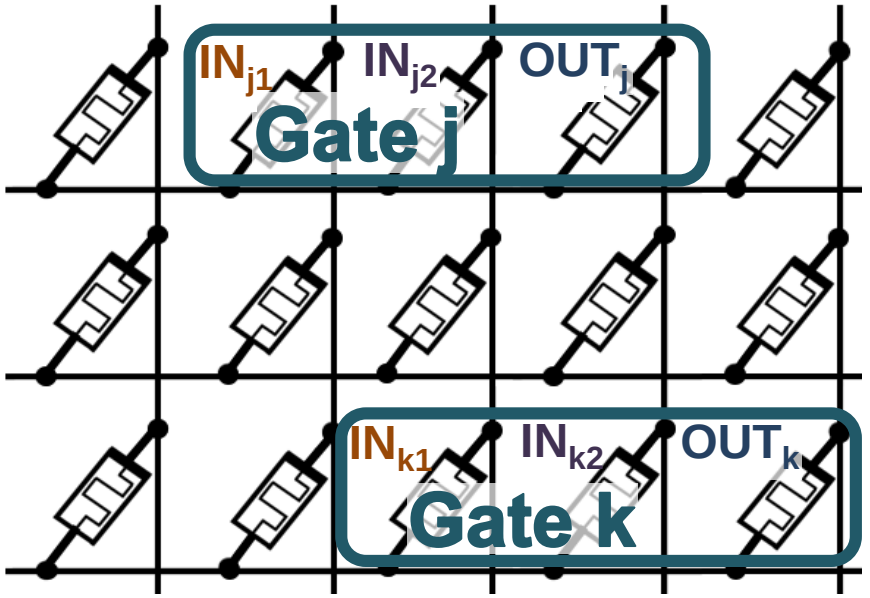
Exploit Parallelism

- Reducing the number of gates is not enough
- Mapping determines the possible parallelism

2 gates in 1 step

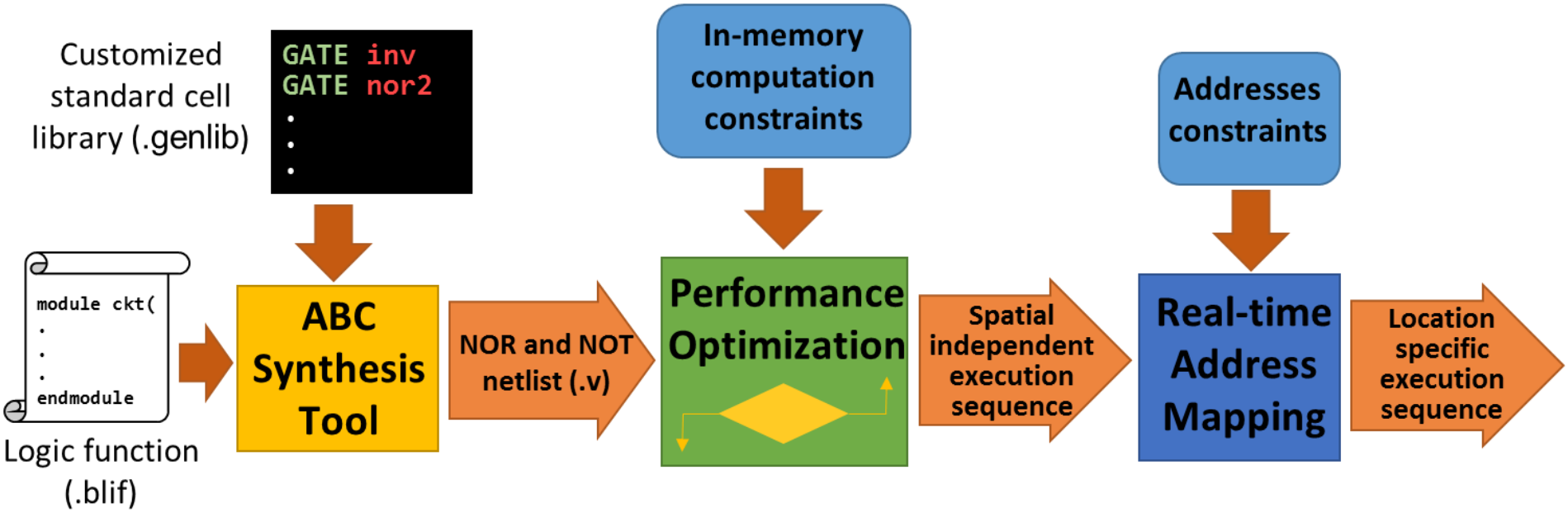


2 gates in 2 steps



SIMPLE MAGIC

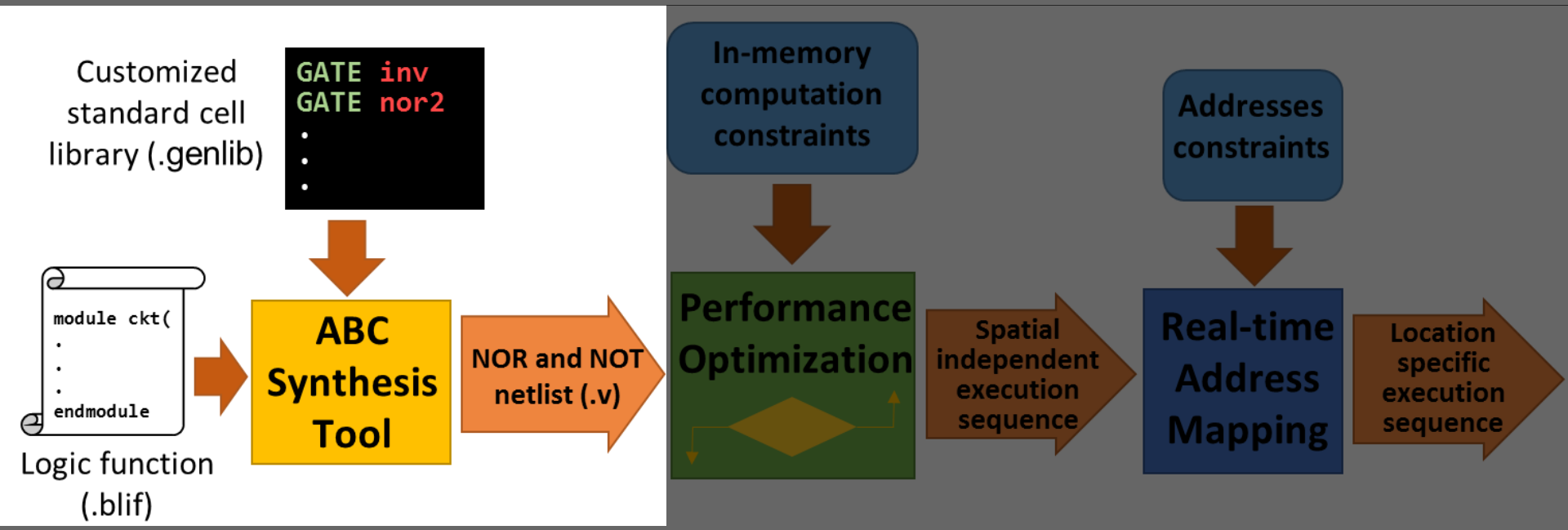
Synthesis and In-memory MaPping of Logic Execution for Memristor-Aided loGIC



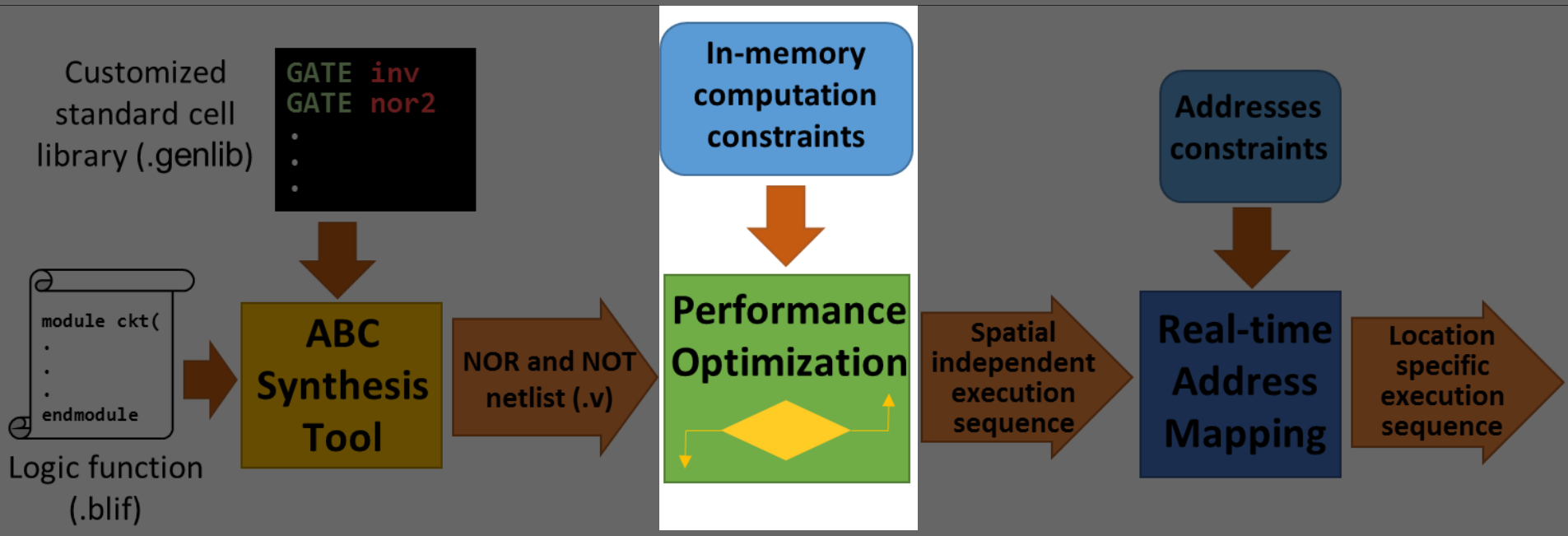
- **Both reducing the number of gates and mapping into the memristive memory**

R. Ben-Hur, N. Wald, N. Talati, and S. Kvatinsky, "Synthesis and Mapping of Boolean Functions for MAGIC," ICCAD 2017

SIMPLE MAGIC

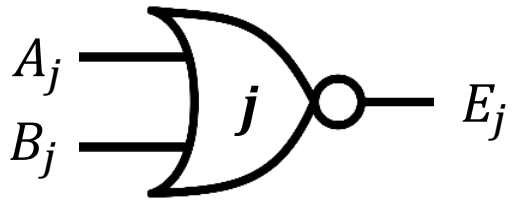


SIMPLE MAGIC



Variables

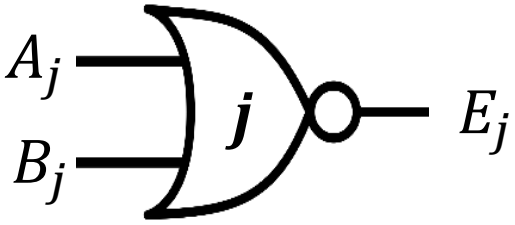
- **For every gate *j*:**
  - Locations (rows/columns):
    - $C_{A_j}, C_{B_j}, C_{E_j}, R_{A_j}, R_{B_j}, R_{E_j}$
  - Clock cycle of execution: T_j



Optimization Function

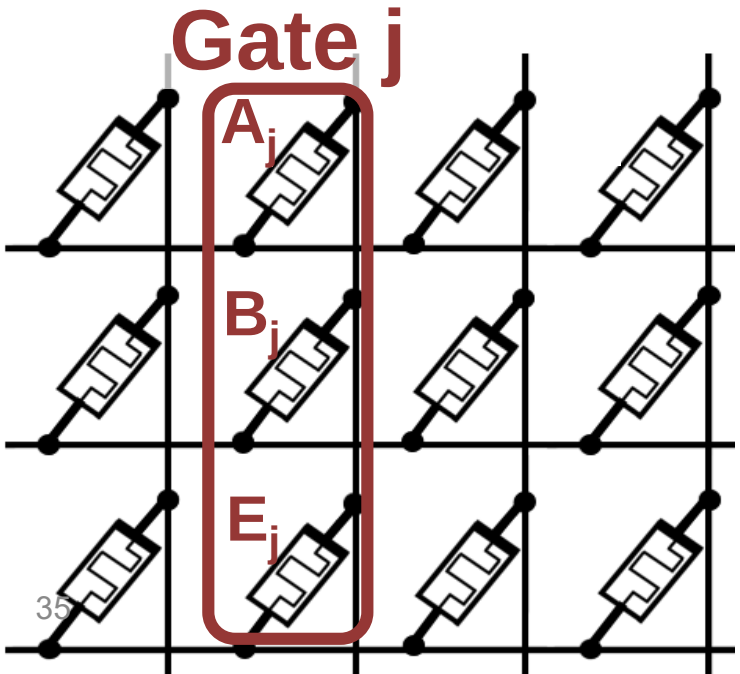
$$Latency_{best\ mapping} = min\{\max_j\{T_j\}\}$$

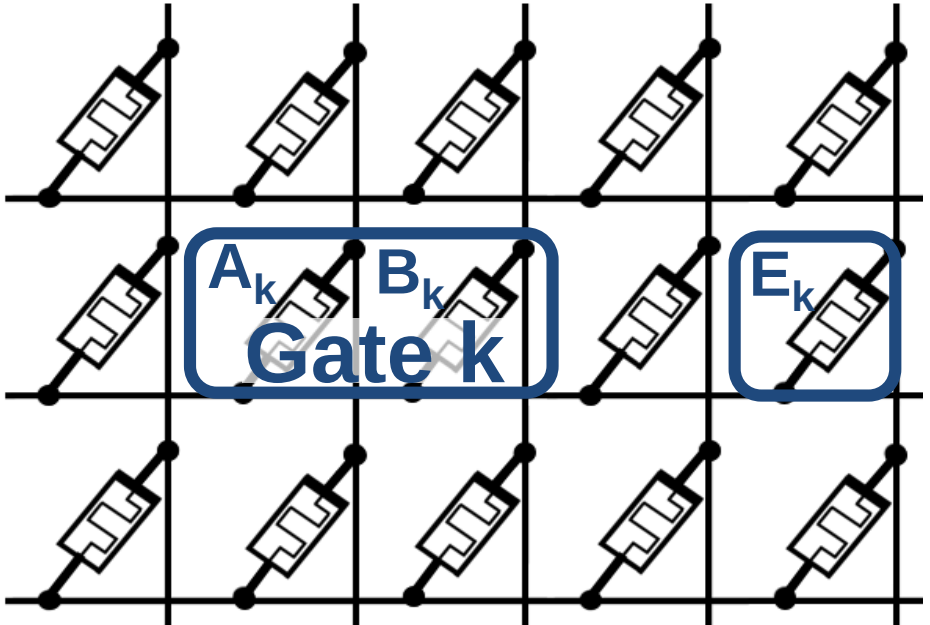
Constrains - Locations



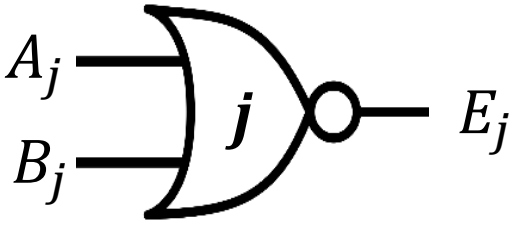
- I/Os of each gate have to be located in the same row and different columns, or vice versa
- For every gate j:

$$\left[\left(C_{A_j} = C_{B_j} = C_{E_j}\right) \cap \left(R_{A_j} \neq R_{B_j} \neq R_{E_j}\right)\right] \cup \left[\left(C_{A_j} \neq C_{B_j} \neq C_{E_j}\right) \cap \left(R_{A_j} = R_{B_j} = R_{E_j}\right)\right]$$

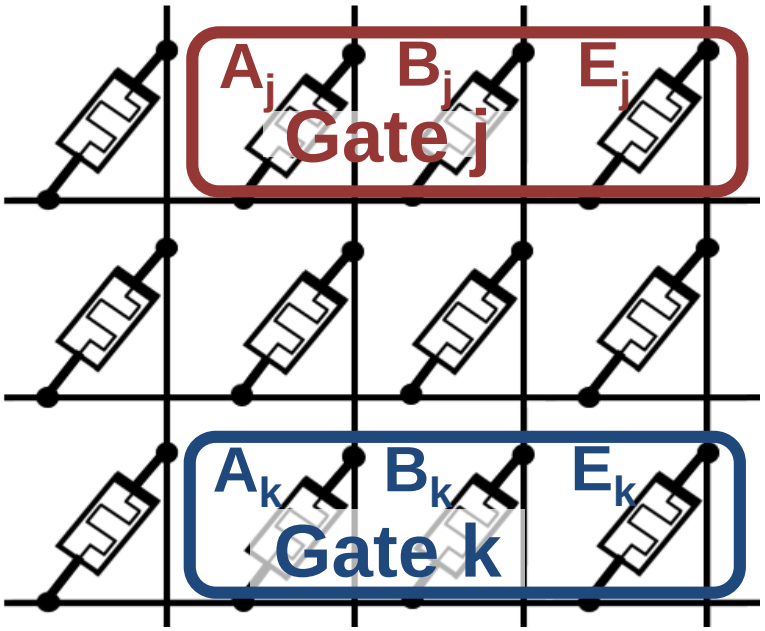


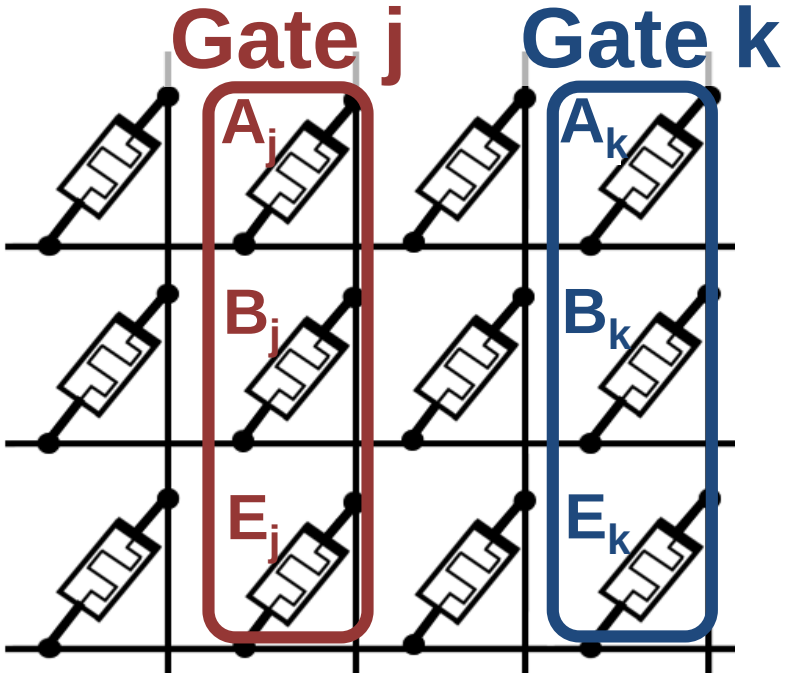


Constrains - Locations

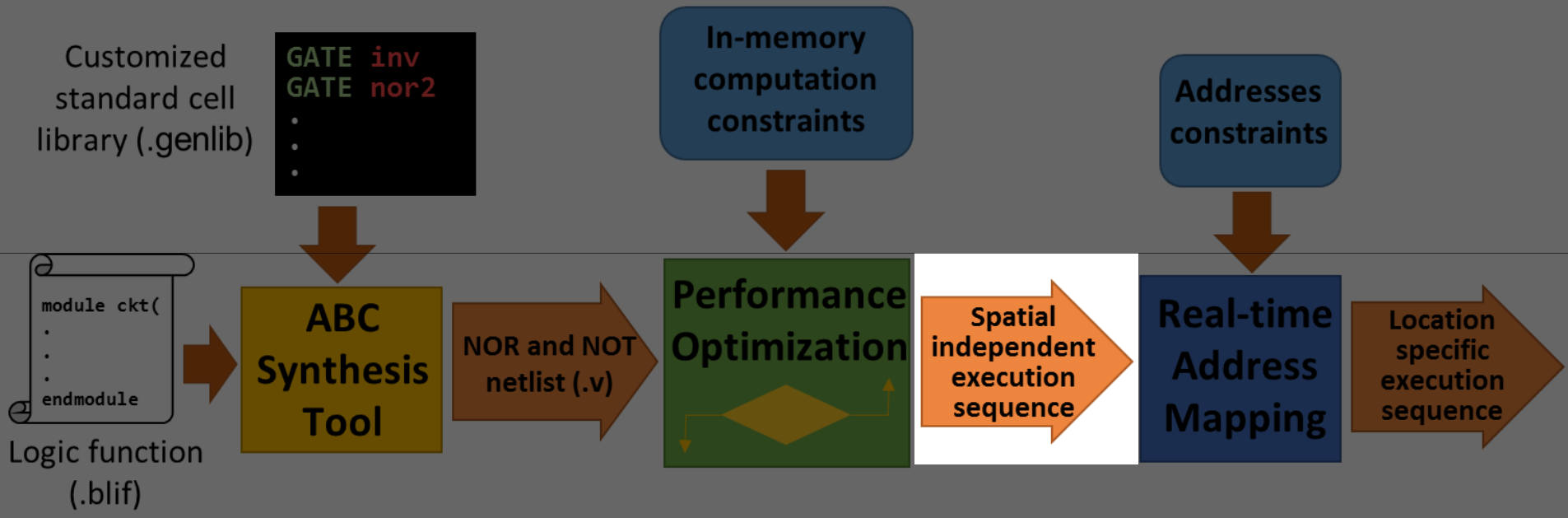


- Simultaneous execution of different gates only when they are aligned in the rows/columns



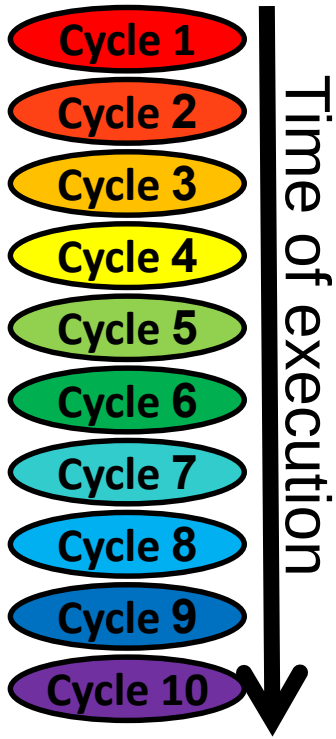


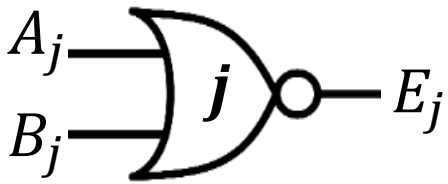
SIMPLE MAGIC



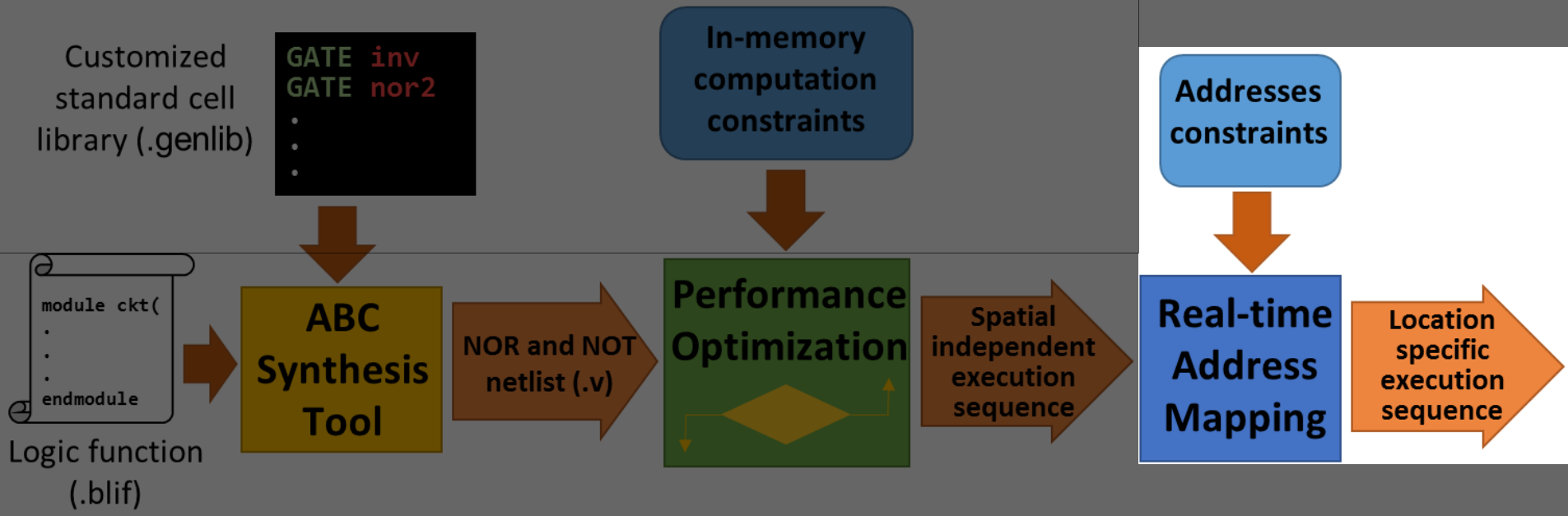
1 Bit Full Adder
Spatial Independent Execution Sequence

	column 1	column 2	column 3	column 4	column 5
row 1			C_{in} A1	E1 A8	
row 2	A A2	B A3		E7 B8	
row 3				E11	
row 4				E9 B11	
row 5				B9	
row 6				E6 A7 A9	
row 7	A A5	B B5		E5 B6	
row 8	E2 A4	E3 B4		E4 A6 A10	
row 9				E8 B10 A11 A12	E12 C_{out}
row 10				E10 S	

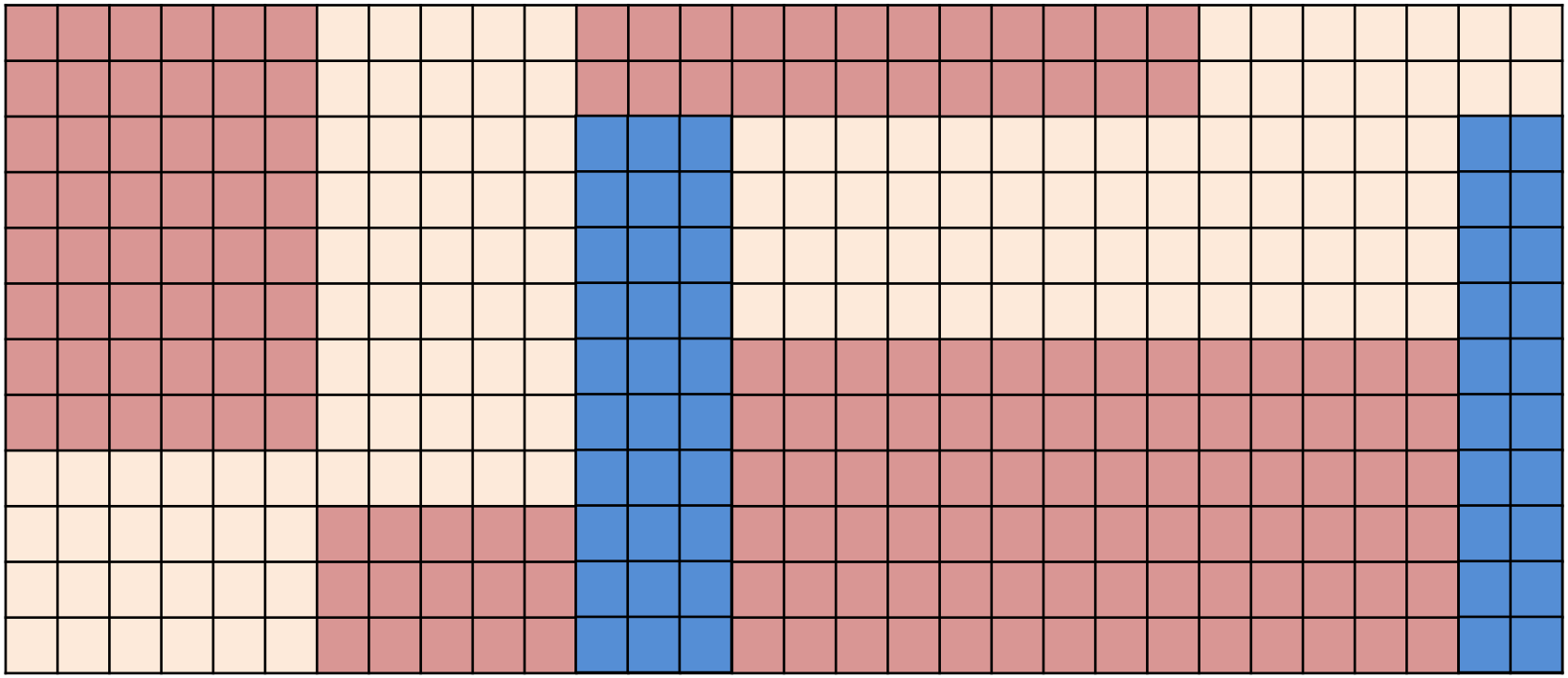




SIMPLE MAGIC



1 Bit Full Adder
Location Specific Execution Sequence

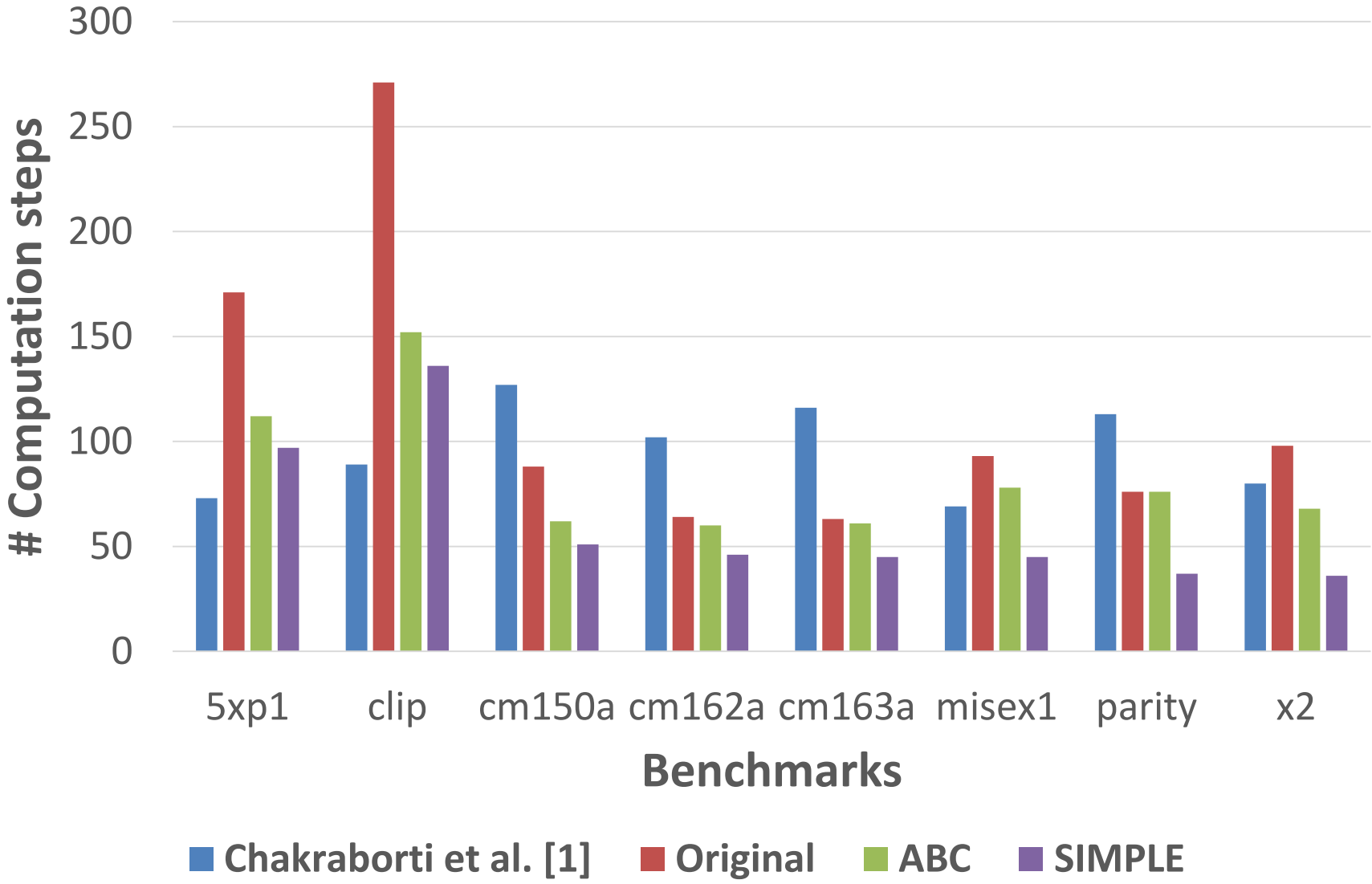


– *occupied cells*

– *available cells*

– *Location for* 1*Bit Full Adder*
(10 *rows X* 5 *columns*)

Experimental Results



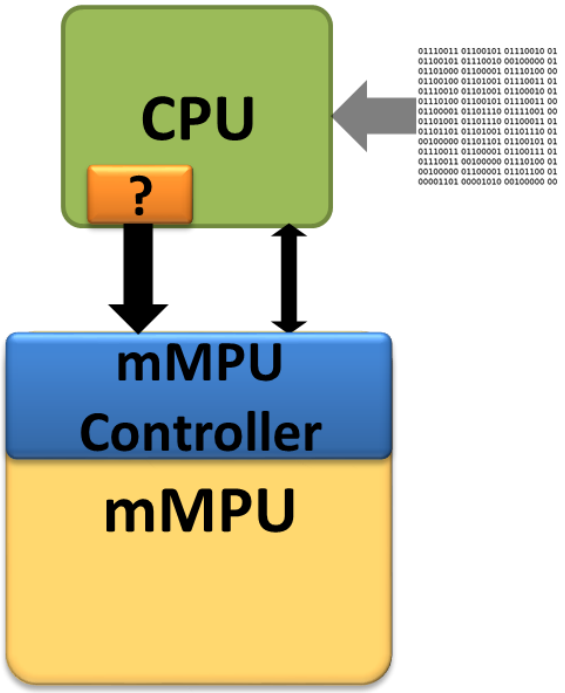
Agenda

- The need for non-von Neumann architectures
- Memristive technologies
- Memristive MPU (mMPU) architecture
- mMPU logic synthesis and automation
- **Summary**

mMPU – Huge Potential for Real Processing In-Memory

- Orders of magnitude better performance & energy
- SIMPLE MAGIC – mMPU controller design
- Current work:
  - Reducing the running time of SIMPLE

  - Extending the optimization functions
  - Real-time mapping



Thanks!

ASIC² ARCHITECTURES SYSTEMS INTELLIGENT COMPUTING INTEGRATED CIRCUITS

HiPer

משרד המדע והטכנולוגיה
Israel Ministry of Science and Technology

ISRAEL SCIENCE FOUNDATION

RBNI

TCE technion computer engineering center

ICRI-CI Intel Collaborative Research Institute Computational Intelligence

Advanced Circuit Research Center ACRC

European Research Council

רשות החדשנות Israel Innovation Authority

CISCO

BSF United States – Israel Binational Science Foundation