institute of
neuroinformatics

University of Zurich^UZH

ETHzürich

Brain-Inspired Principles for Scalable and Energy-Efficient Embedded Computing

Melika Payvand

Feb. 9th, 2026

EFCL Winter School 2026

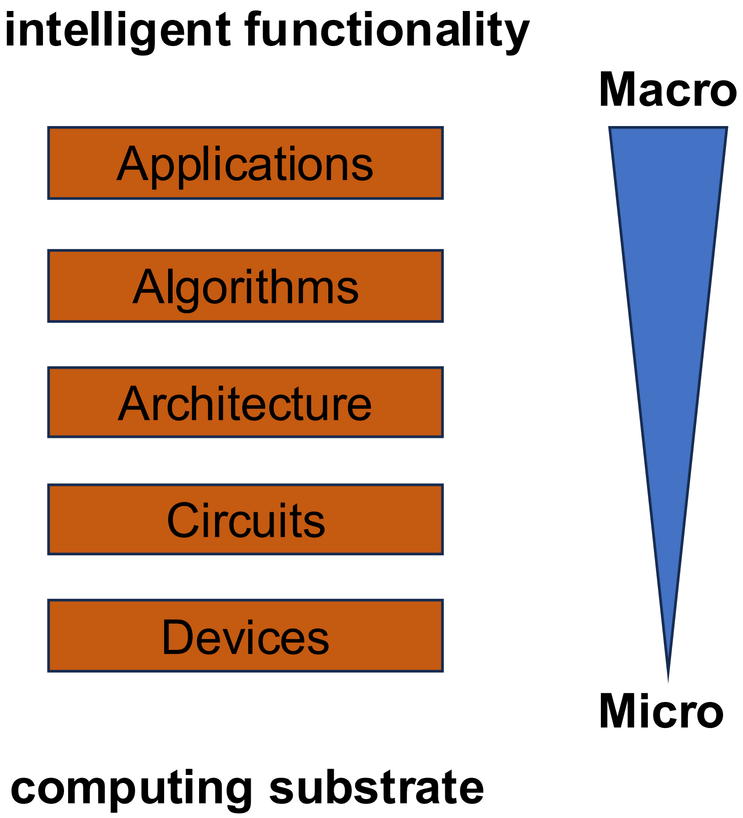
Zurich

Emerging Intelligent Substrates (EIS) lab

Q: How does the **intelligent functionality** relate to its underlying **computing substrate**?

Emerging Intelligent Substrates (EIS) lab

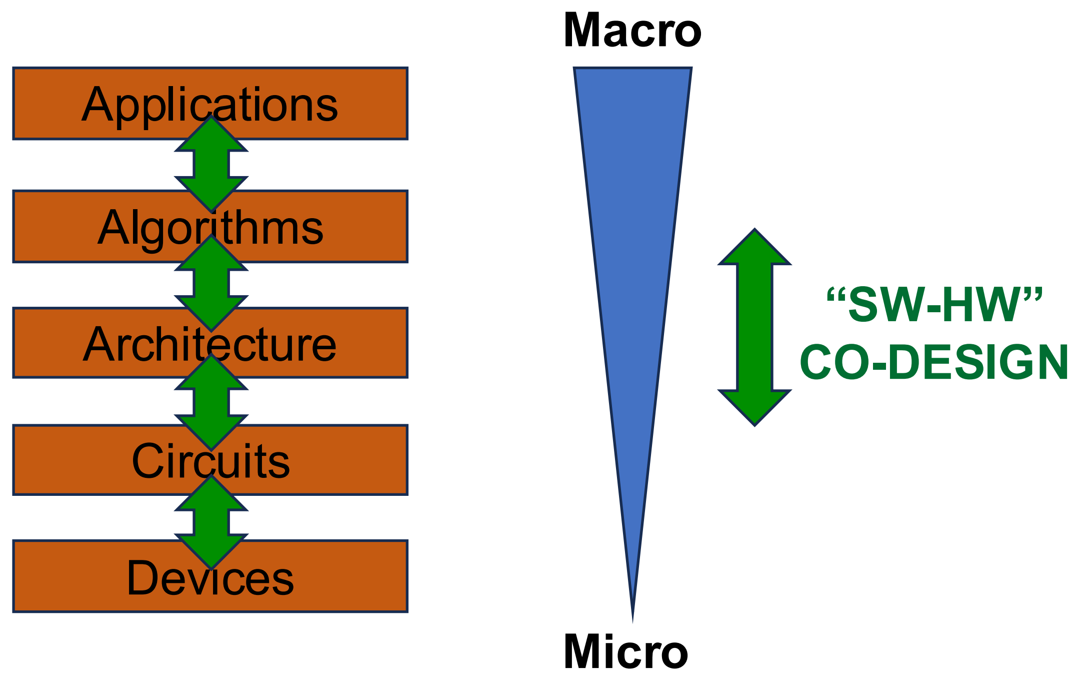
Q: How does the **intelligent functionality** relate to its underlying **computing substrate**?



Emerging Intelligent Substrates (EIS) lab

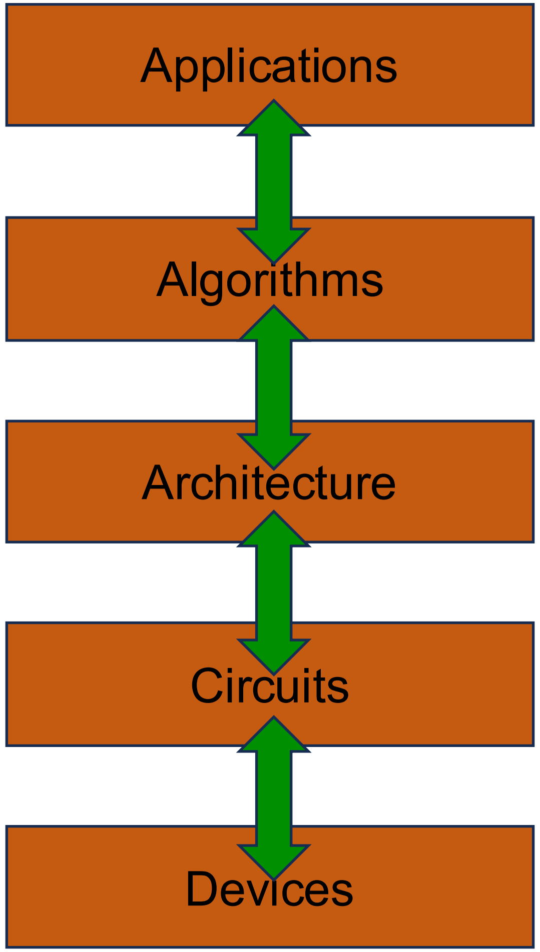
Q: How does the **intelligent functionality** relate to its underlying **computing substrate**?

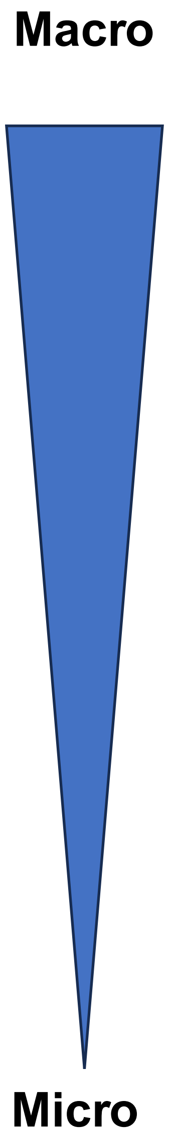
intelligent functionality



computing substrate

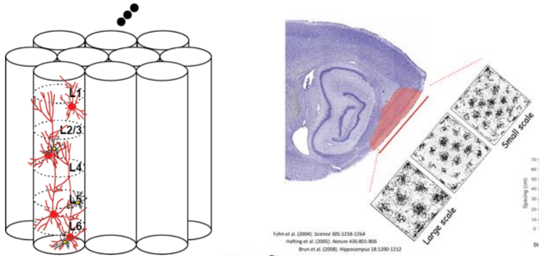
Emerging Intelligent Substrates (EIS) lab





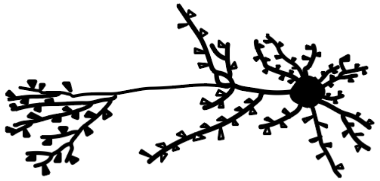
Animal behavior:
Sensing, perception,
action

Brain regions and
their organization

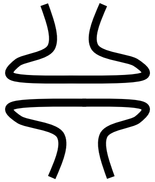


Sparse network arch.

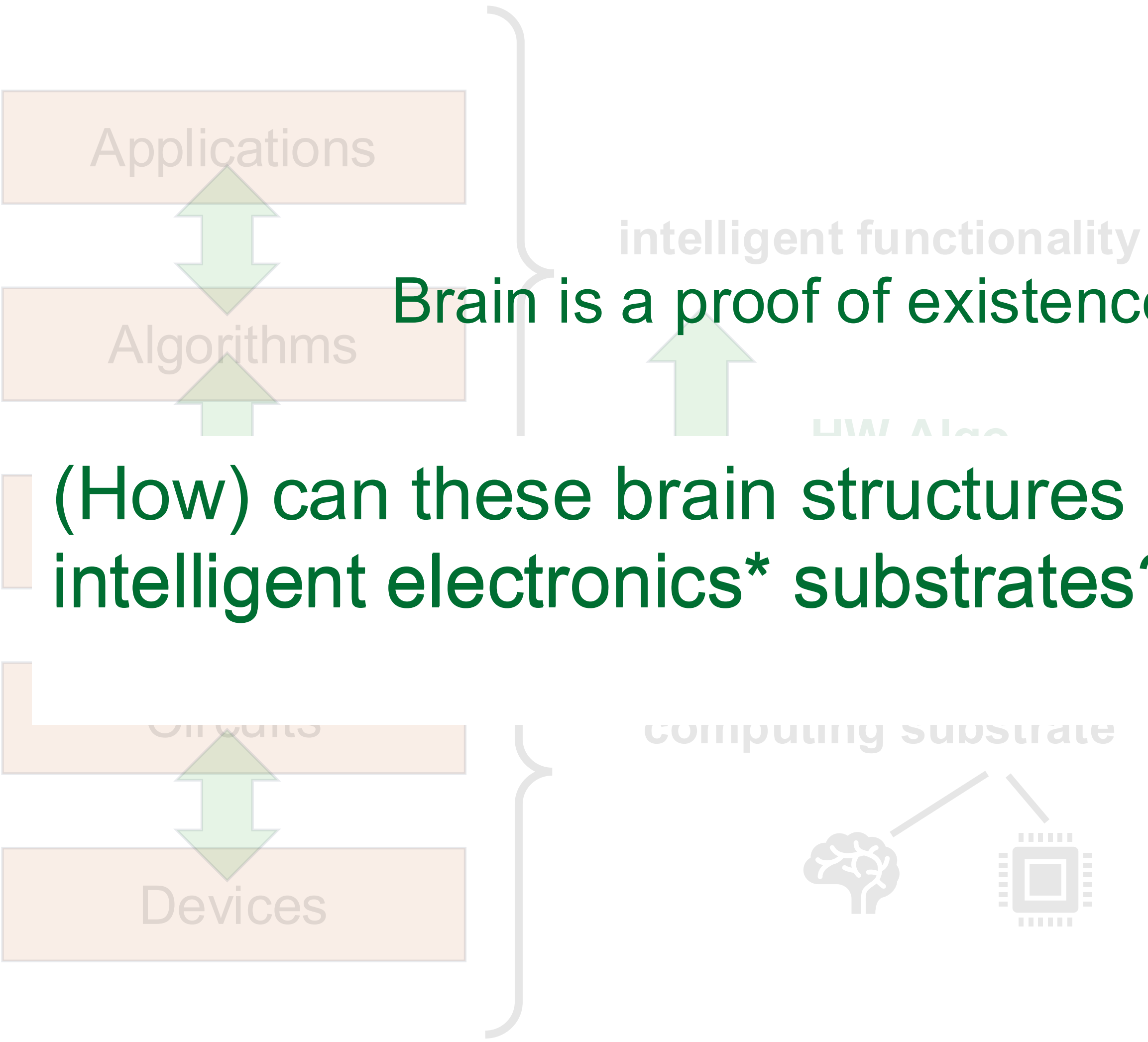
Dendritic feature detection

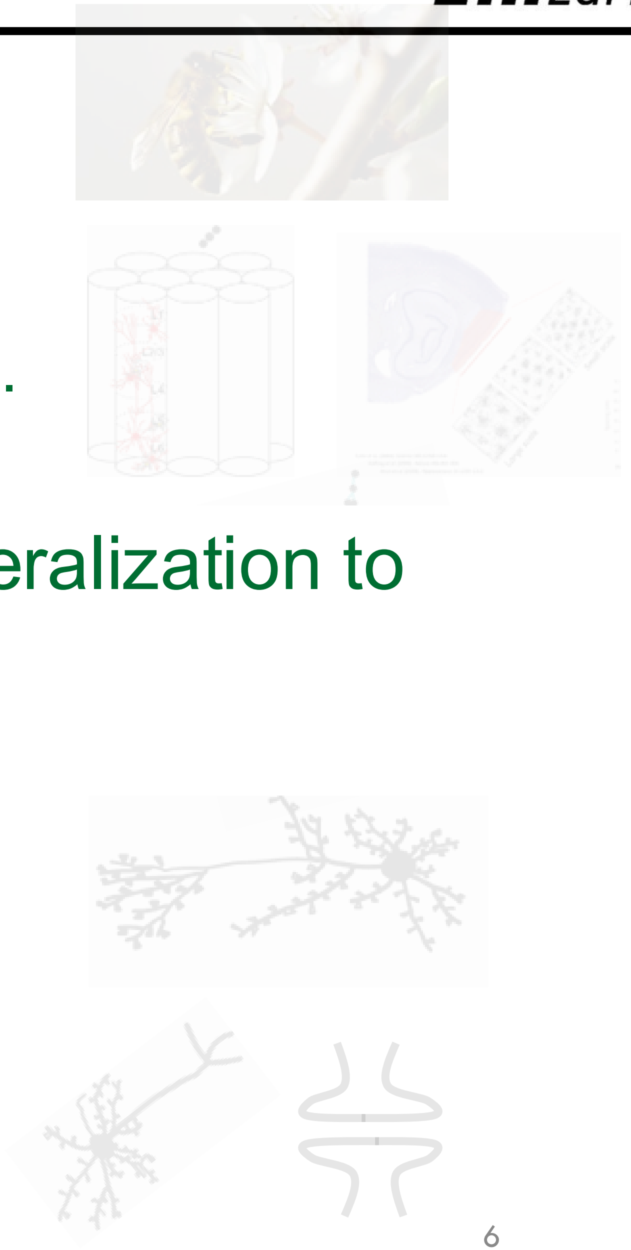


Synaptic filtering



Emerging Intelligent Substrates (EIS) lab



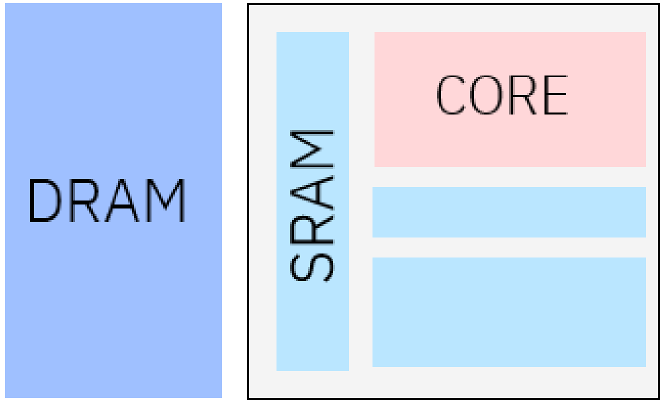


Brain is a proof of existence for intelligent systems.

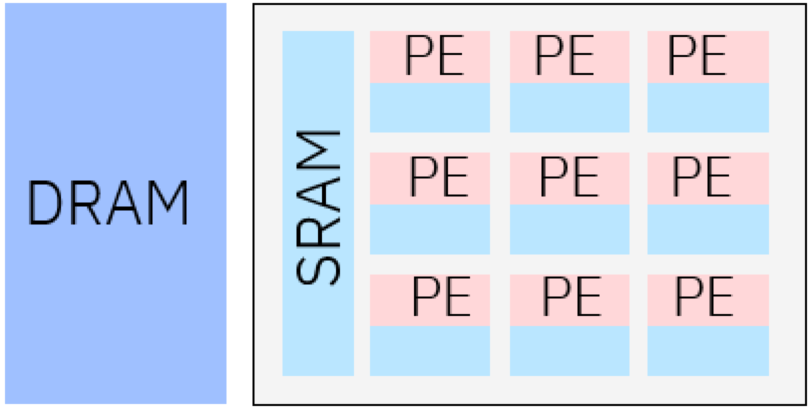
(How) can these brain structures bring efficiency/generalization to intelligent electronics* substrates?

Intelligent electronics* substrates progress over time

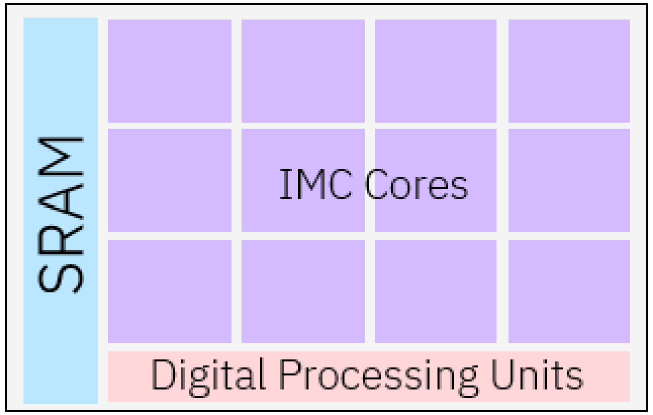
Conventional computers



Custom DL accelerators



In-memory computing (IMC)-based DL accelerators



Brains

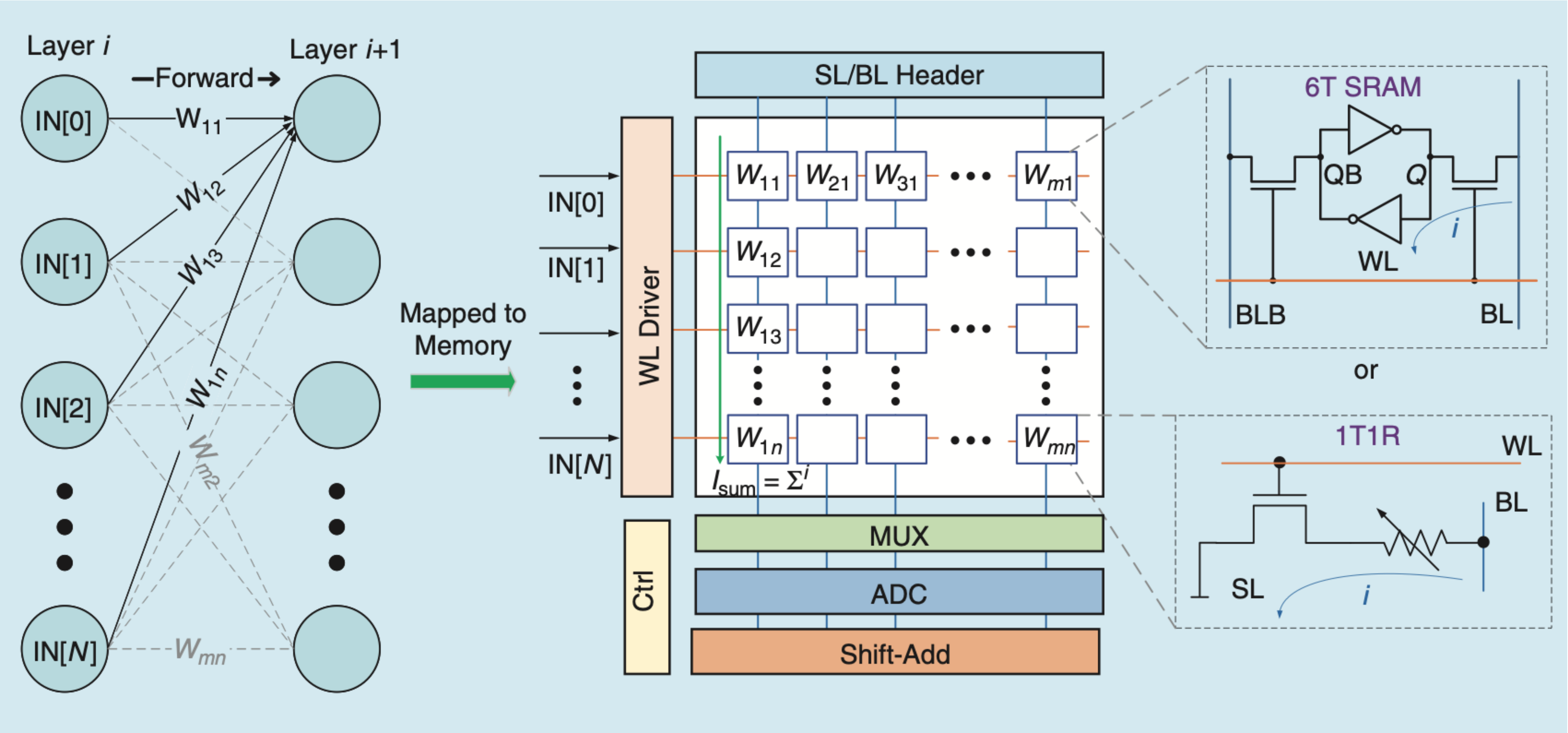
- **Parallelism**
- **Less precise arithmetic**

- **Massive parallelism**
- **Less precise (potentially analog) arithmetic**
- **In-place computing**

We have been progressively incorporating brain-inspired attributes

Taken from Abu Sebastian's slides- ISCAS 2025

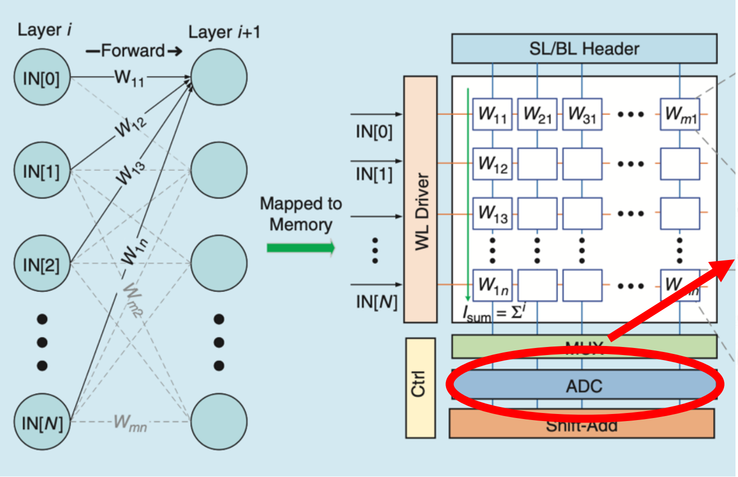
“Intelligent” systems: current challenges on the system level



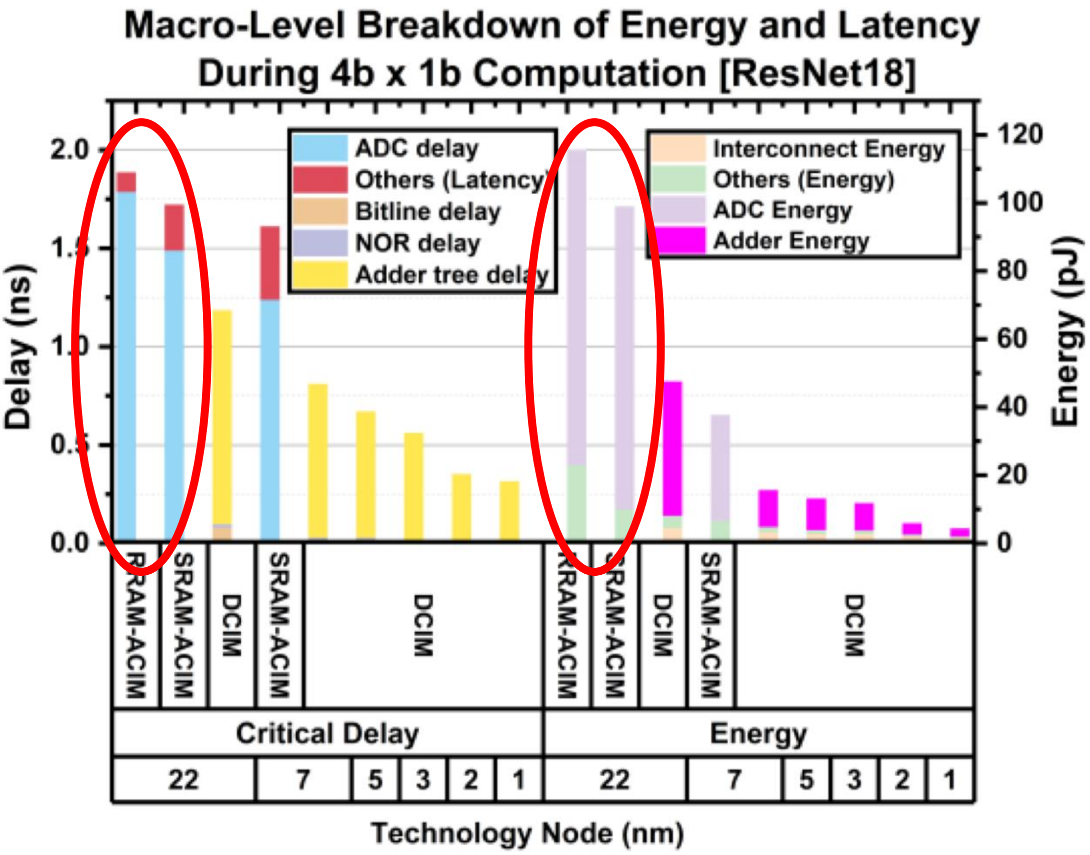
S. Yu, et al, IEEE CAS magazine, 2021

“Intelligent” systems: current challenges on the system level

1) IMC periphery overhead is ex



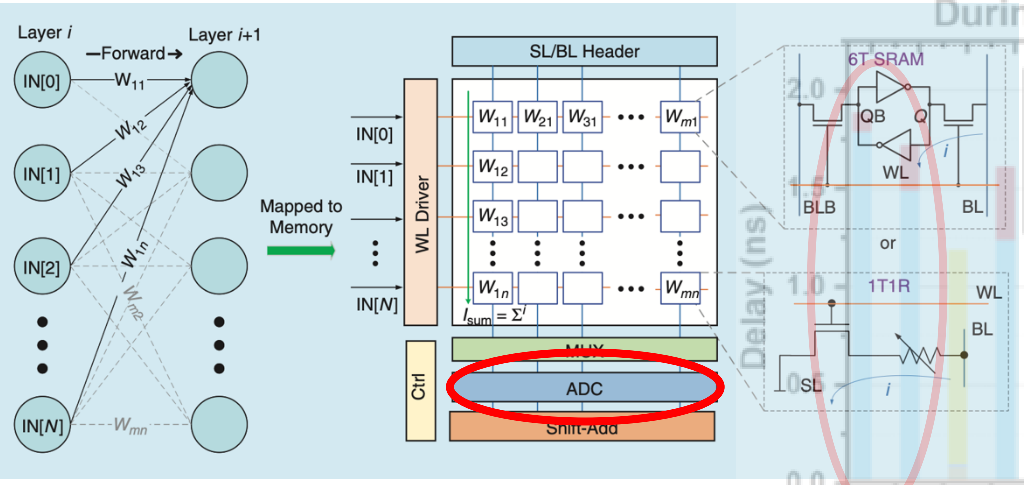
S. Yu, et al, IEEE CAS magazine, 2021



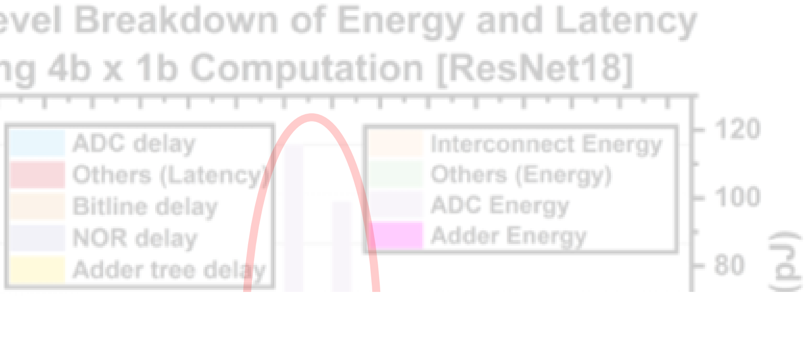
Lee et al, IEEE TCAS I 2024

“Intelligent” systems: current challenges on the system level

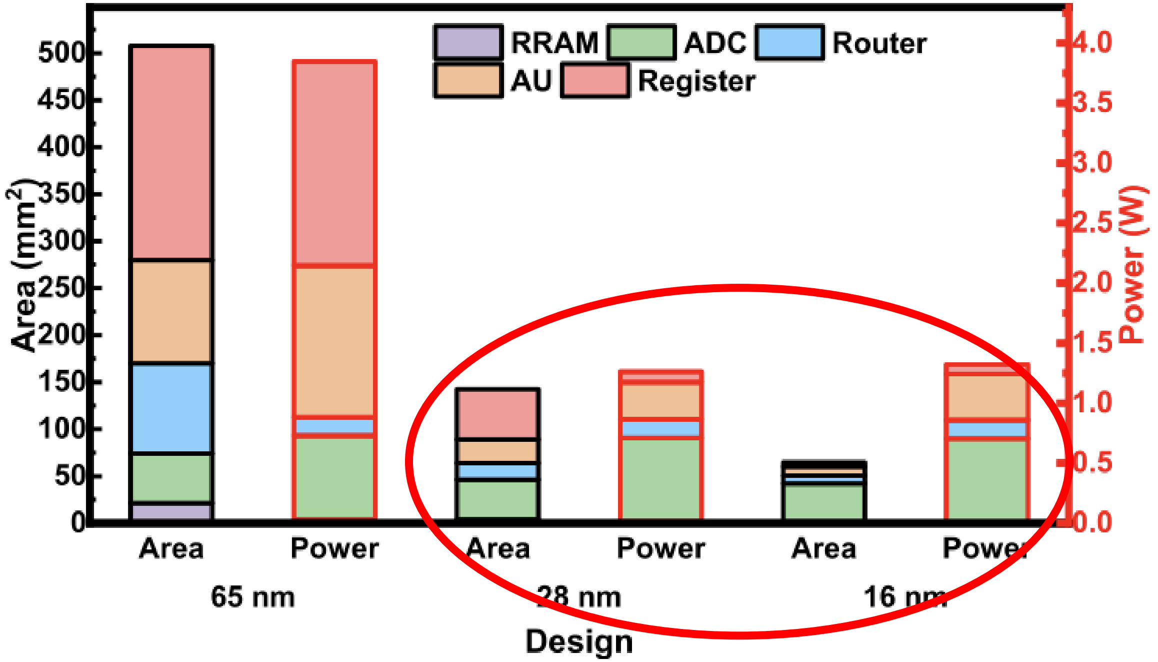
1) IMC periphery overhead is expensive.



S. Yu, et al, IEEE CAS magazine, 2021



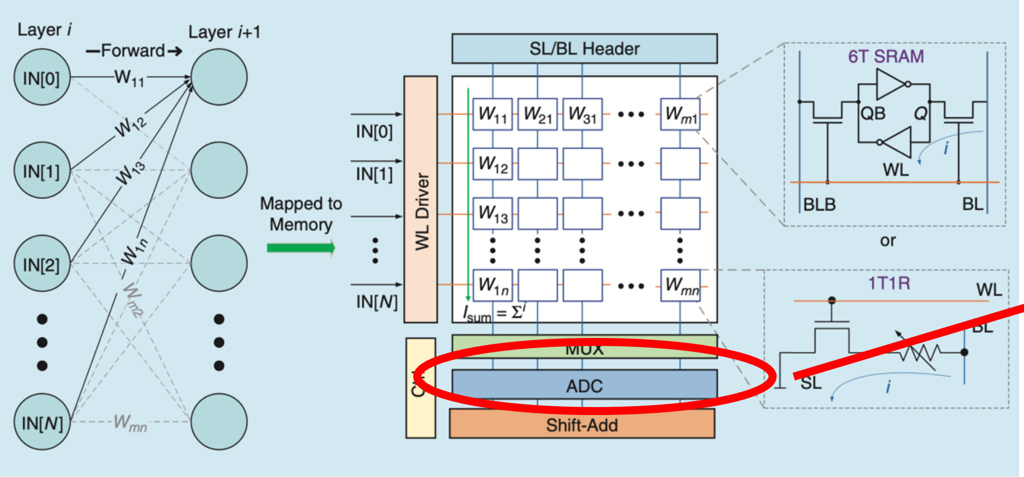
Lee et al, IEEE TC



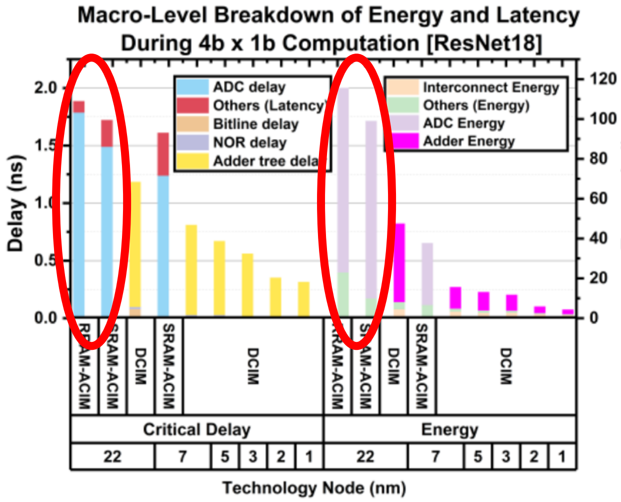
X. Wang et al, IEEE TCAS II 2022

“Intelligent” systems: current challenges on the system level

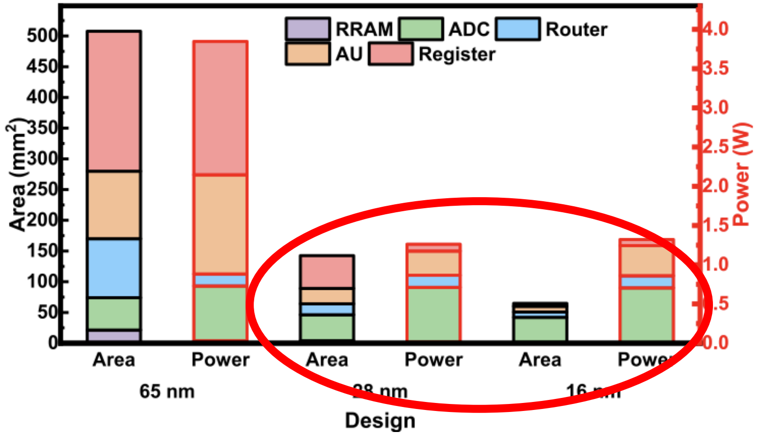
1) IMC periphery overhead is expensive.



S. Yu, et al, IEEE CAS magazine, 2021



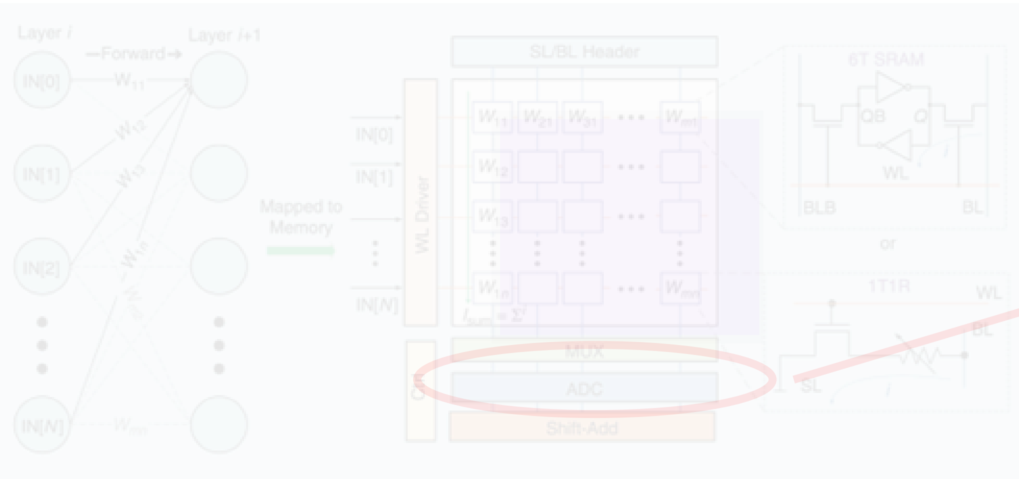
Lee et al, IEEE TCAS I 2024



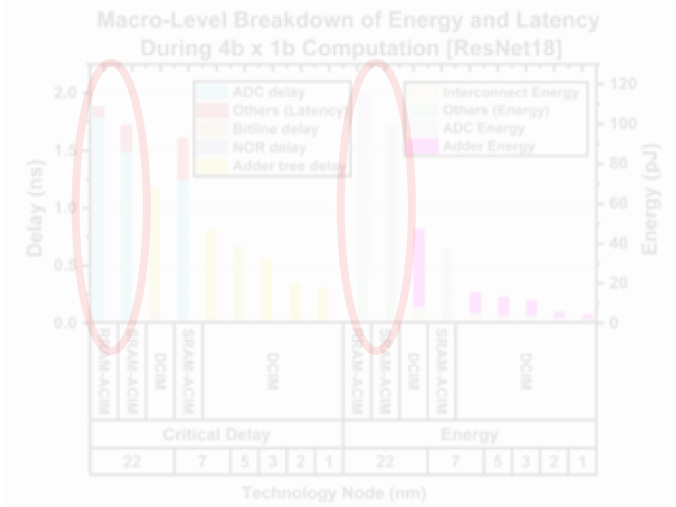
X. Wang et al, IEEE TCAS II 2022

“Intelligent” systems: current challenges on the system level

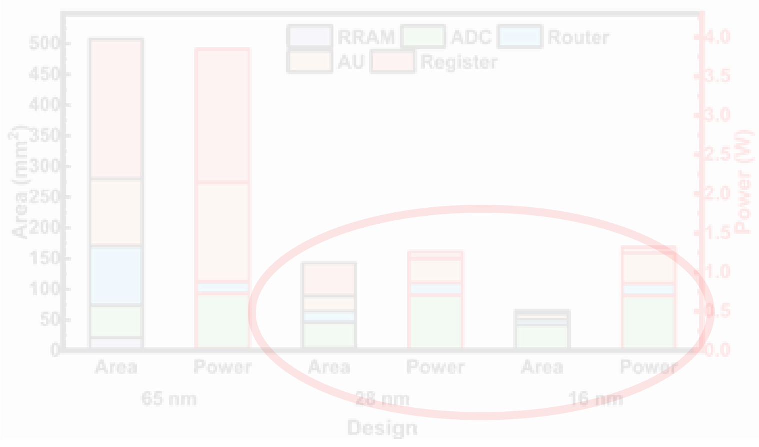
1) IMC periphery overhead is expensive.



S. Yu, et al, IEEE CAS magazine, 2021

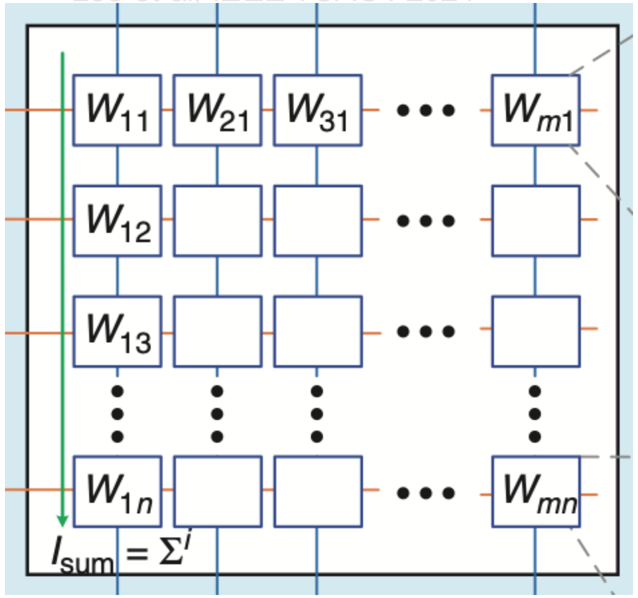


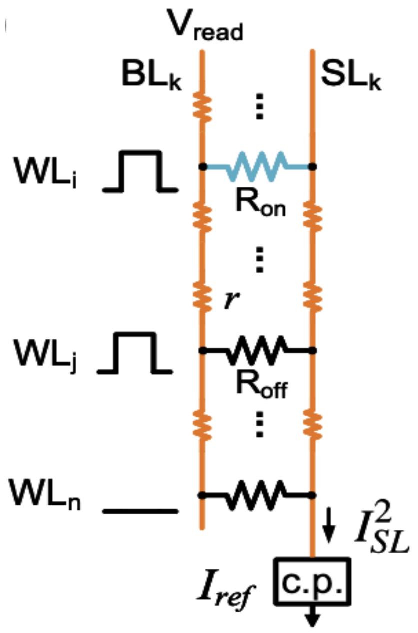
Lee et al, IEEE TCAS I 2024



2022

Also…IMC array size is limited due to parasitics.





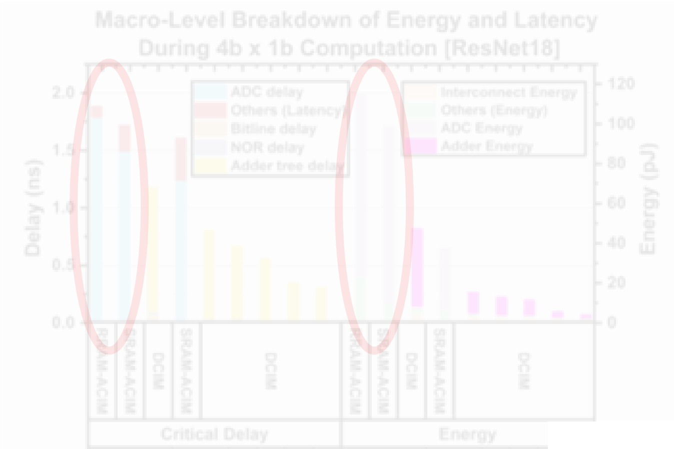
J. Chen, et al, IEEE TCAS II, 2023

“Intelligent” systems: current challenges on the system level

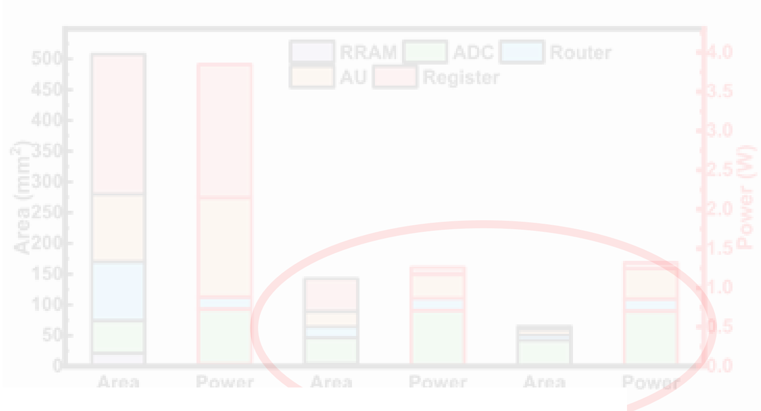
1) IMC periphery overhead is expensive.

Layer i
Forward
Layer i+1
IN[0]
IN[1]
IN[2]
IN[N]
W_{11}
W_{12}
W_{13}
W_{1n}
W_{m2}
W_{mn}
Mapped to Memory
IN[0]
IN[1]
IN[N]
WL Driver
SL/BL Header
W_{11}
W_{21}
W_{31}
W_{m1}
W_{12}
W_{13}
W_{1n}
W_{mn}
$I_{sum} = \Sigma I$
MUX
ADC
Shift-Add
Ctrl
6T SRAM
QB
Q
WL
BLB
BL
or
1T1R
WL
BL
SL

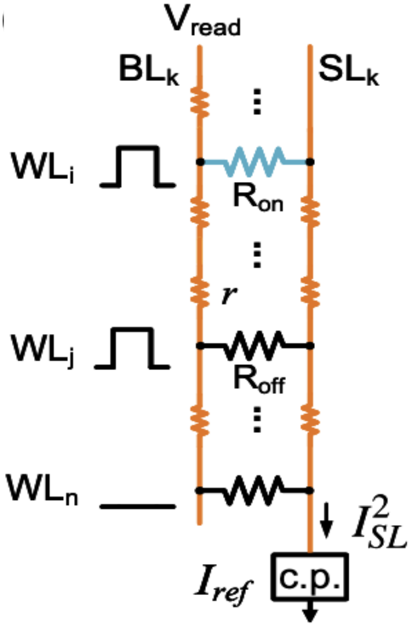
S. Yu, et al, IEEE CAS magazine, 2021

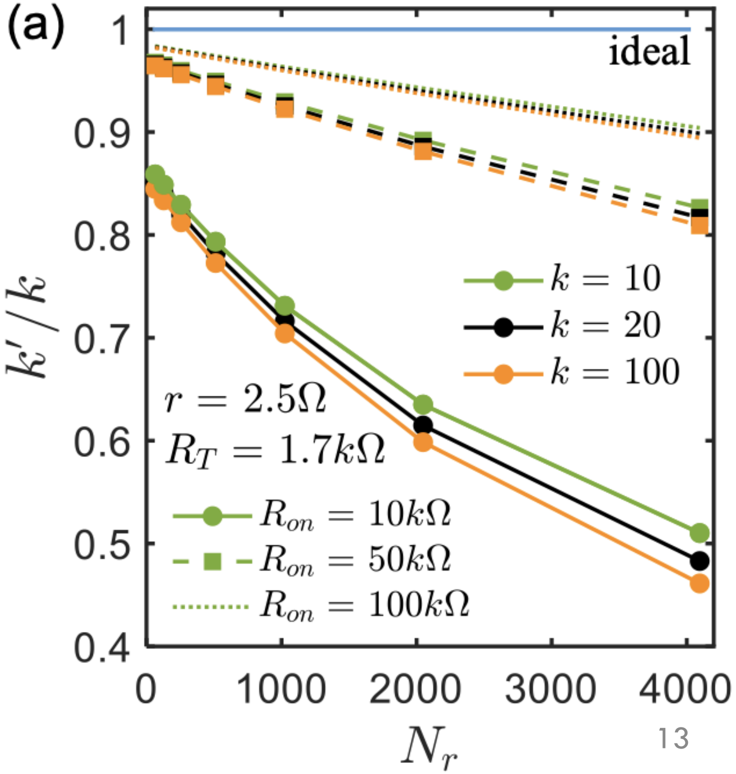


2024



Also…IMC array size is limited due to parasitics.





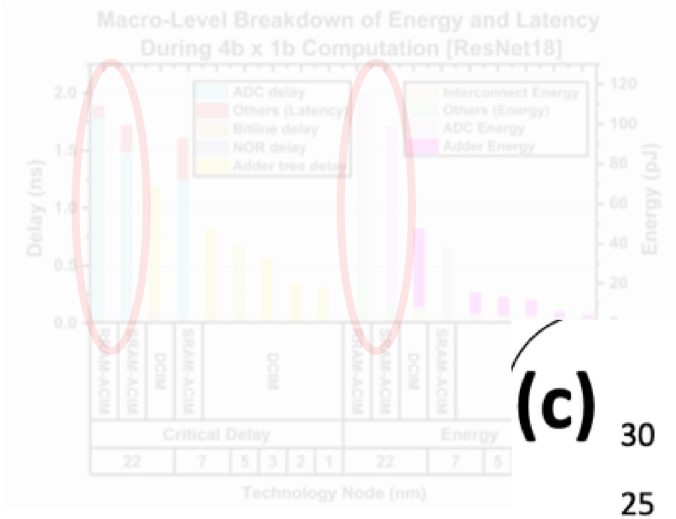
J. Chen, et al, IEEE TCAS II, 2023

“Intelligent” systems: current challenges on the system level

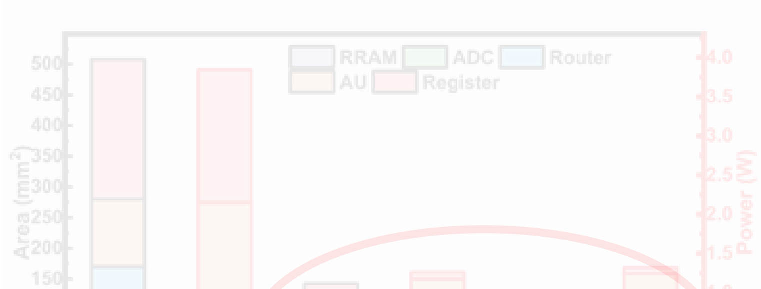
1) IMC periphery overhead is expensive.

Layer i
Layer i+1
Forward
IN[0]
IN[1]
IN[2]
IN[N]
Mapped to Memory
SL/BL Header
WL Driver
MUX
ADC
Shift-Add
6T SRAM
QB
Q
WL
BLB
BL
or
1T1R
WL
BL
SL

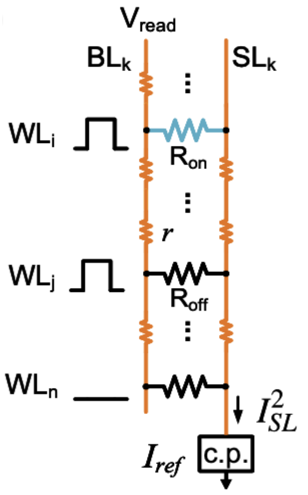
S. Yu, et al, IEEE CAS magazine, 2021

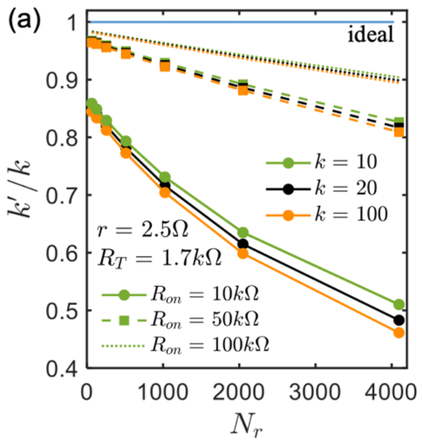


Lee et al, IEEE TCAS I 2024

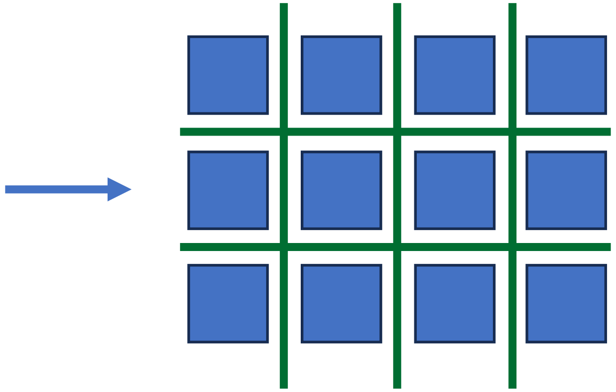


Also…IMC array size is limited due to parasitics.

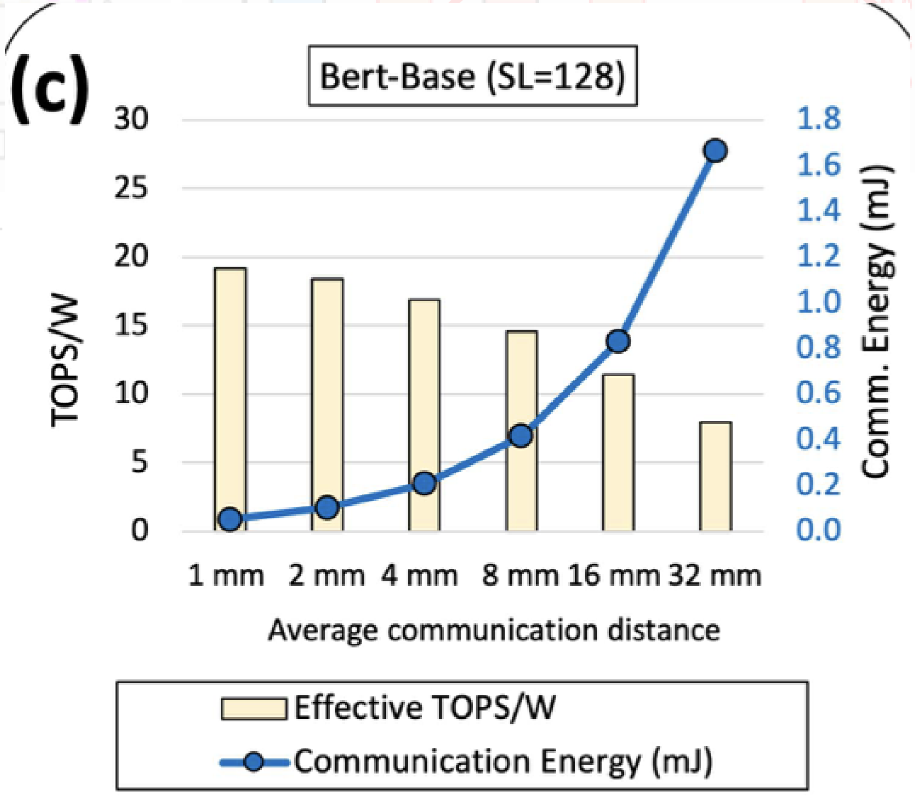




J. Chen, et al, IEEE TCAS II, 2023



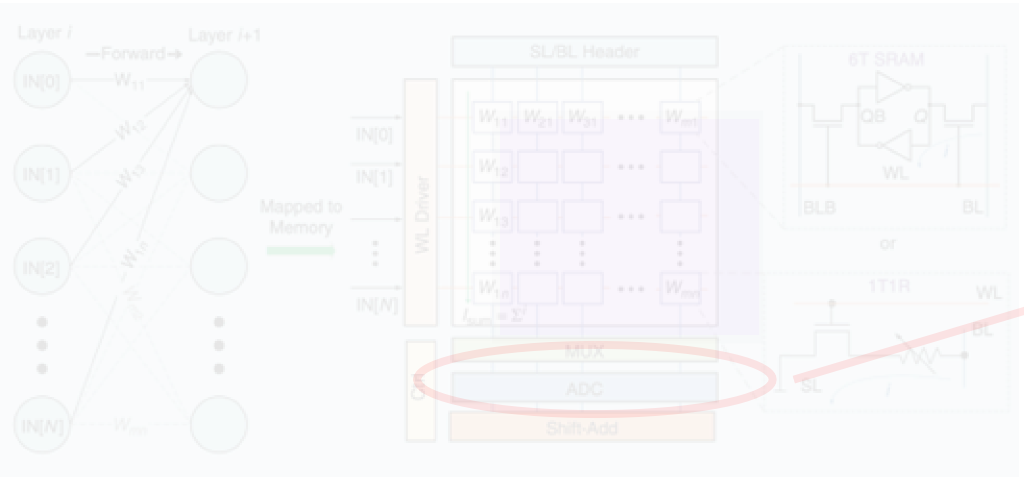
Multi-core tiled
IMC architectures.



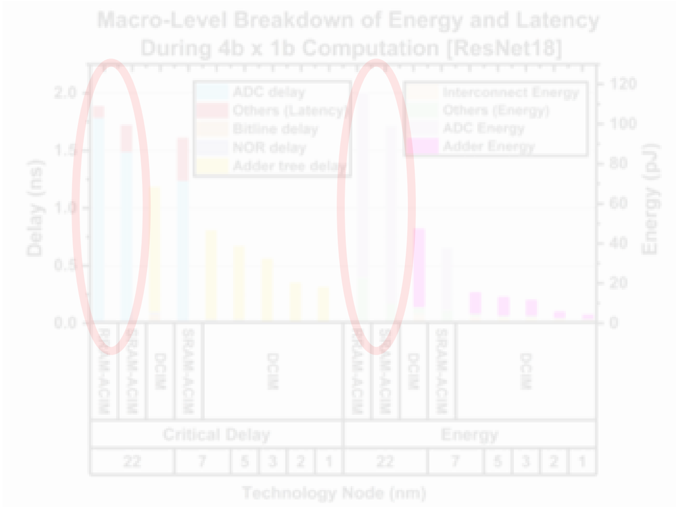
S. Jain et al, IEEE TVLSI 2023

“Intelligent” systems: current challenges on the system level

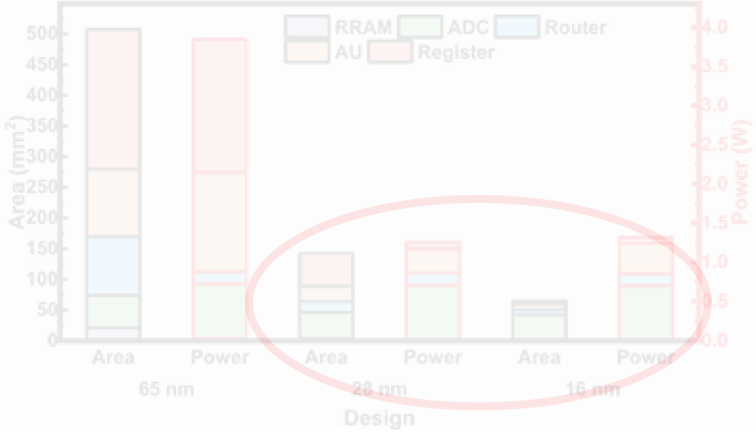
1) IMC periphery overhead is expensive.



S. Yu, et al, IEEE CAS magazine, 2021

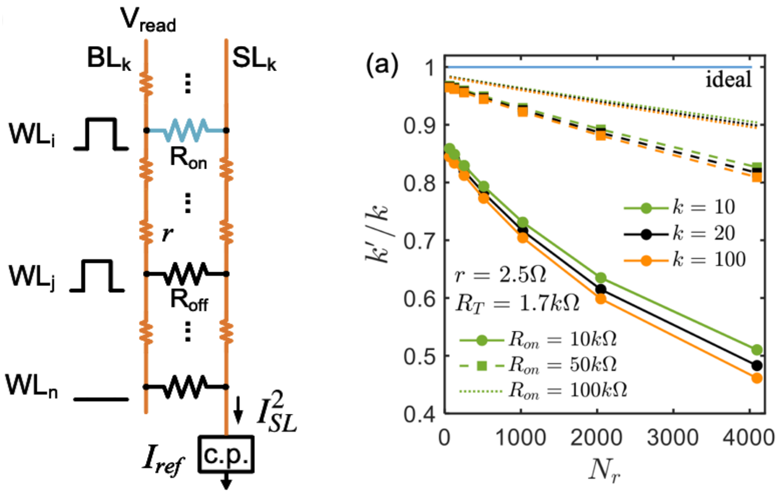


Lee et al, IEEE TCAS I 2024

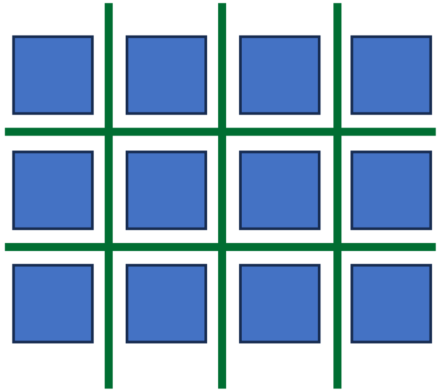


X. Wang et al, IEEE TCAS II 2022

Also…IMC array size is limited due to parasitics.

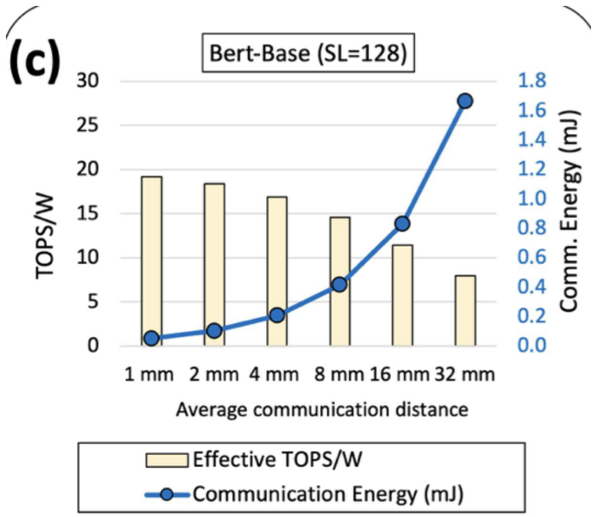


J. Chen, et al, IEEE TCAS II, 2023



Multi-core tiled IMC architectures.

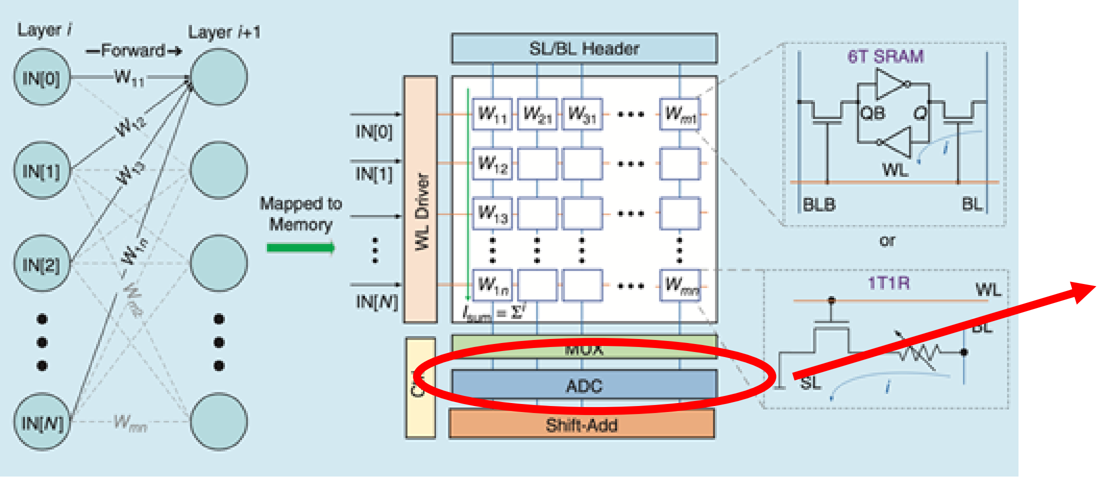
2) But… data moves again in routers…



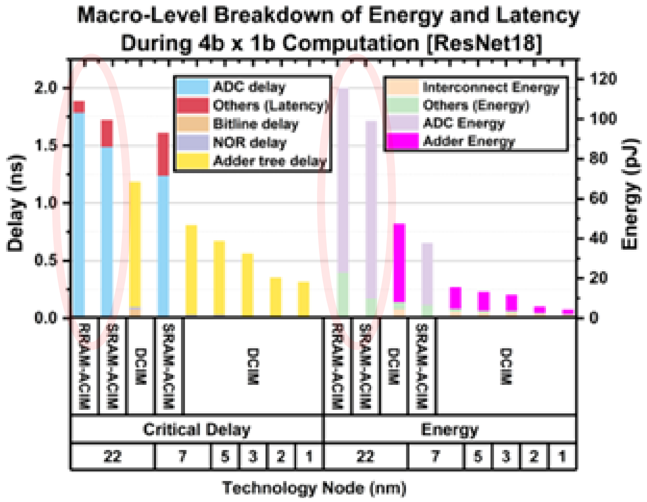
S. Jain et al, IEEE TVLSI 2023

“Intelligent” systems: current challenges on the system level

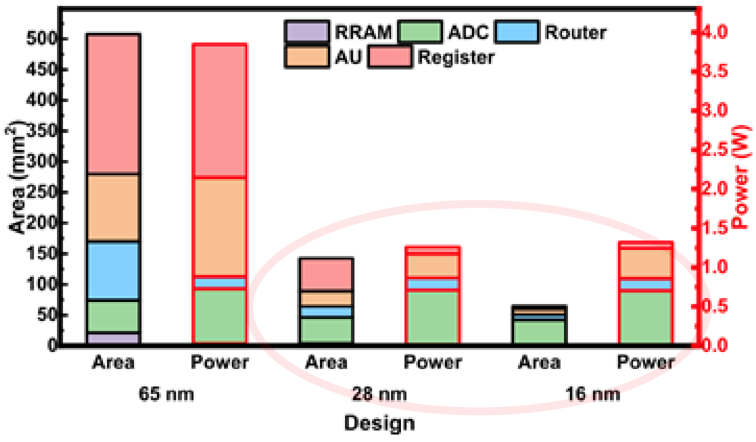
1) IMC periphery overhead is expensive.



S. Yu, et al, IEEE CAS magazine, 2021

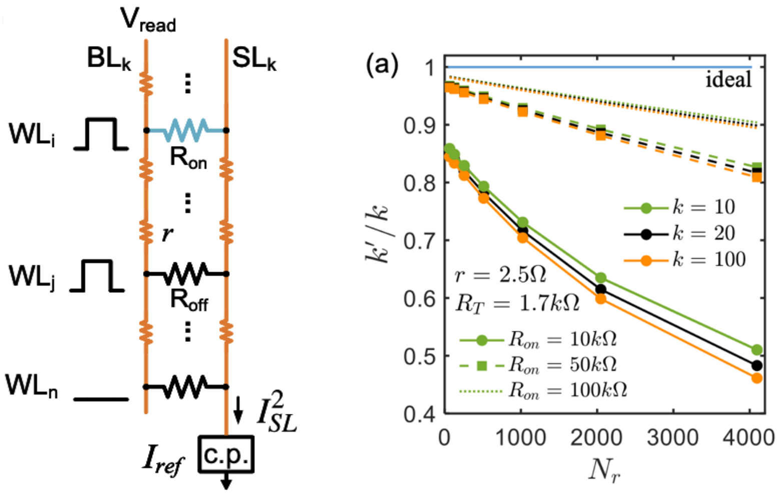


Lee et al, IEEE TCAS I 2024

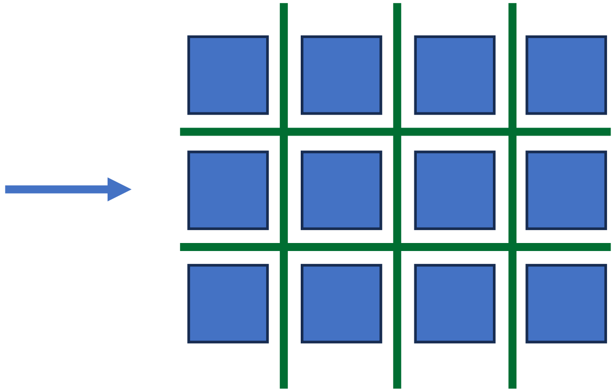


X. Wang et al, IEEE TCAS II 2022

Also…IMC array size is limited due to parasitics.

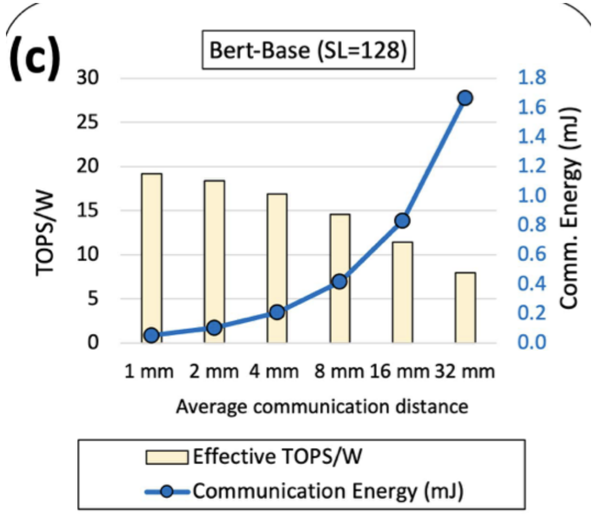


J. Chen, et al, IEEE TCAS II, 2023



Multi-core tiled IMC architectures.

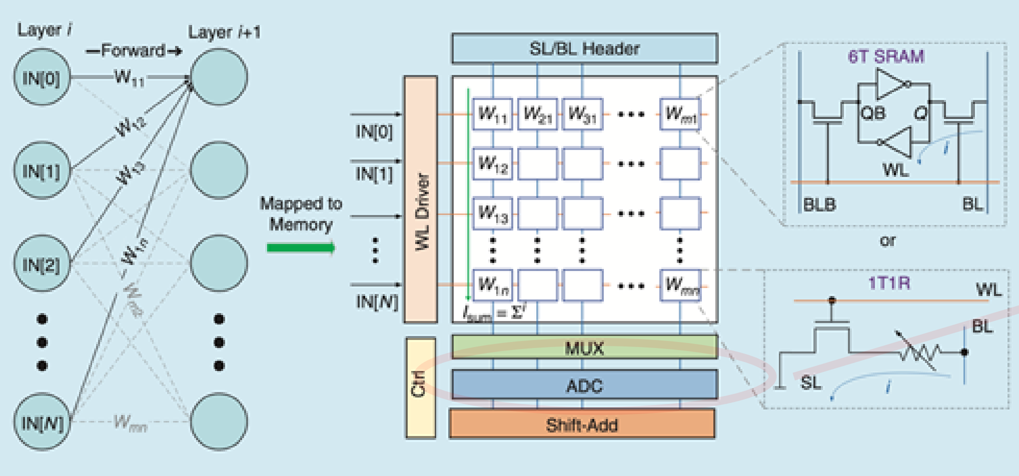
2) But… data moves again in routers…

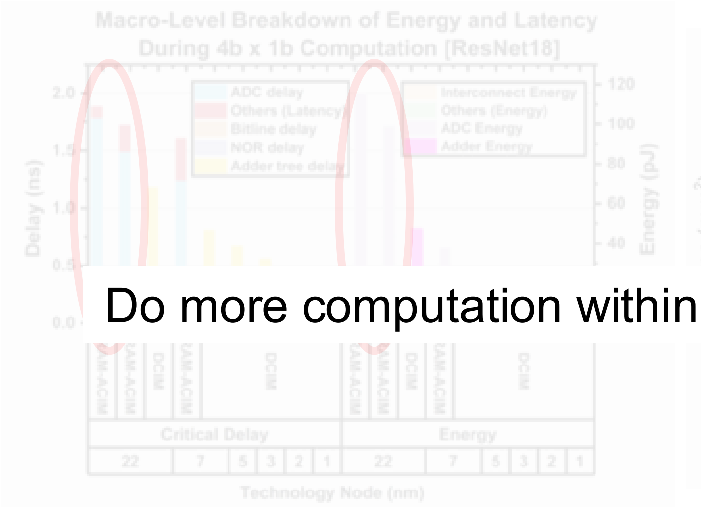


S. Jain et al, IEEE TVLSI 2023

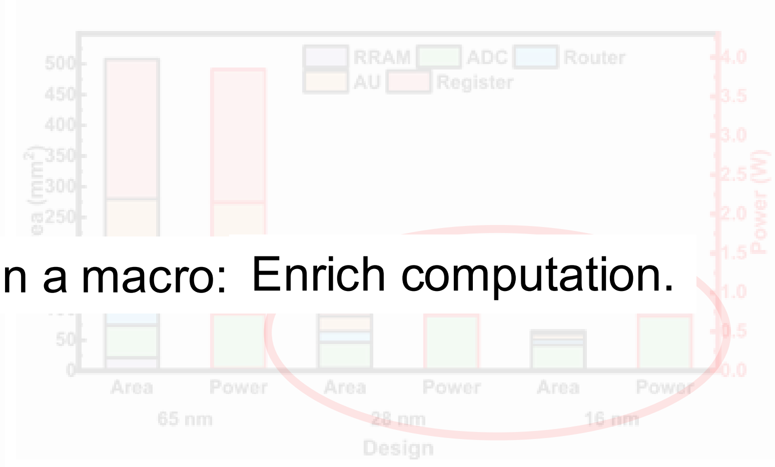
“Intelligent” systems: current challenges on the system level

1) IMC periphery overhead is expensive.





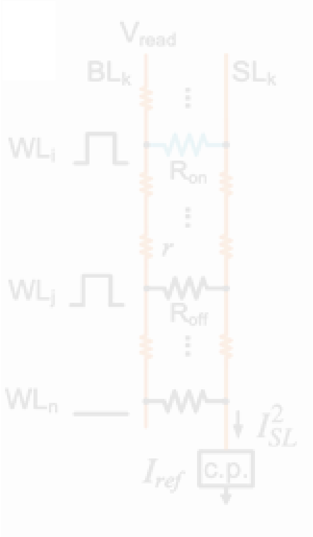
Lee et al, IEEE TCAS I 2024

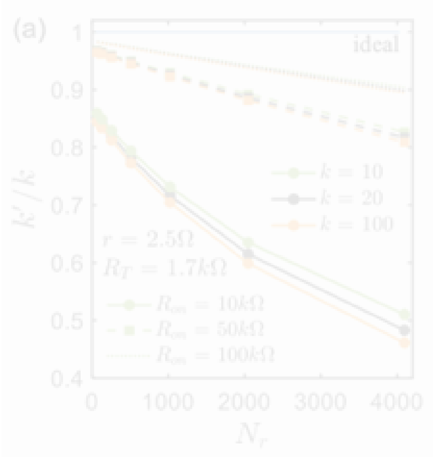


X. Wang et al, IEEE TCAS II 2022

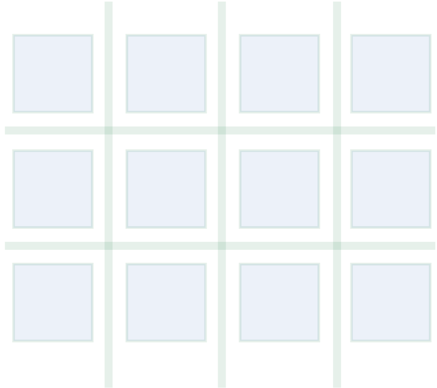
Do more computation within a macro: Enrich computation.

Also…IMC array size is limited due to parasitics.



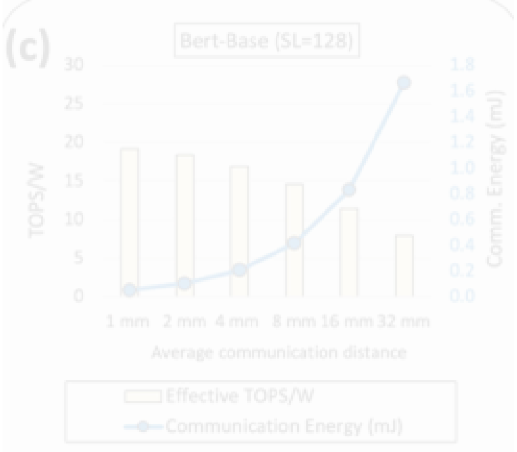


J. Chen, et al, IEEE TCAS II, 2023



Multi-core tiled
IMC architectures.

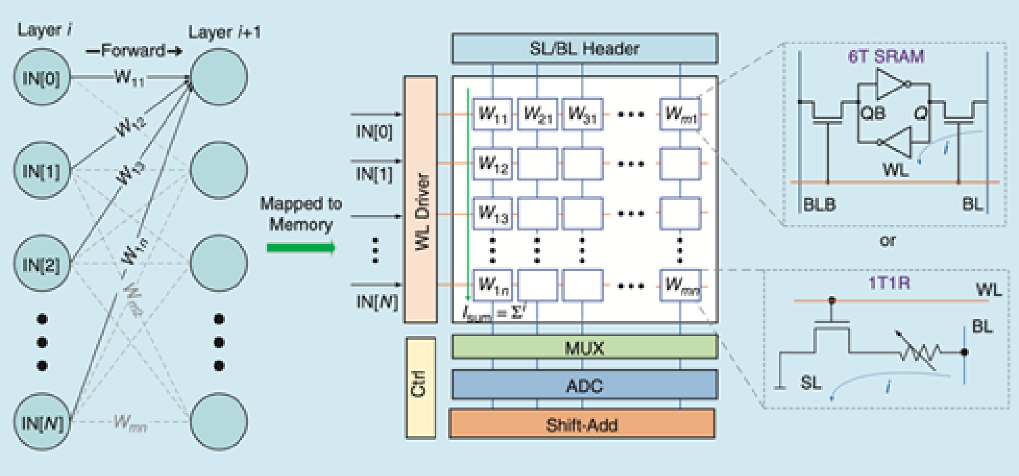
2) But… data moves again in routers…

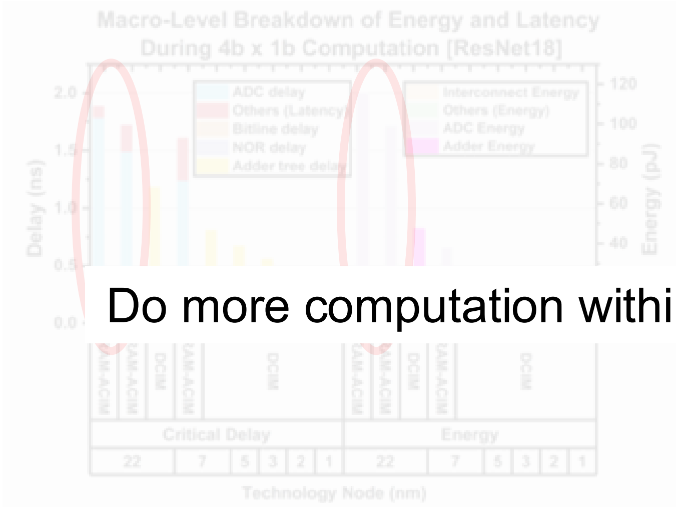


S. Jain et al, IEEE TVLSI 2023

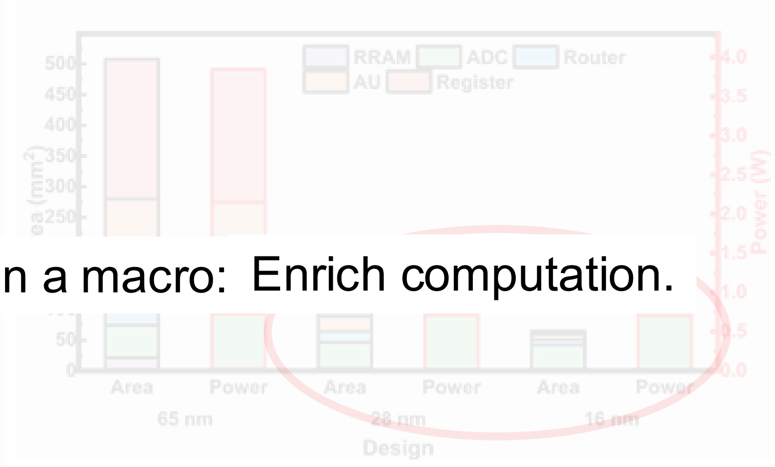
“Intelligent” systems: current challenges on the system level

1) IMC periphery overhead is expensive.





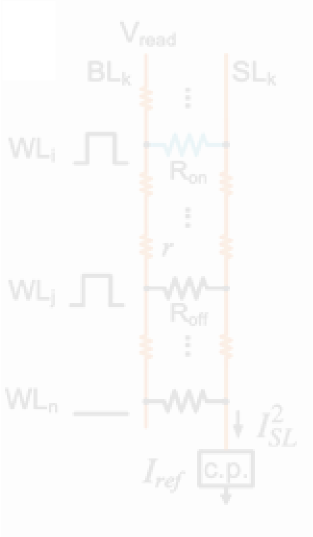
Lee et al, IEEE TCAS I 2024

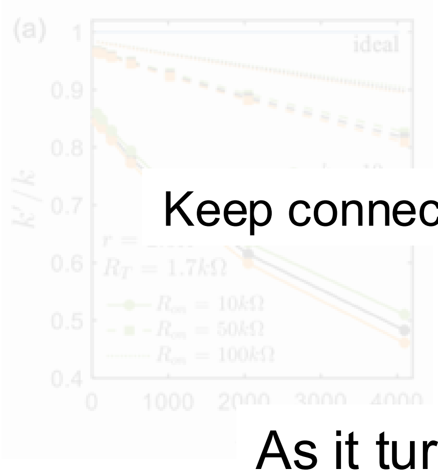


X. Wang et al, IEEE TCAS II 2022

Do more computation within a macro: Enrich computation.

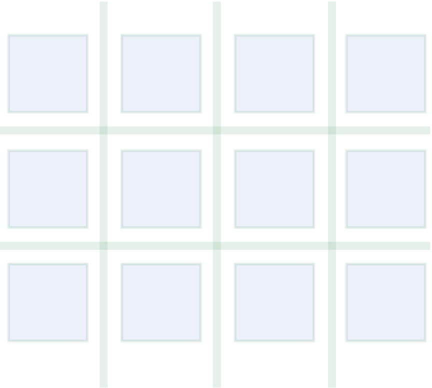
Also…IMC array size is limited due to parasitics.





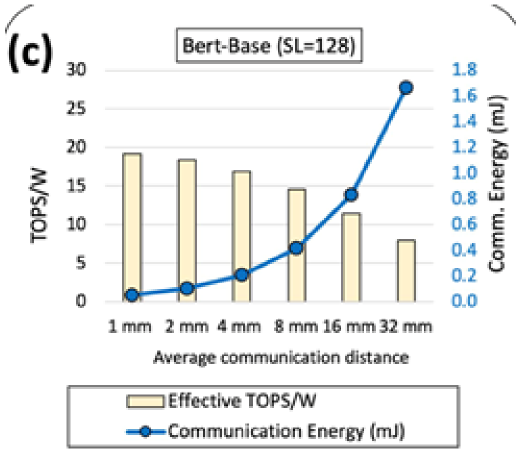
J. Chen, et al, IEEE TCAS II, 2023

Keep connectivity local.



IMC architectures.

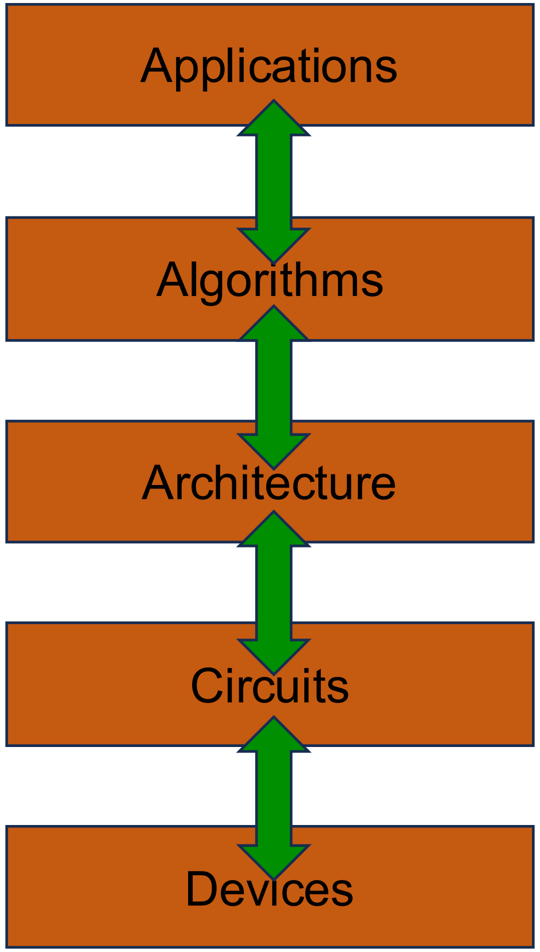
2) But… data moves again in routers…



S. Jain et al, IEEE TVLSI 2023

As it turns out, that’s also what the brains do.

Emerging Intelligent Substrates (EIS) lab

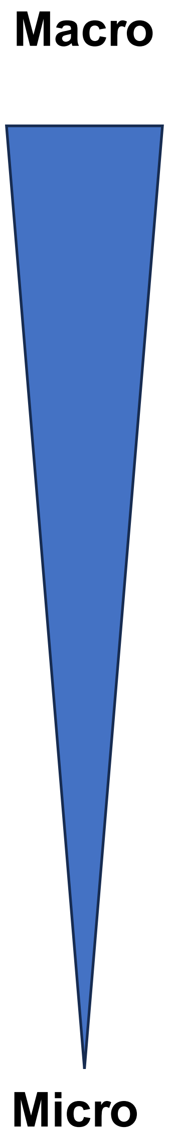

Applications

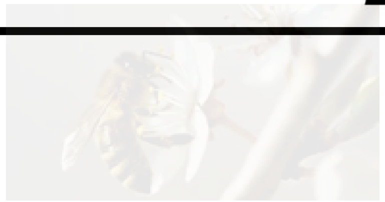
Algorithms

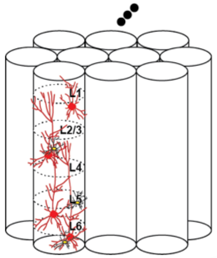
Architecture

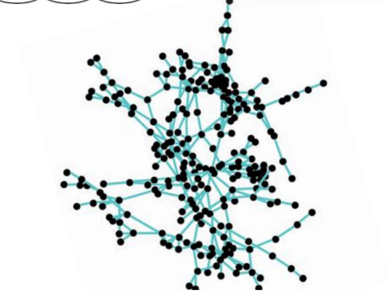
Circuits

Devices

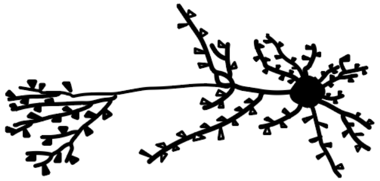




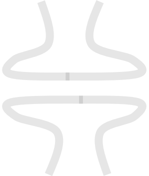




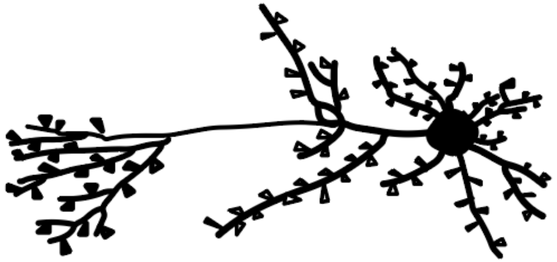
Keep connectivity local.

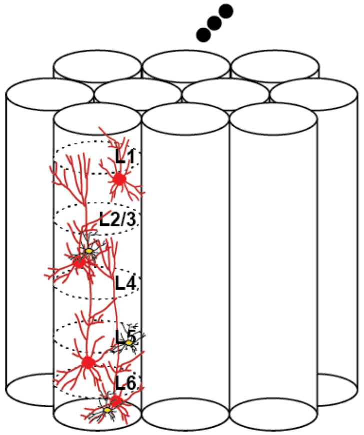


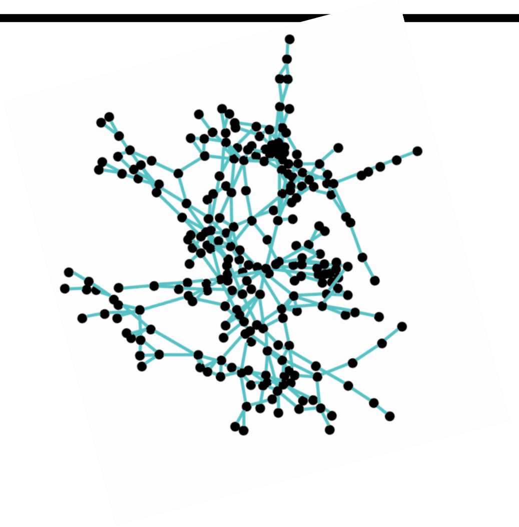
Enrich computation.



Emerging Intelligent Substrates (EIS) lab





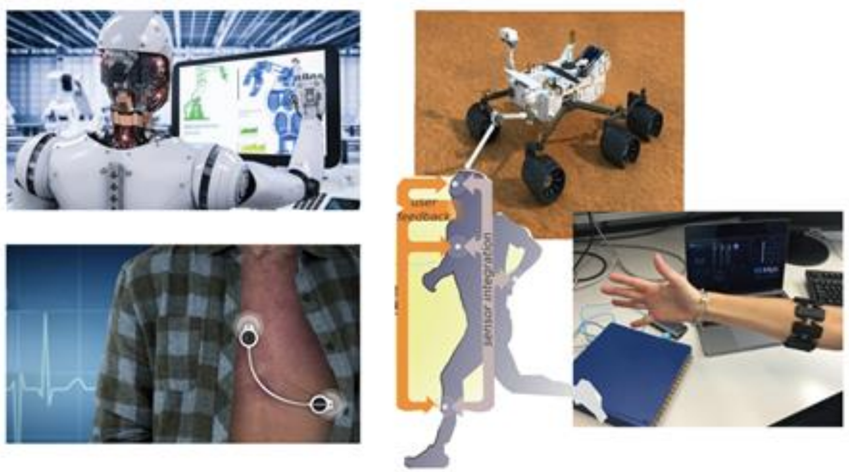


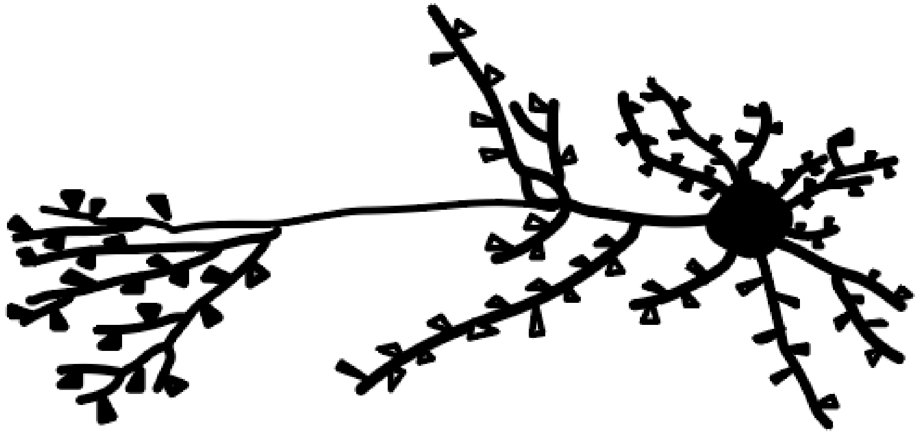
Part I:

Dendritic structures enrich computation locally.

Part II:

Local connectivity enables scalable efficient architectures.

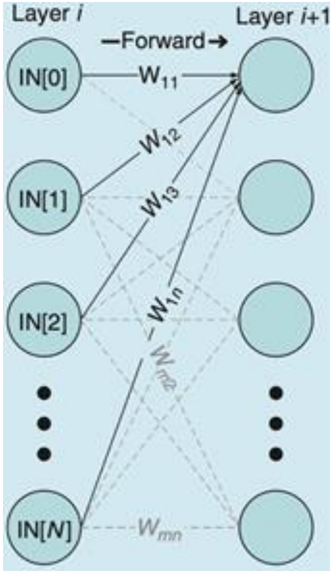




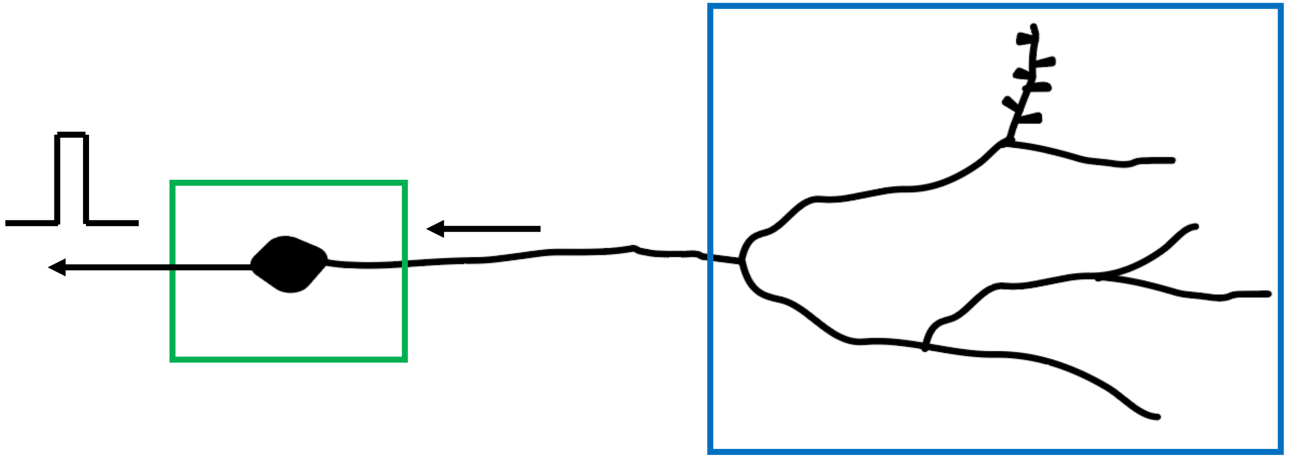
Part I:

Dendritic structures enrich computation locally.

What are dendrites and why should you care?



$$Y_{i+1} = f(IN_i\, W_{i,i+1})$$



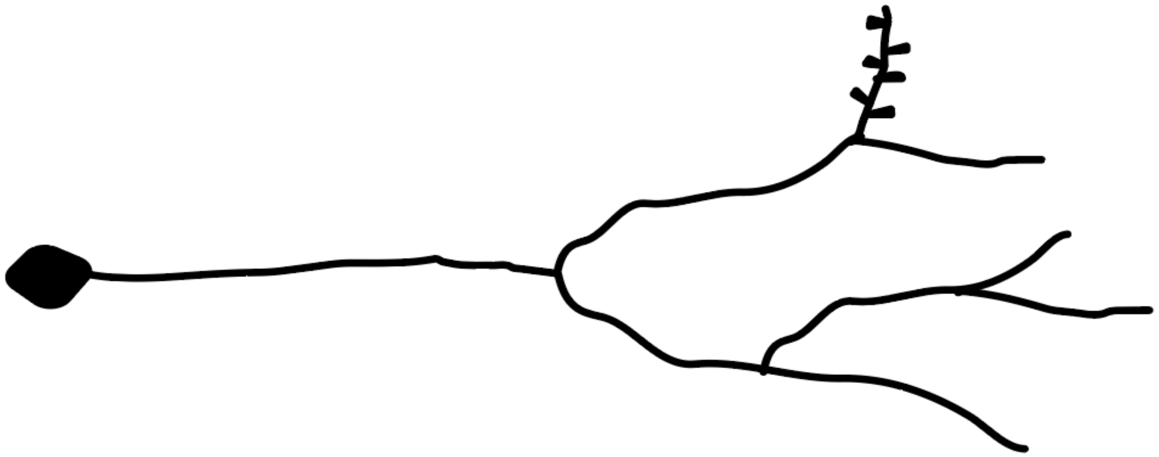
Soma:

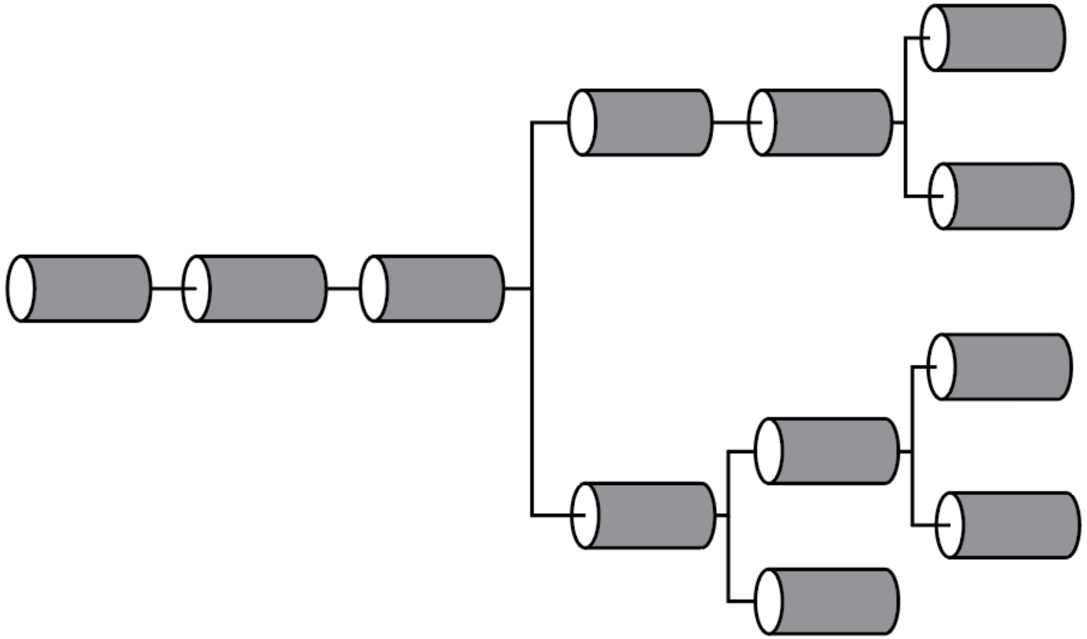
- (leaky) integrates the input
- applies thresholding function (f)
- generates the output

Dendrites:

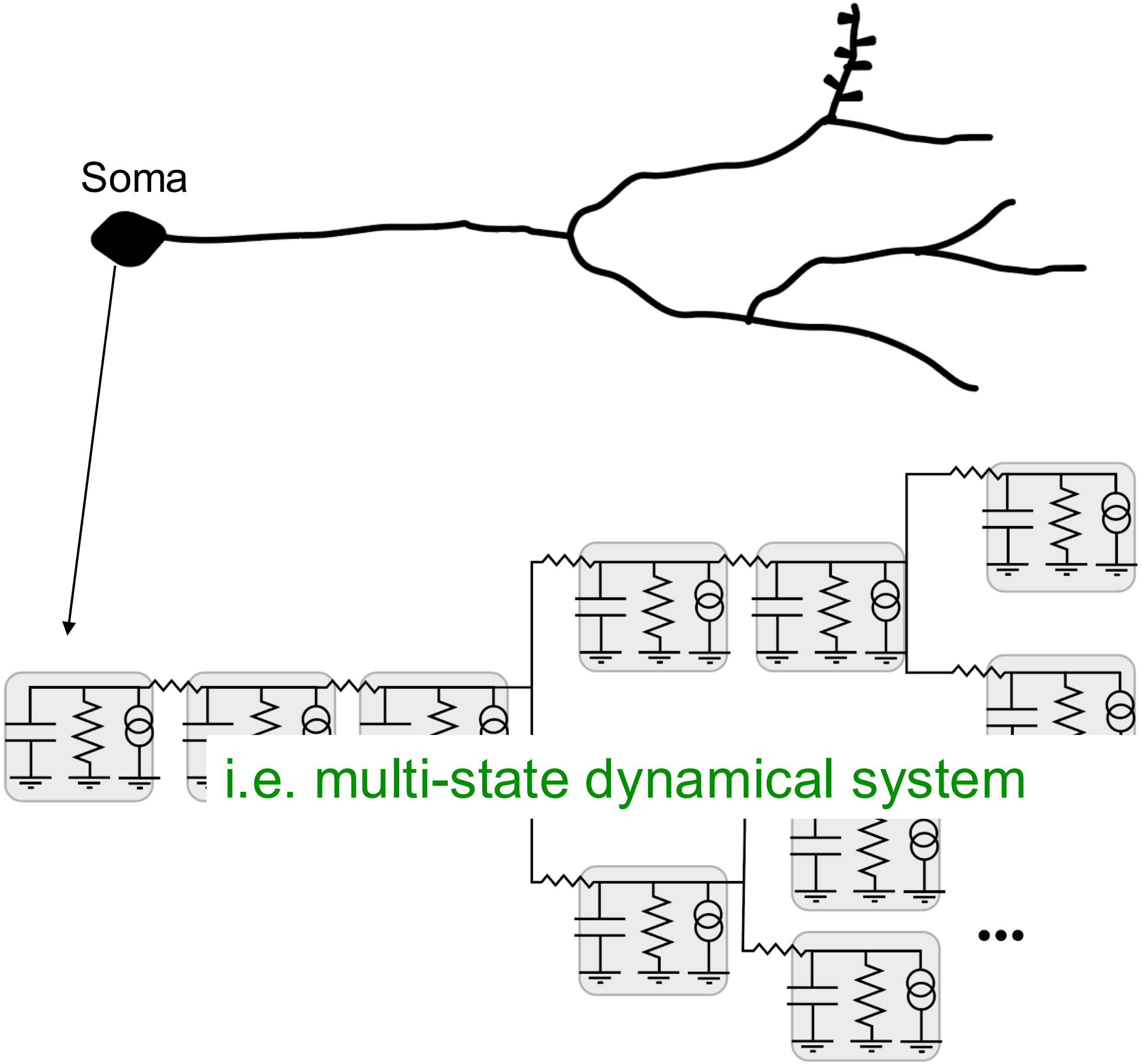
- Receive the input
- Spatio-temporal pre-processing

What are dendrites and why should you care?



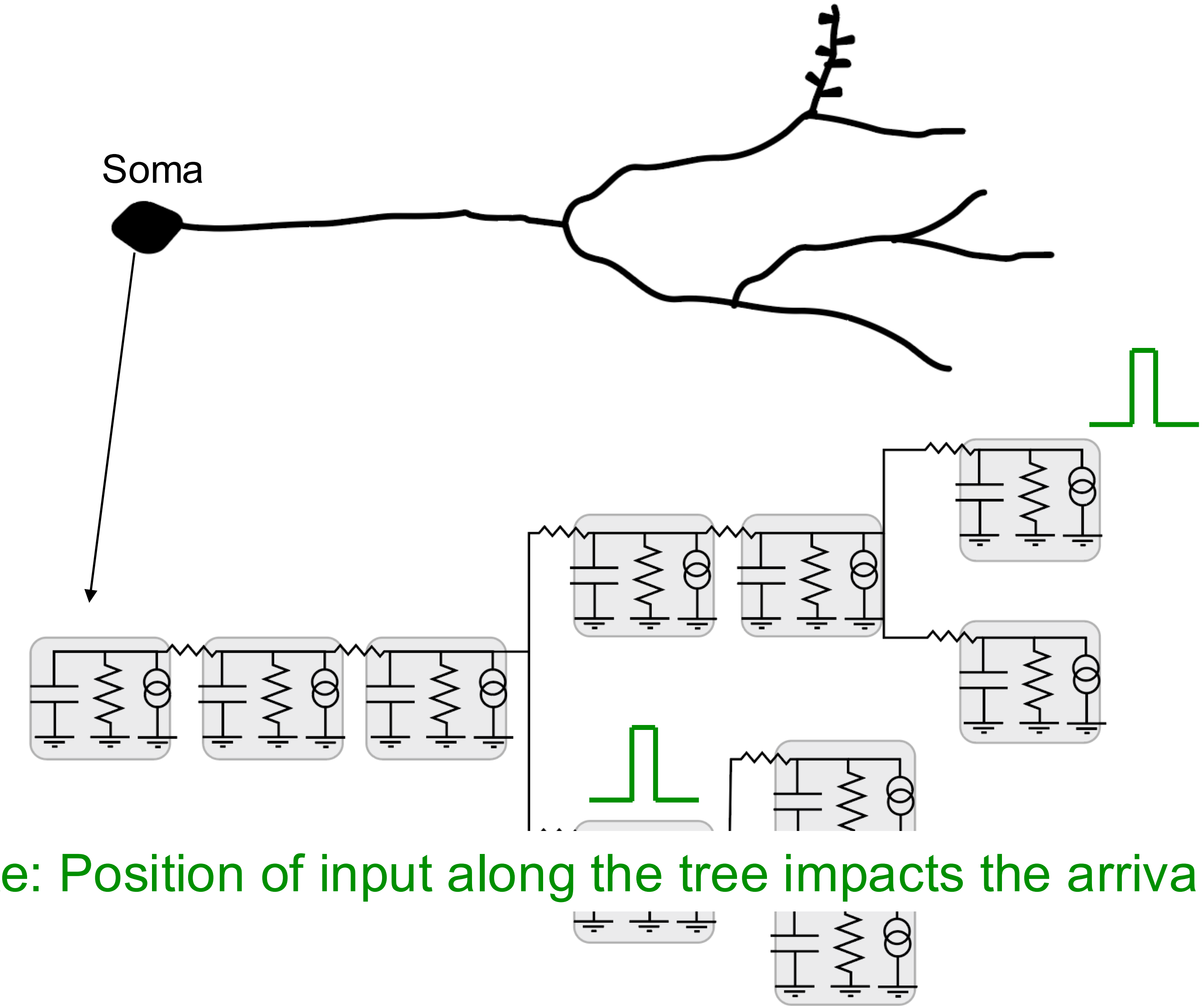


What are dendrites and why should you care?



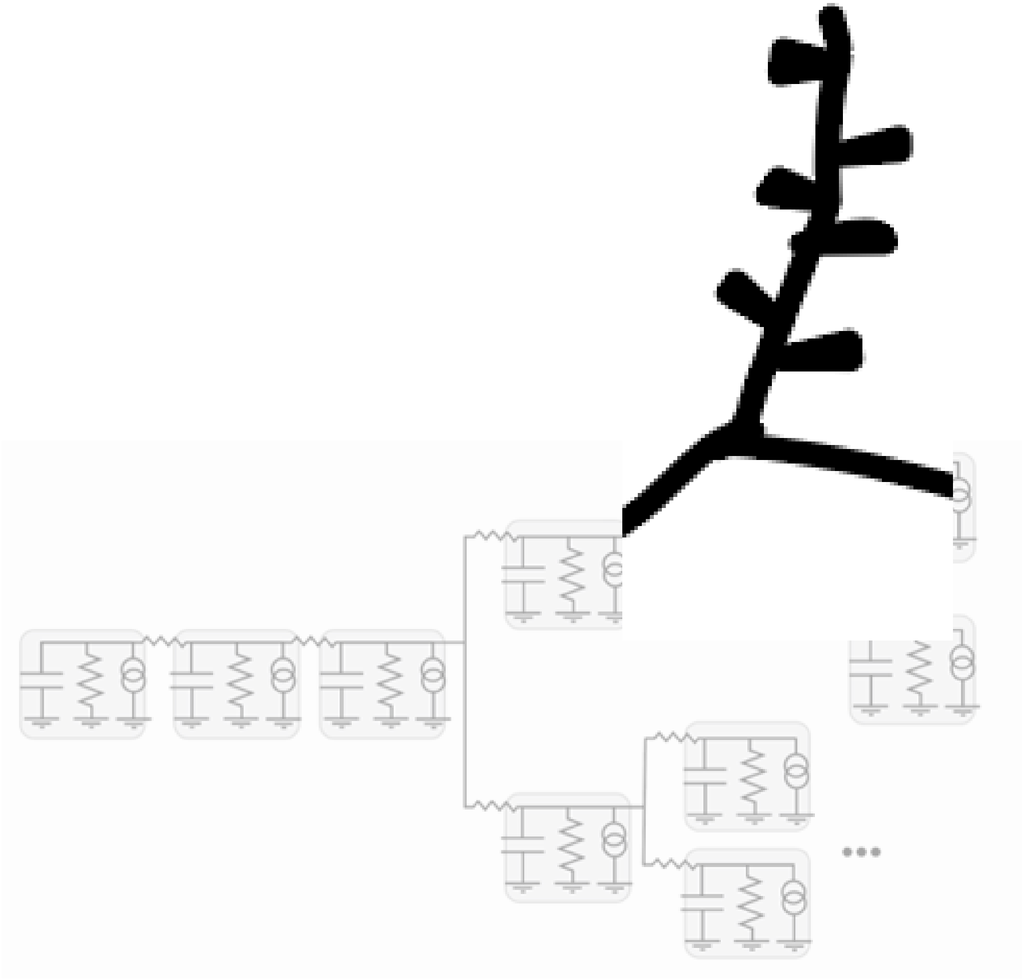
i.e. multi-state dynamical system

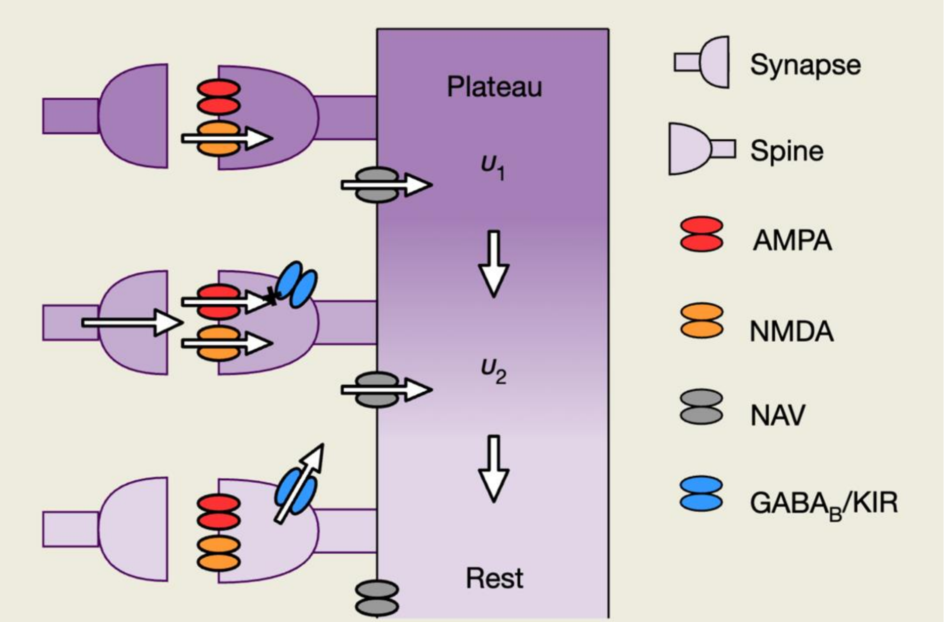
What are dendrites and why should you care?



Note: Position of input along the tree impacts the arrival time at the soma.

What are dendrites and why should you care?

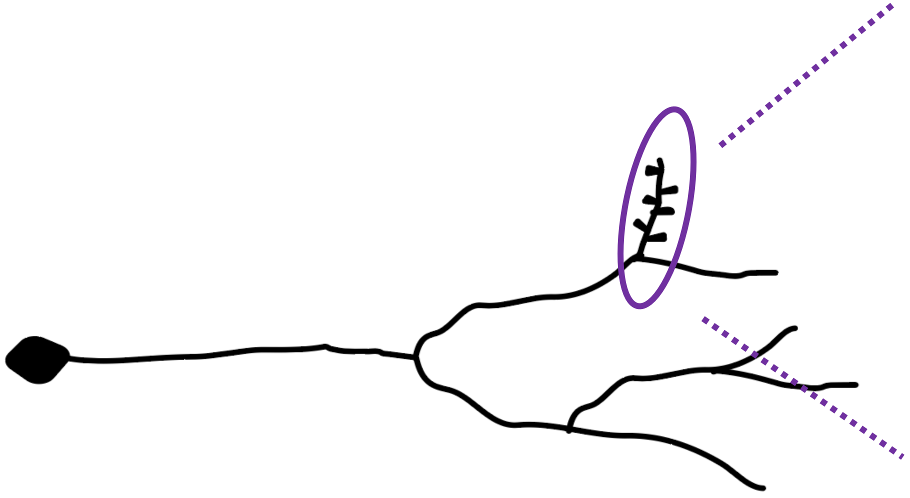


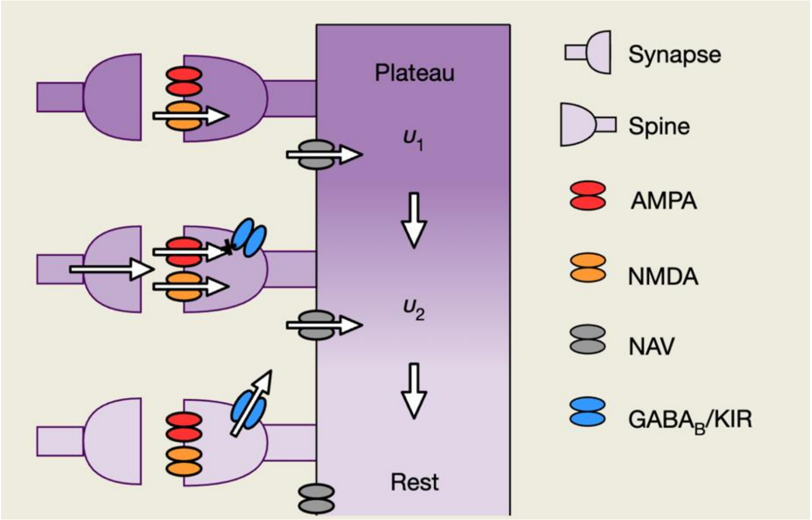


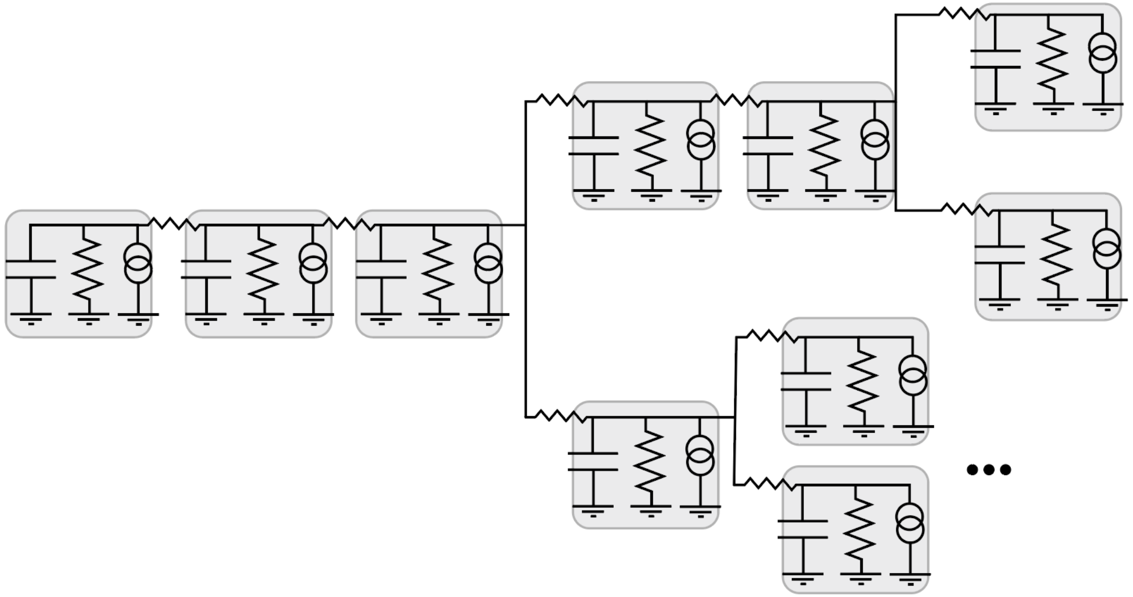
K. Boahen, Nature 2022

- Sequence detection:
  - Each dendritic spine acts as a gate for the next.
- Key operation:
  - multiplicative input-dependent gating

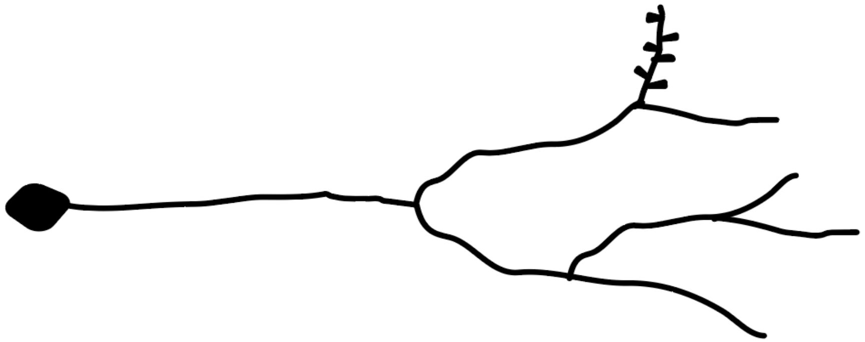
What are dendrites and why should you care?



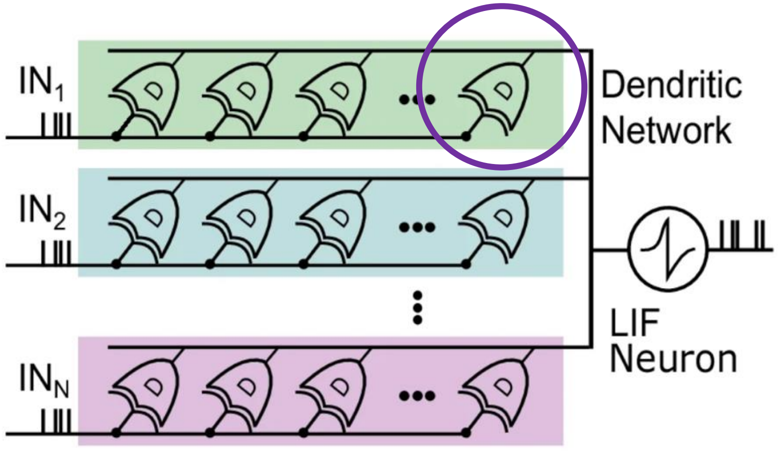


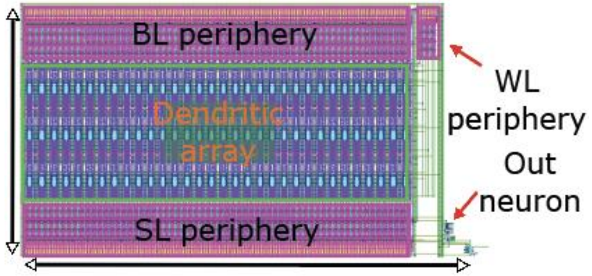


DenRAM: Modeling dendrites as passive cabels

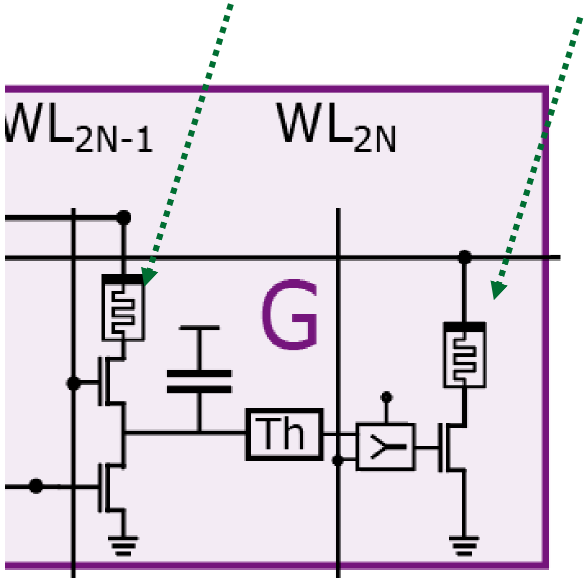


DenRAM





R_D: *delay d* G_W: *Weight W*



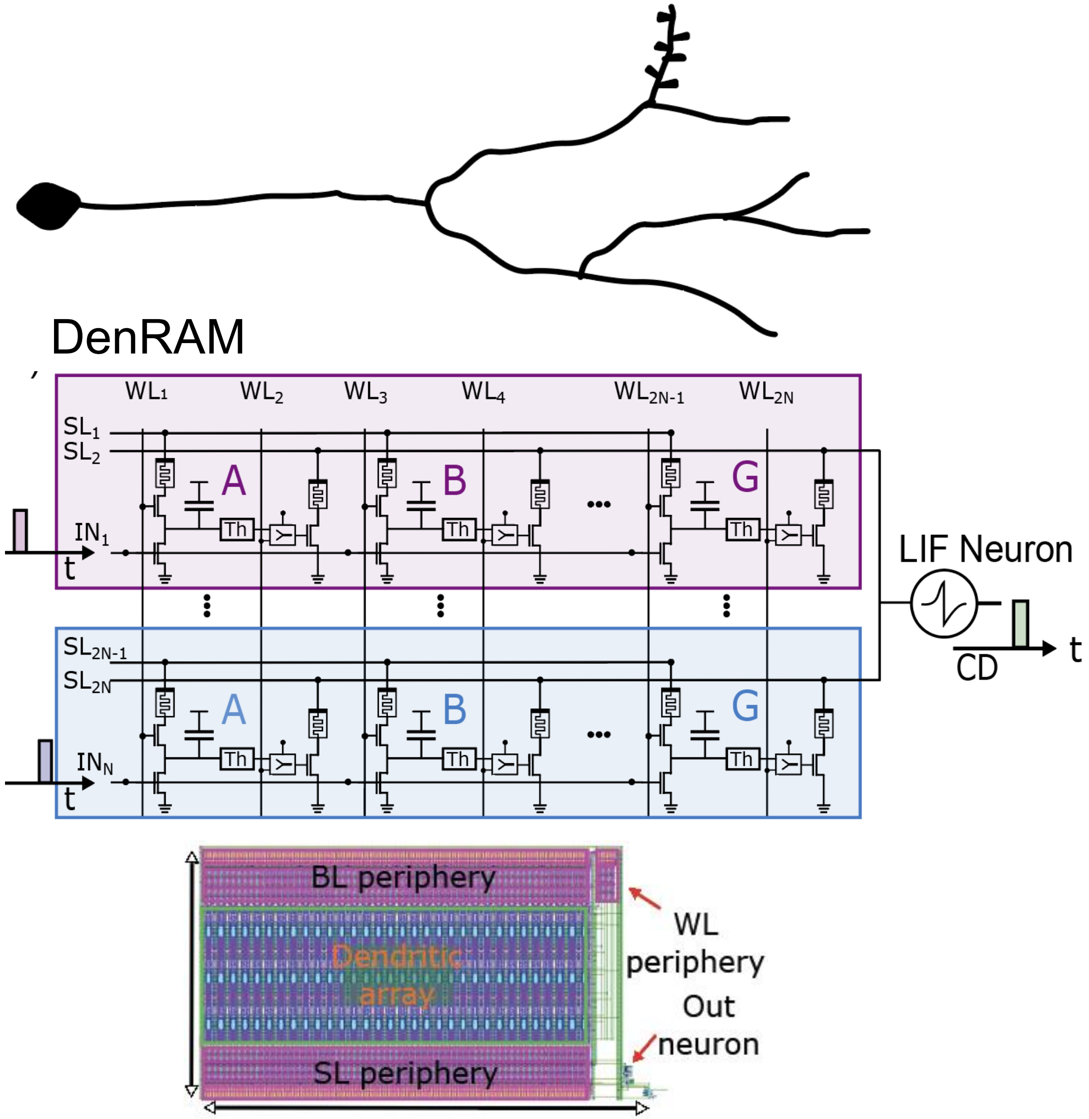
Synaptic kernel:

$$k_t = W\,\delta[t-d]$$

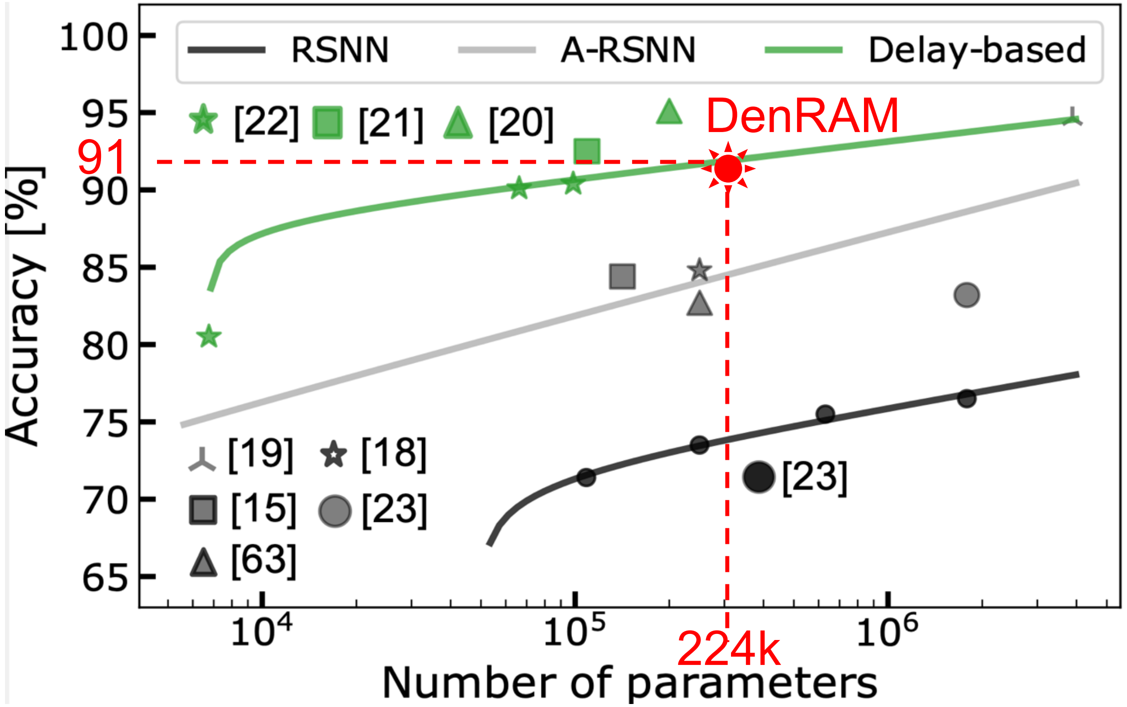
Filippo Moro

D'Agostino*, Moro*, Torchet*, et al, Nat. Comm. 2024

DenRAM: Delays improve acc. and param. efficiency



Task: Spoken digit dataset



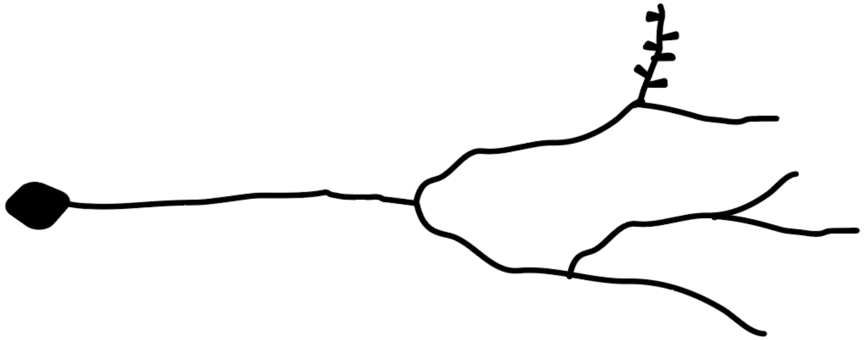
Tristan Torchet

Static delays and training weights.

[15] Haessig, G. et al. Front. Neurosci. 14, 420 (2020).
[18] Rao, A. et al. Nat. Mach. Intell. 4, 467–479 (2022).
[19] Nowotny, T. et al. Preprint (2022).
[20] Bittar, A. & Garner, P. N. Front. Neurosci. 16, 865897 (2022).
[21] Hammouamri, I. ICLR (2024).
[22] Sun, P. et al. ICASSP (2023).
[23] Patiño-Saucedo, A. et al. ISCAS (2023).
[63] Shrestha, S. B. & Orchard, G. NeurIPS (2018).

D'Agostino*, Moro*, Torchet*, et al, Nat. Comm. 2024

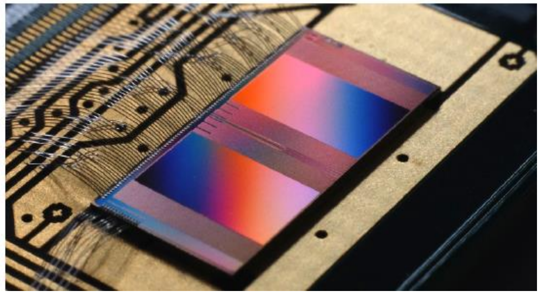
DelGrad: Delays improve acc. and param. efficiency

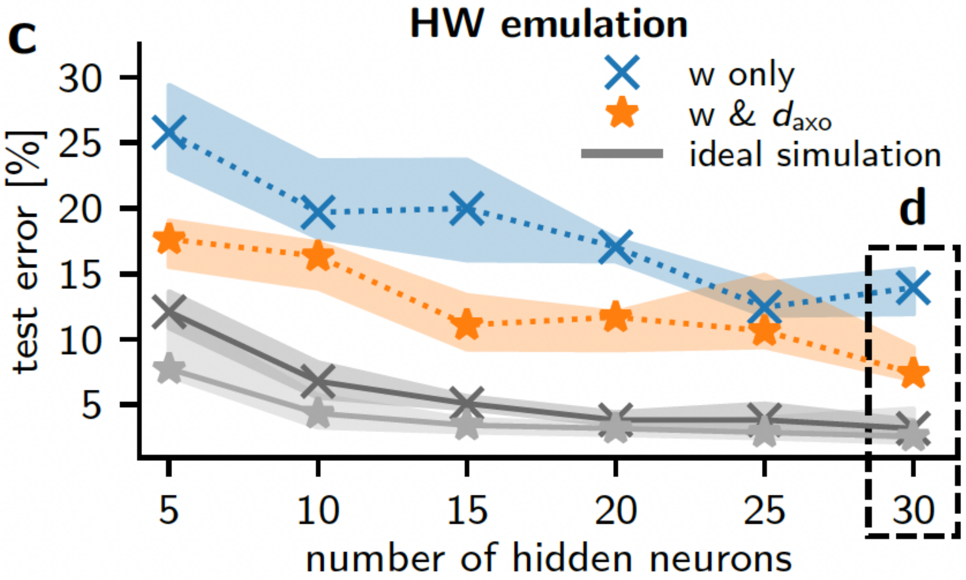


Laura Kriener

Jimmy Weber

On BrainScaleS-2 Neuromorphic chip





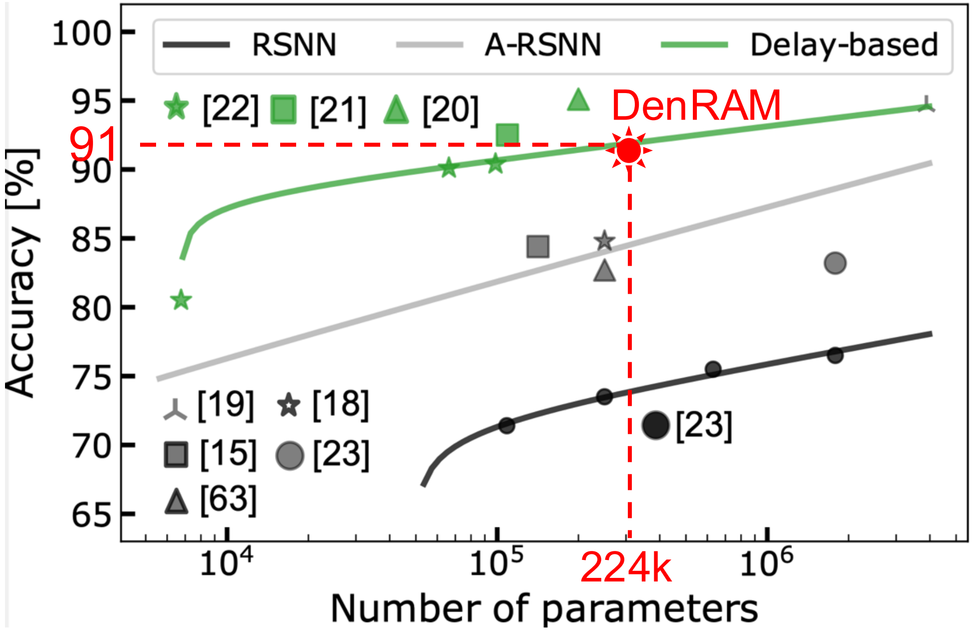
Learning delays and weights using chip-in-the-loop training

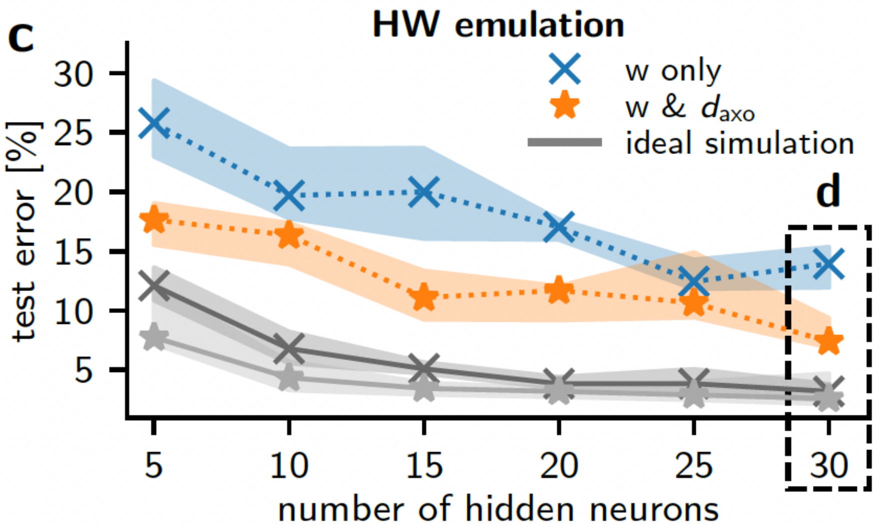
Pehle*, Billaudelle*, et al, Frontiers in Neuroscience 2022

Goeltz*, Weber*, Kriener*, et al, Nat. Comm. 2025

Summary: (Learning) delays increase accuracy and reduce size!

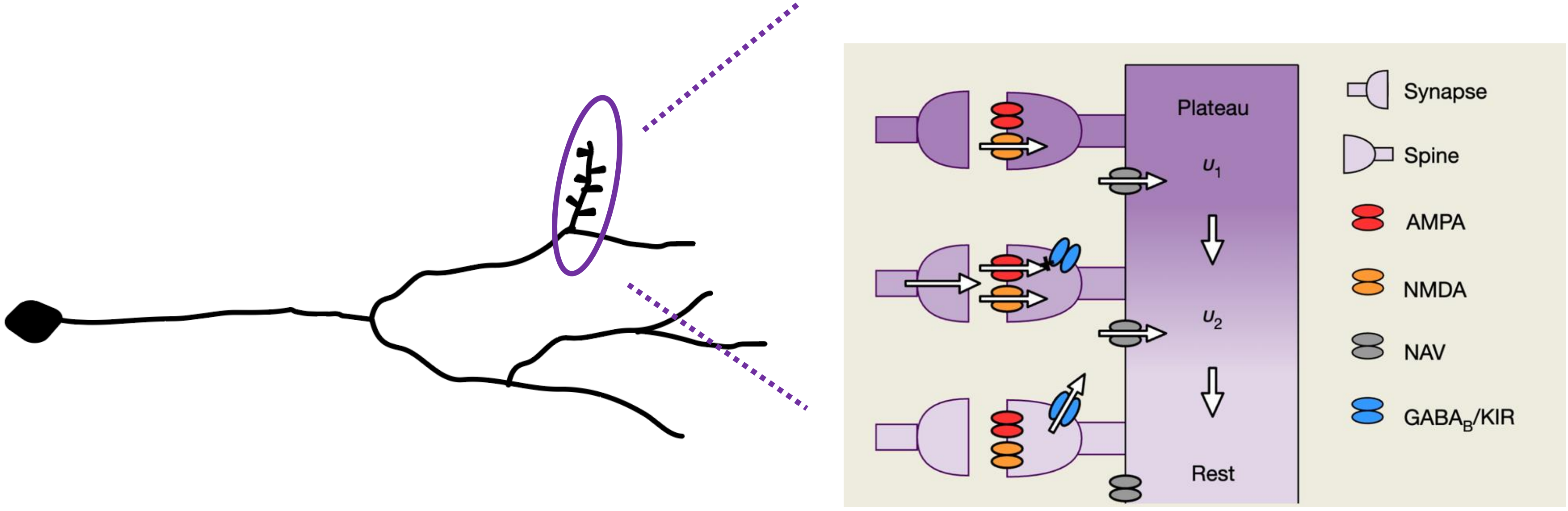
$$k_t = W\ \delta[t - d]$$

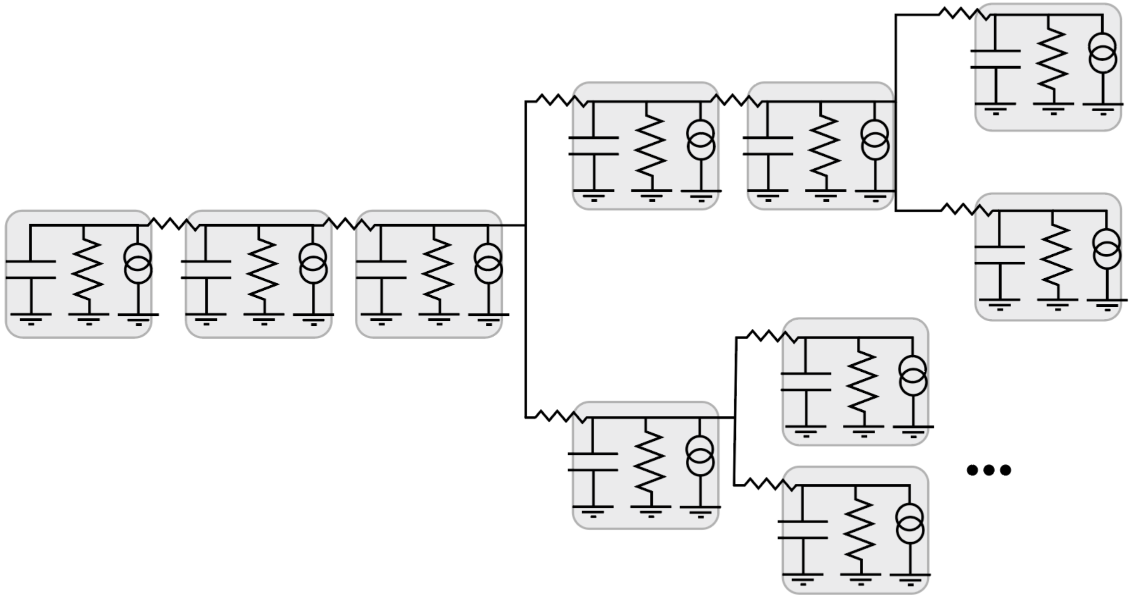




Torchet*, Metzner*, et al, Arxiv 2025

What are dendrites and why should you care?





- Key operation:
  - multiplicative input-dependent gating

Gated recurrent networks

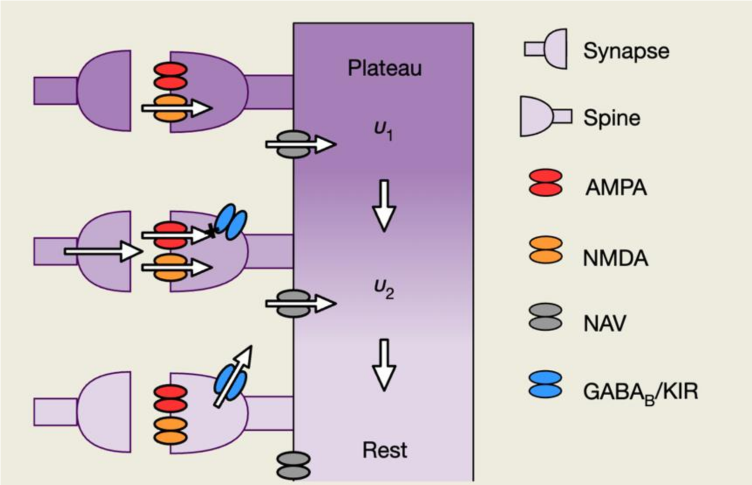
$$h_t = (1 - z_t) \odot h_{t-1} + z_t \odot \tilde{h}_t$$

$$z_t = \sigma(W_z x_t)\,, \quad \tilde{h}_t = W_h x_t$$

“Were RNNs all we needed?”, Feng, et al, 2024

mGRADE: minimal Recurrent Gate + Delays

Model:
mGRADE: Gated recurrent + delays



$$h_t = (1 - z_t) \odot h_{t-1} + z_t \odot \tilde{h}_t$$

$$z_t = \sigma(W_z x_t)\,, \quad \tilde{h}_t = W_h x_t$$

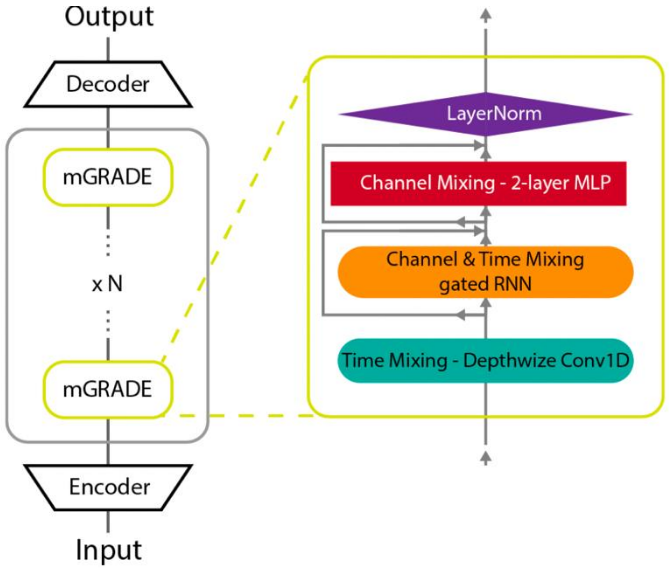
$$x_t = (k * u)[t]$$

Task: Long Range Arena (LRA), GSC

LRA:
Suit of multi-time scale sequence tasks

Task 1: “ListOps”: result of a math seq.
Task 2: “Text”: Binary sentiment analysis of text
Task 3: “Retrieval”: Citation link bet. 2 papers
Task 4: “Image”: Pixel-by-pixel image class.
Task 5: “Pathfinder”: Finding path in an image

Network



Output
Decoder
mGRADE
x N
mGRADE
Encoder
Input
LayerNorm
Channel Mixing - 2-layer MLP
Channel & Time Mixing gated RNN
Time Mixing - Depthwize Conv1D

Tristan Torchet

Christian Metzner

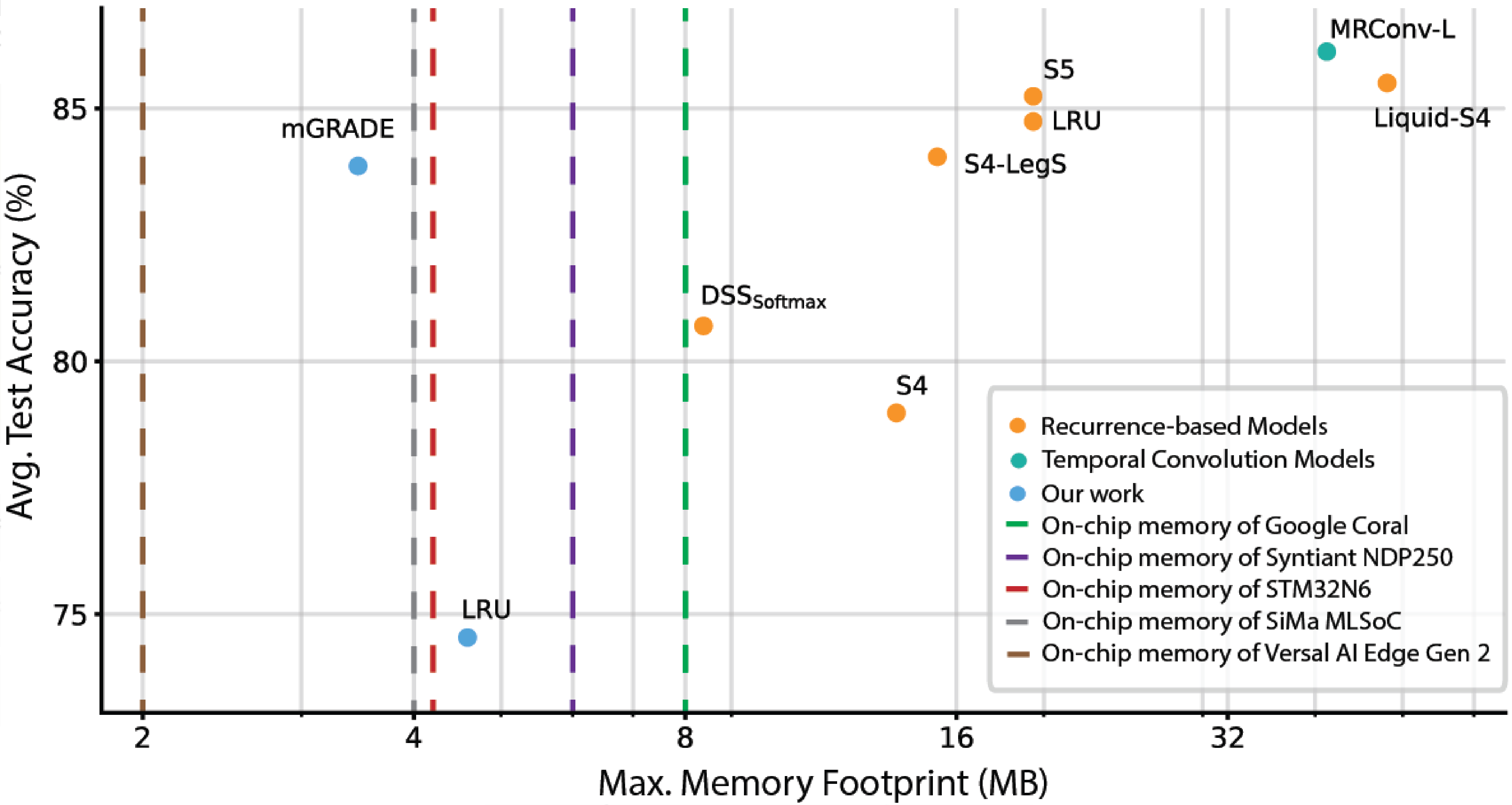
Results:

Torchet*, Metzner*, et al, Arxiv 2025

mGRADE: Results

University of Zurich^UZH

ETH zürich

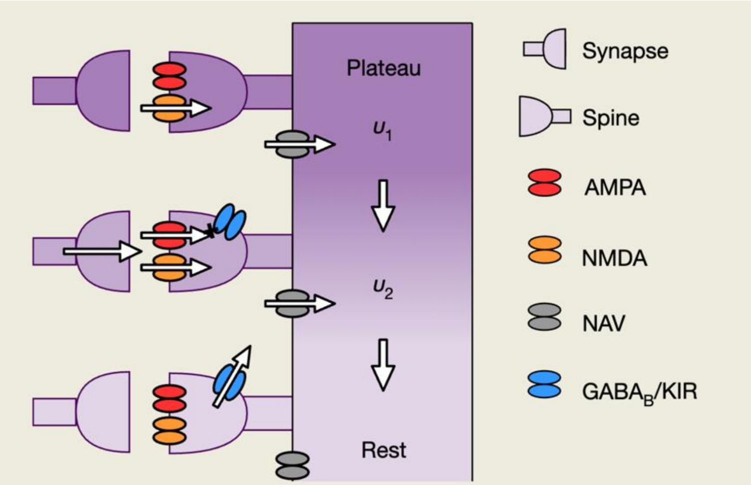


Christian Metzner

Torchet*, Metzner*, et al, Arxiv 2025

mGRADE: minimal Recurrent Gate + Delays

Model:
mGRADE: Gated recurrent + delays



$$h_t = (1 - z_t) \odot h_{t-1} + z_t \odot \tilde{h}_t$$

$$z_t = \sigma(W_z x_t)\,, \quad \tilde{h}_t = W_h x_t$$

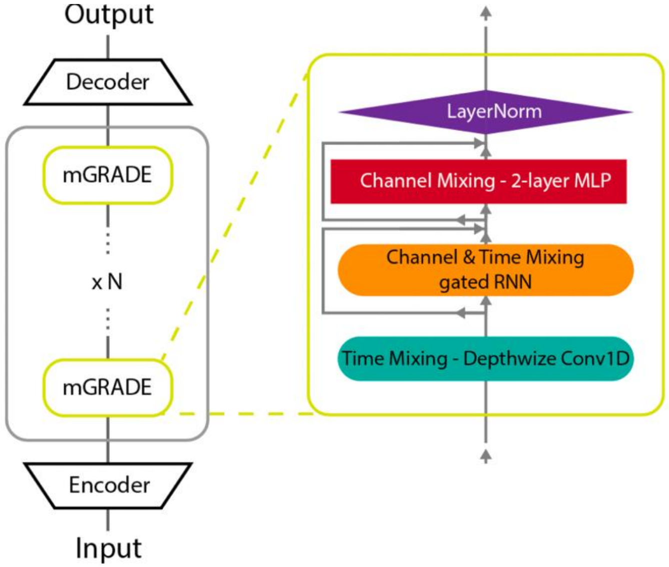
$$x_t = (k * u)[t]$$

Task: Long Range Arena (LRA), GSC

LRA:
Suit of multi-time scale sequence tasks

Task 1: “ListOps”: result of a math seq.
Task 2: “Text”: Binary sentiment analysis of text
Task 3: “Retrieval”: Citation link bet. 2 papers
Task 4: “Image”: Pixel-by-pixel image class.
Task 5: “Pathfinder”: Finding path in an image

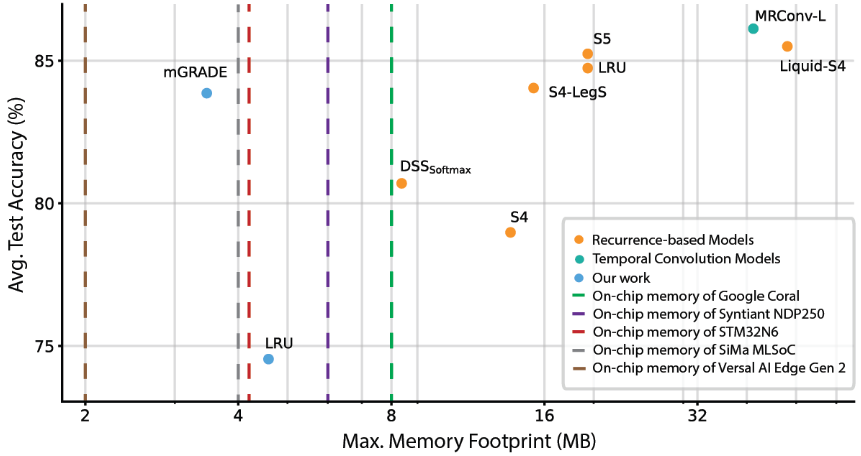
Network


Output
Decoder
mGRADE
x N
mGRADE
Encoder
Input
LayerNorm
Channel Mixing - 2-layer MLP
Channel & Time Mixing gated RNN
Time Mixing - Depthwize Conv1D

Tristan Torchet

Christian Metzner

Results:

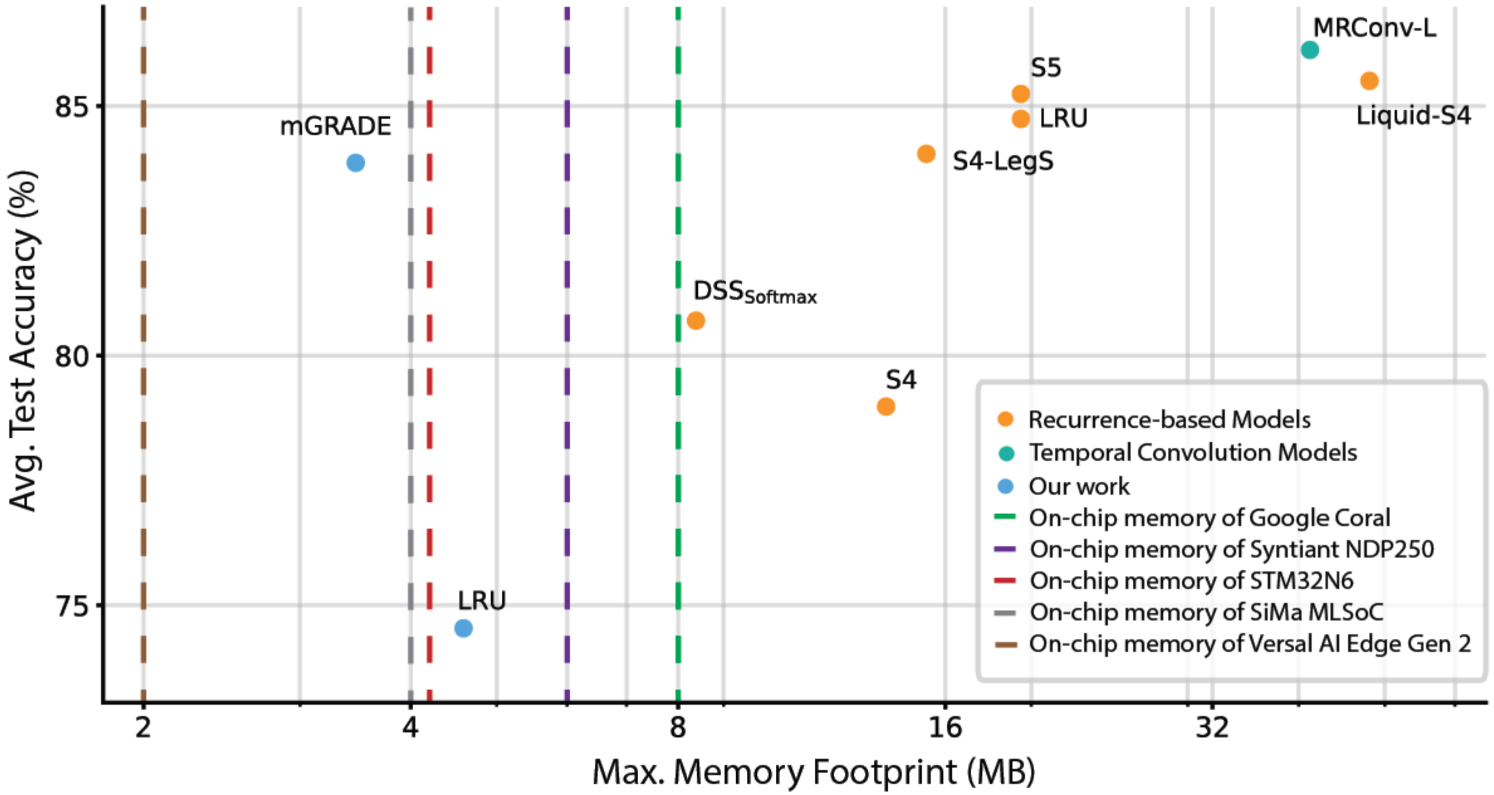


Delay convolutions model short-term dynamics
gating model long-term dynamics:
good inductive bias!

Torchet*, Metzner*, et al, Arxiv 2025

Summary: Input dependent gating + delays = light-weight multi-time scale

$$x_t = (k * u)[t] \qquad z_t = \sigma(W_z x_t)\,, \quad \tilde{h}_t = W_h x_t \qquad h_t = (1 - z_t) \odot h_{t-1} + z_t \odot \tilde{h}_t$$



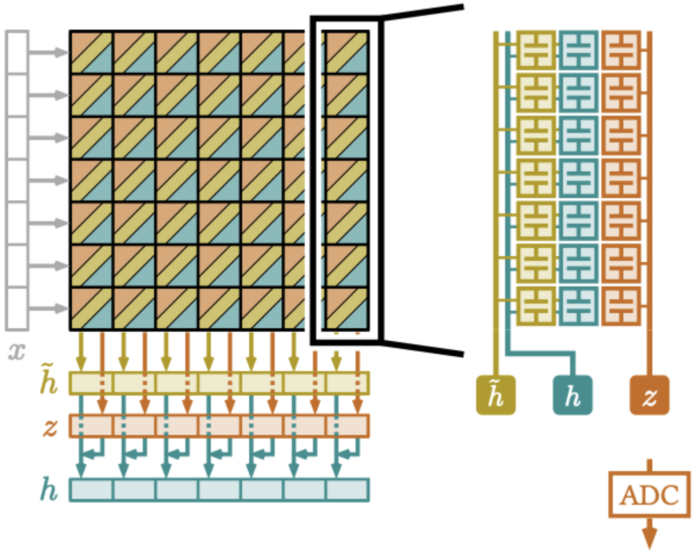
How to build this in HW?

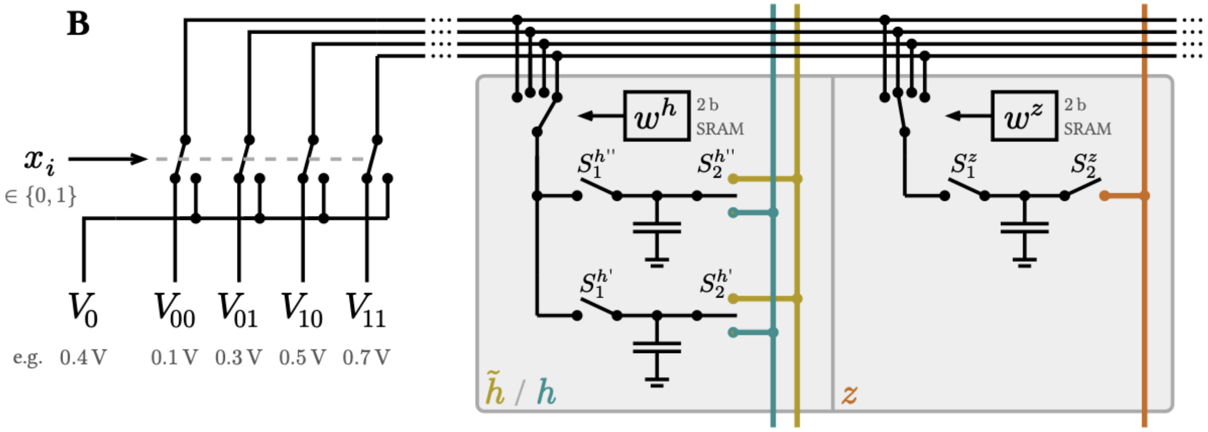
Torchet*, Metzner*, et al, Arxiv 2025

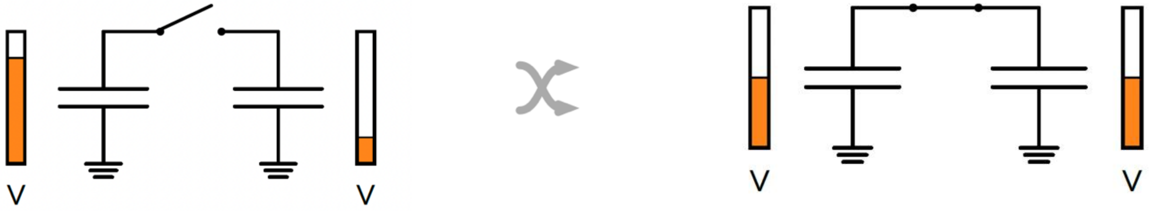
MINIMALIST: SC circuits for IMC of gated recurrent units

$$h_t = (1 - z_t) \odot h_{t-1} + z_t \odot \tilde{h}_t \qquad z_t = \sigma(W_z x_t)\,, \quad \tilde{h}_t = W_h x_t$$

Sebastian Billaudelle



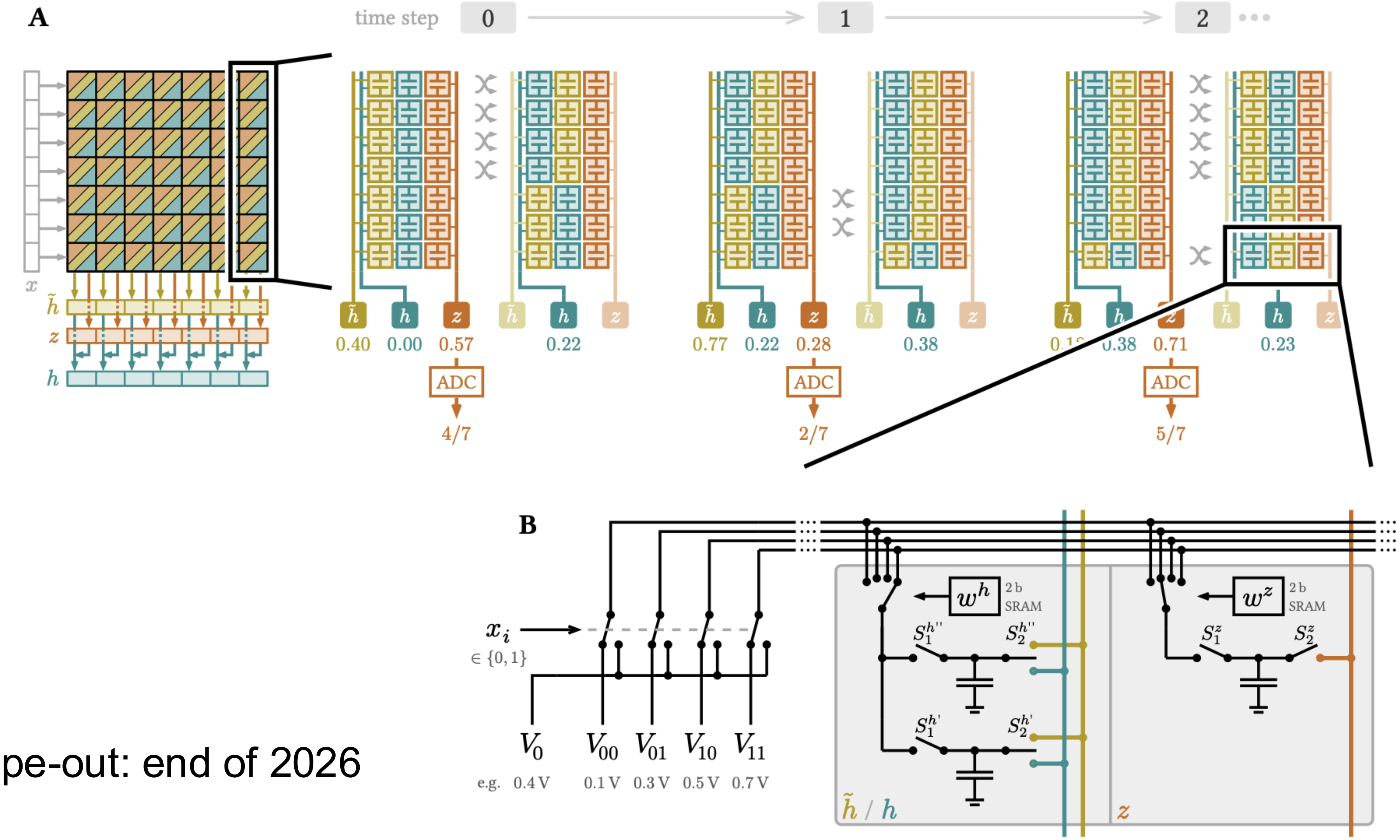




Closing the switch calculates the weighted sum: mix and stir

Billaudelle*, Kriener*, et al, ArXiv 2025

MINIMALIST: SC circuits for IMC of gated recurrent units



Sebastian Billaudelle

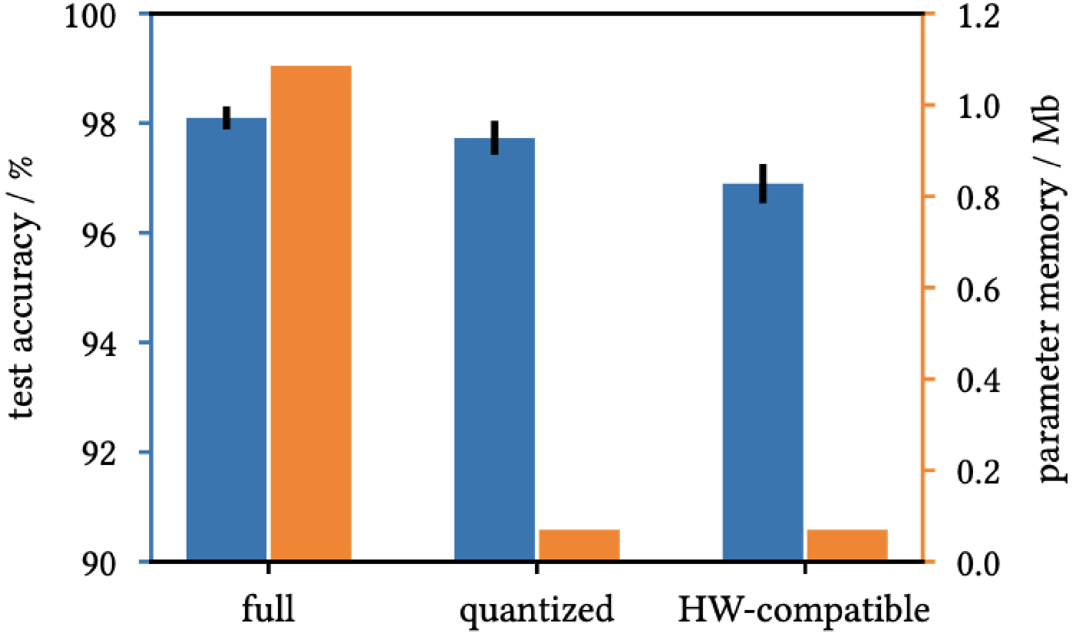
Target tape-out: end of 2026

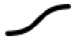
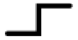
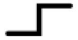
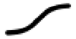
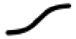
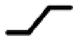
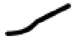
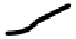
$$z_t = \sigma(W_z x_t)\,, \quad \tilde{h}_t = W_h x_t \qquad h_t = (1 - z_t) \odot h_{t-1} + z_t \odot \tilde{h}_t$$

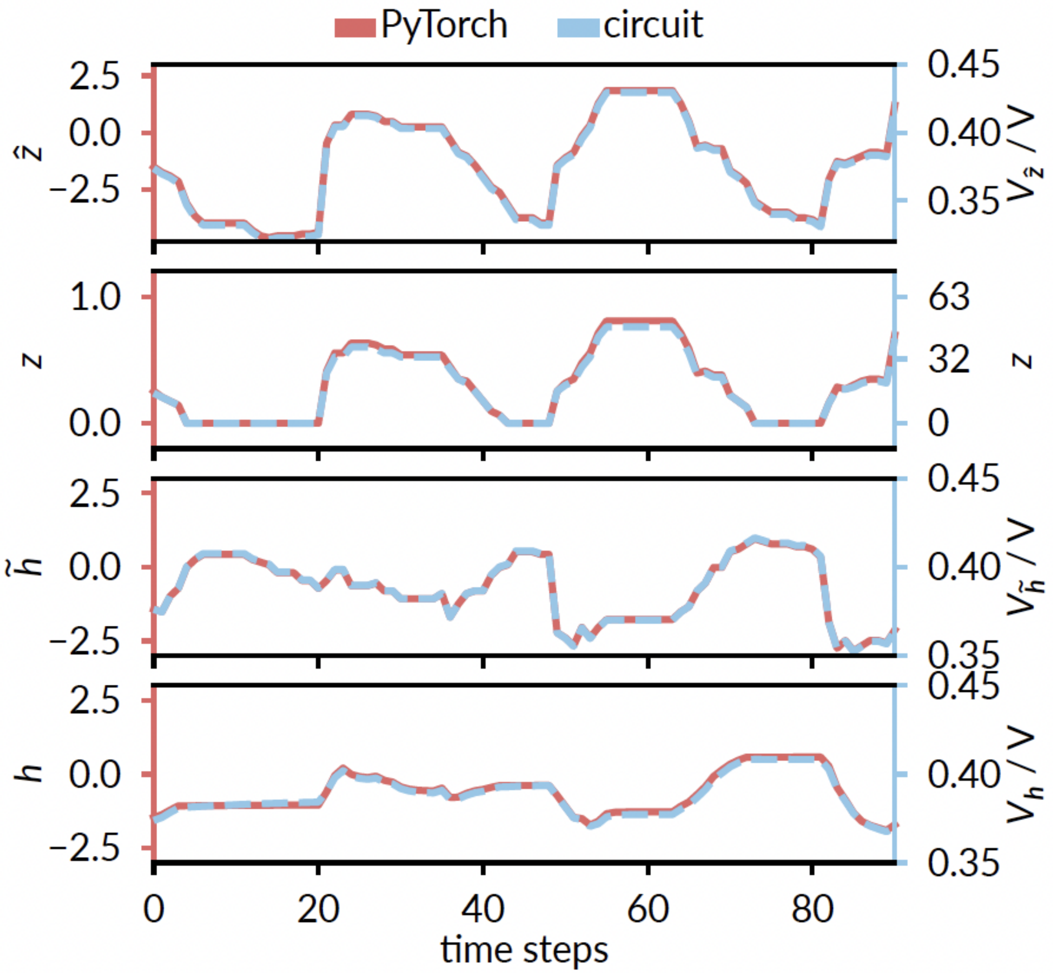
Billaudelle*, Kriener*, et al, ArXiv 2025

Hardware aware training of MINIMALIST

SMNIST Benchmark



W	f32	i2	i2
b	f32	i6	i6
z	f32	f32	i6
σ^h			
σ^z			
$\sigma^{\tilde{h}}$			$\mathbb{1}$

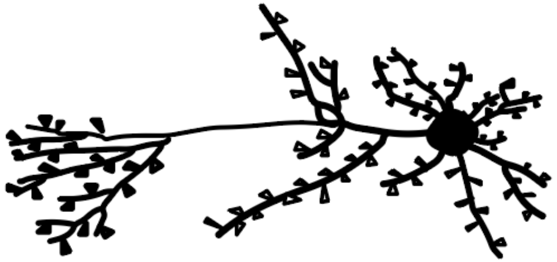


Laura
Kriener

$$z_t = \sigma(W_z x_t)\,, \quad \tilde{h}_t = W_h x_t \qquad h_t = (1 - z_t) \odot h_{t-1} + z_t \odot \tilde{h}_t$$

Billaudelle*, Kriener*, et al, ArXiv 2025

Part I: Summary

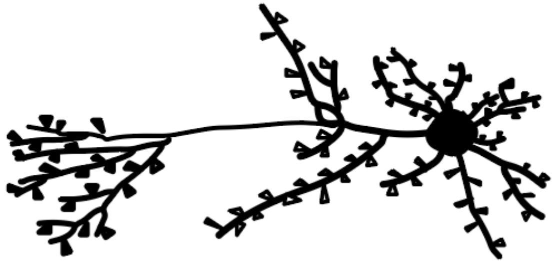


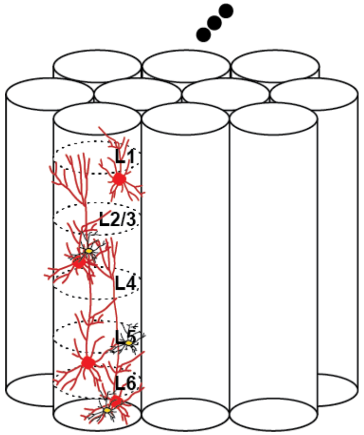
Dendritic structures enrich computation locally.

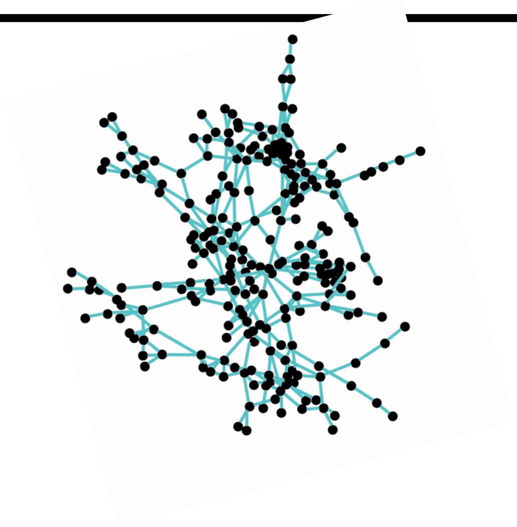
Enriching neural network architectures with dendritic kernels brings memory footprint and accuracy advantage!!

Co-design has been key in achieving this!

Emerging Intelligent Substrates (EIS) lab





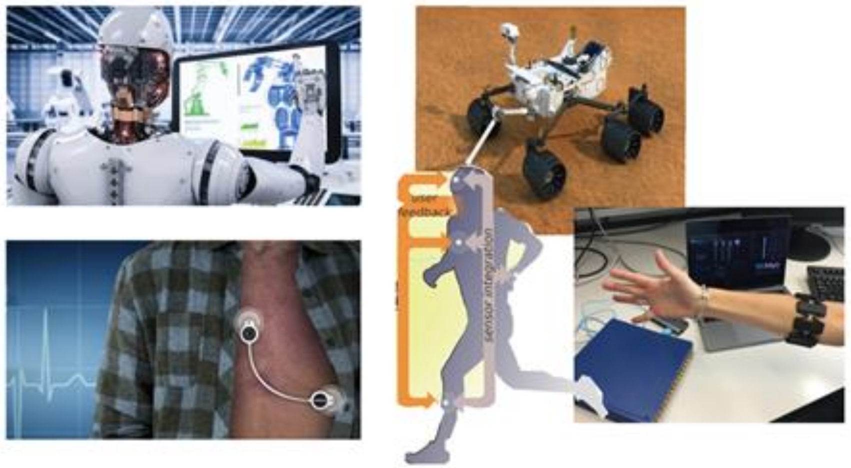


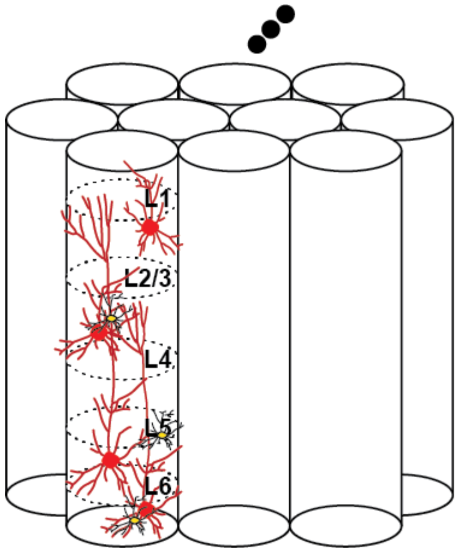
Part I:

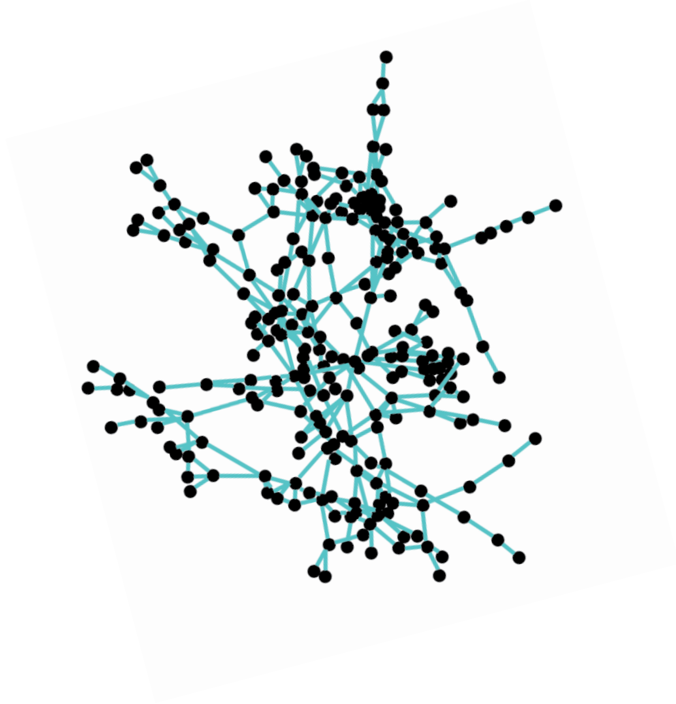
Dendritic structures enrich computation locally.

Part II:

Local connectivity enables scalable efficient architectures.



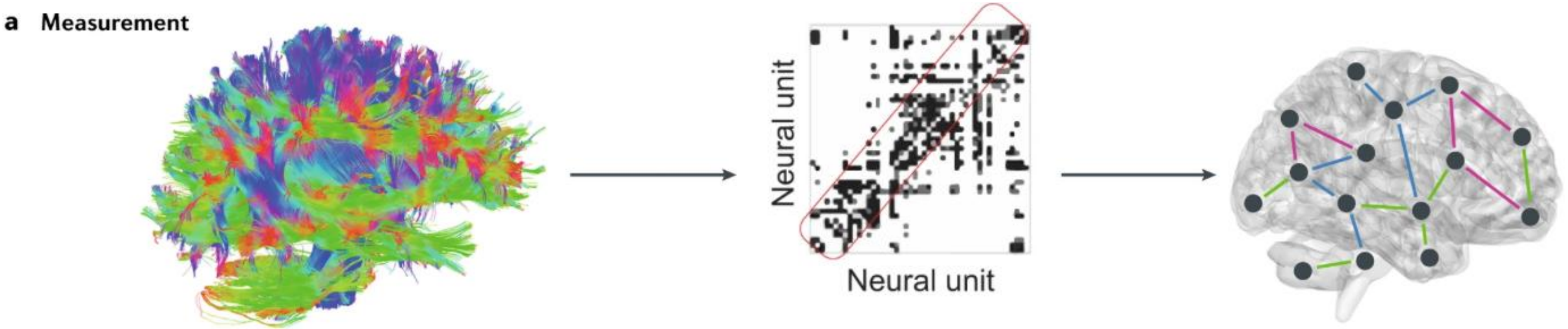




Part II:

Local connectivity enables scalable efficient architectures.

Brain connectivity shows a small-world topology.



Example: white matter tracts (via diffusion tensor imaging)

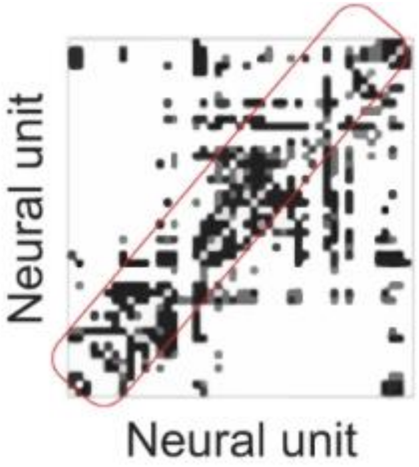
Adjacency matrix

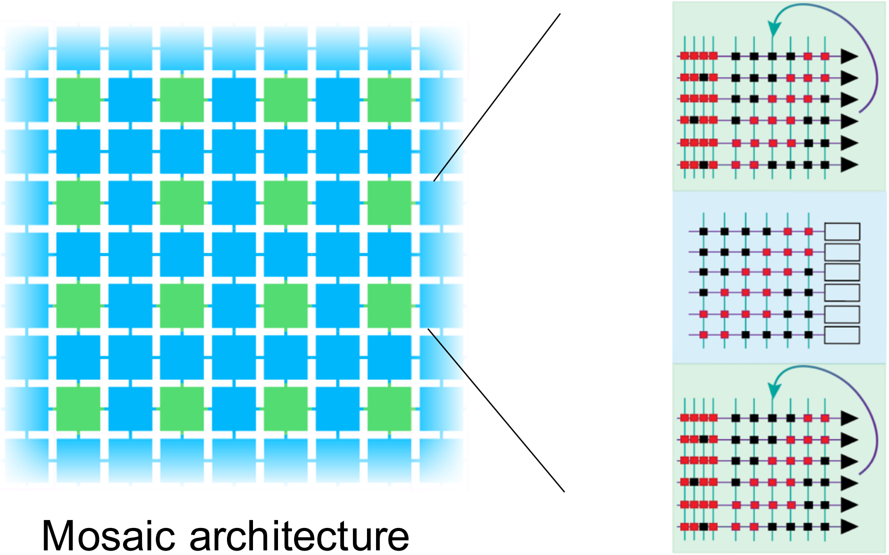
Structural brain network

- Small-world topology:
  - Strong local connectivity and sparse global connectivity
  - Short path length and high clustering: Information spreads very fast in the network.
  - Connectivity matrix: sparse and only dense around diagonal connections

C.W. Lynn and D. Basset, Nat. Rev. Phys. 2019

Mosaic: small-world connectivity in multi-core IMC chips.





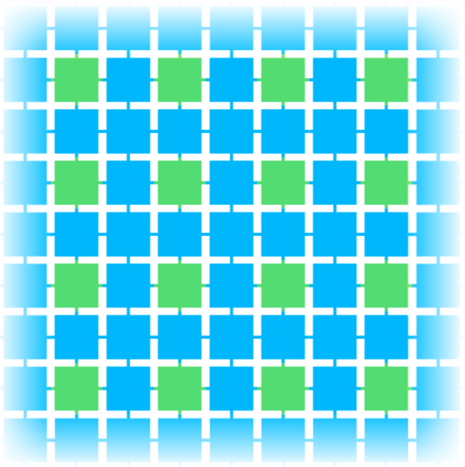
Mosaic architecture

- Mosaic follows similar connectivity profile to the brain’s connectivity profile.
- In-memory compute core used as high-clustering nodes.
- In-memory router nodes used for sparse global connectivity.

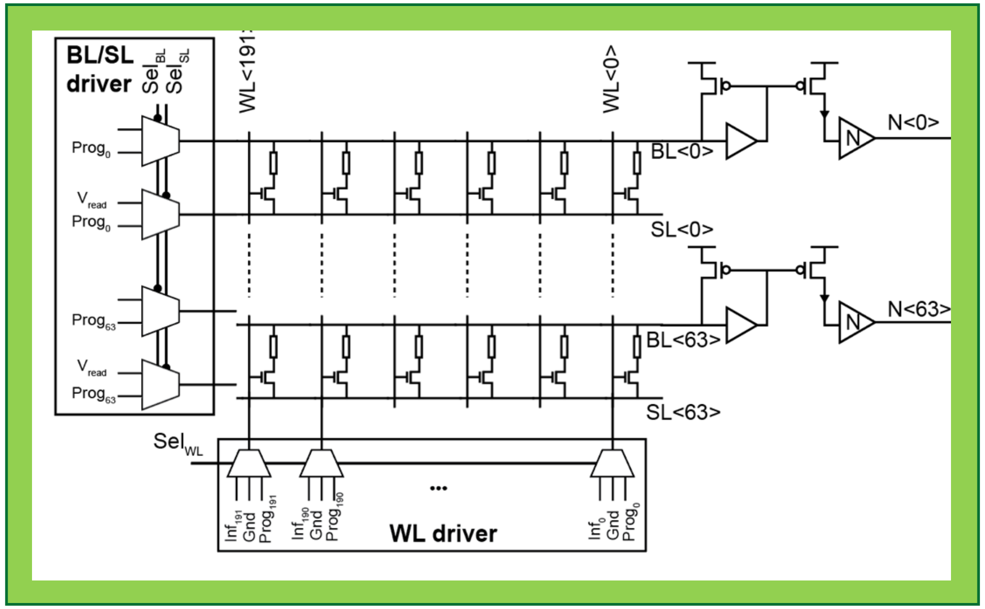
RRAM is used as the memory for both compute and route.

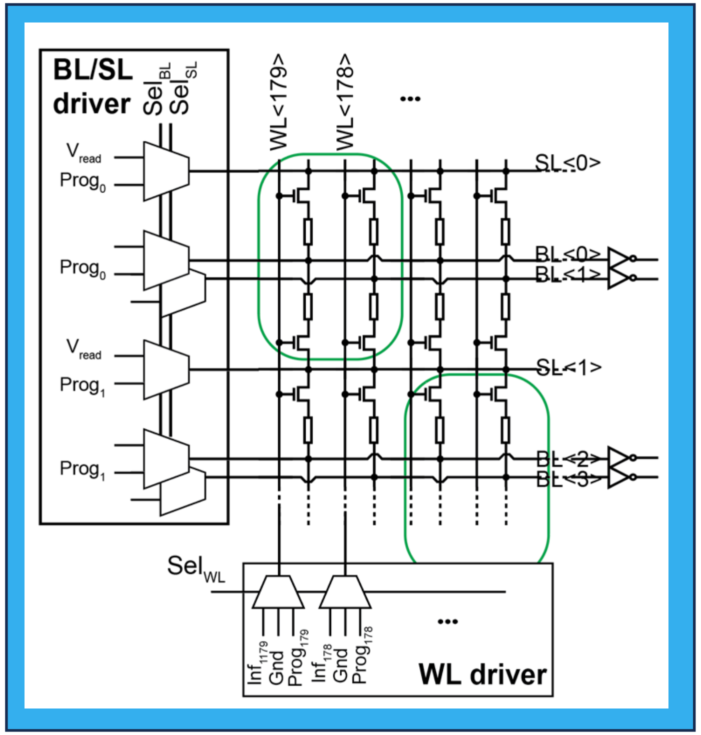
Dalgaty*, Moro*, Demirag*, et al, Nat. Comm. 2024

Mosaic: Local connectivity reduces the communication distance.



Mosaic architecture

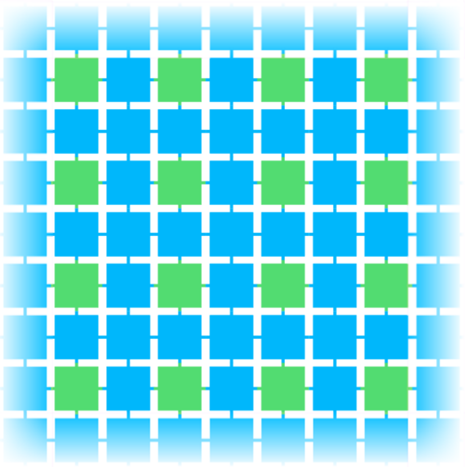




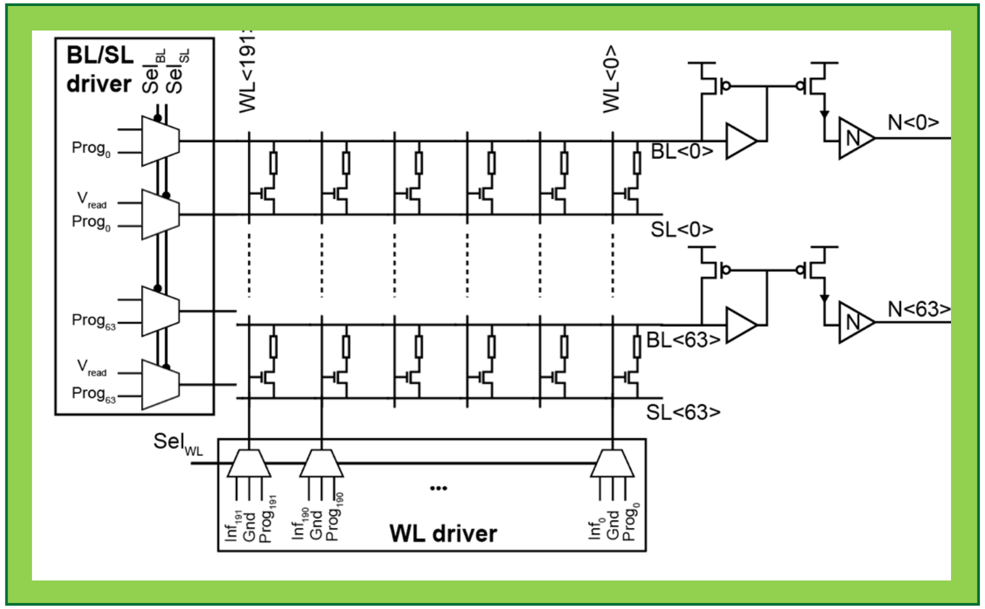
RRAM is used as the memory for both compute and route.

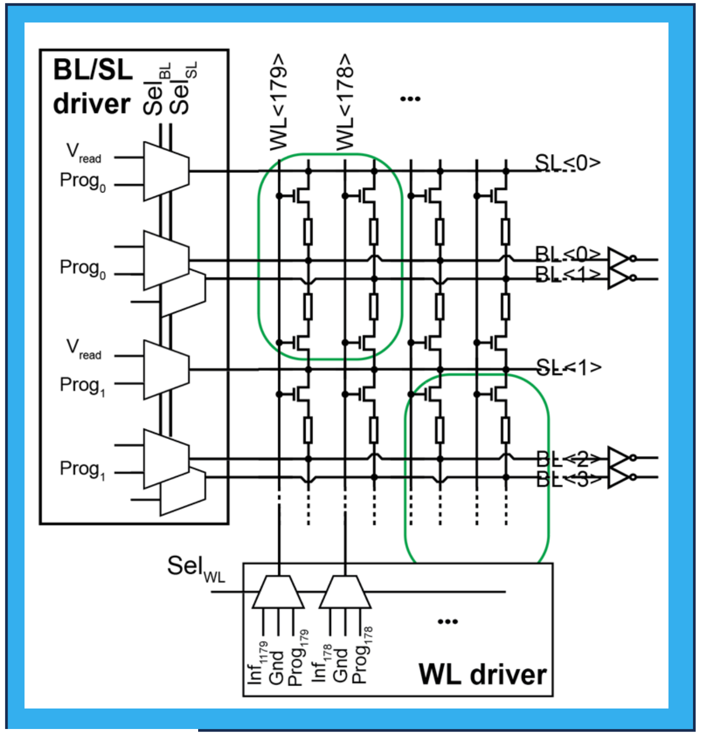
Dalgaty*, Moro*, Demirag*, et al, Nat. Comm. 2024

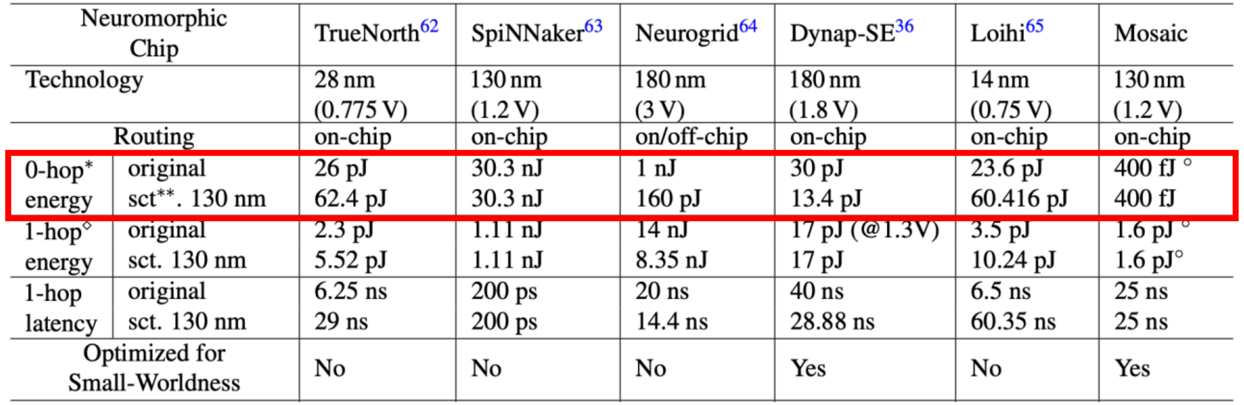
Mosaic: Local connectivity reduces the communication distance.



Mosaic architecture



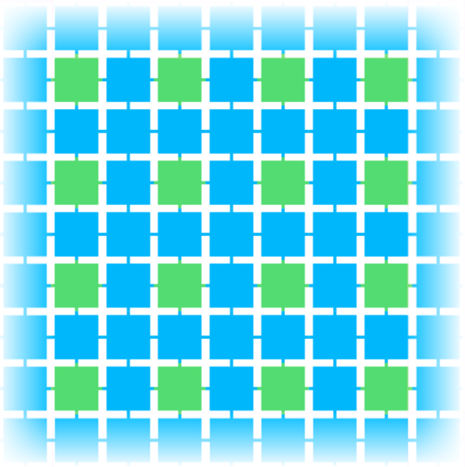




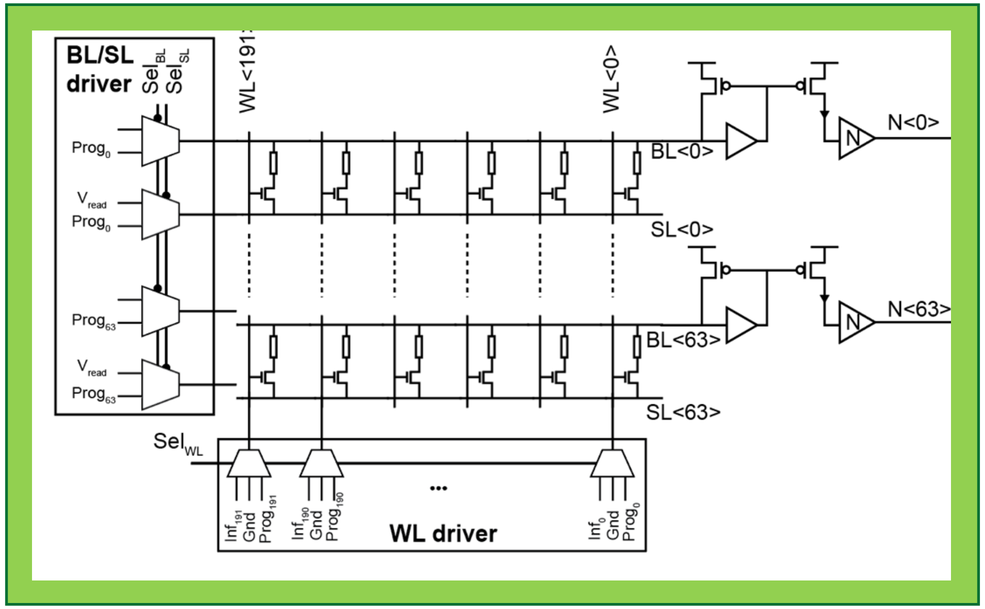
Neuromorphic Chip		TrueNorth[62]	SpiNNaker[63]	Neurogrid[64]	Dynap-SE[36]	Loihi[65]	Mosaic
Technology		28 nm (0.775 V)	130 nm (1.2 V)	180 nm (3 V)	180 nm (1.8 V)	14 nm (0.75 V)	130 nm (1.2 V)
Routing		on-chip	on-chip	on/off-chip	on-chip	on-chip	on-chip
0-hop* energy	original	26 pJ	30.3 nJ	1 nJ	30 pJ	23.6 pJ	400 fJ °
	sct**. 130 nm	62.4 pJ	30.3 nJ	160 pJ	13.4 pJ	60.416 pJ	400 fJ
1-hop◇ energy	original	2.3 pJ	1.11 nJ	14 nJ	17 pJ (@1.3V)	3.5 pJ	1.6 pJ °
	sct. 130 nm	5.52 pJ	1.11 nJ	8.35 nJ	17 pJ	10.24 pJ	1.6 pJ°
1-hop latency	original	6.25 ns	200 ps	20 ns	40 ns	6.5 ns	25 ns
	sct. 130 nm	29 ns	200 ps	14.4 ns	28.88 ns	60.35 ns	25 ns
Optimized for Small-Worldness		No	No	No	Yes	No	Yes

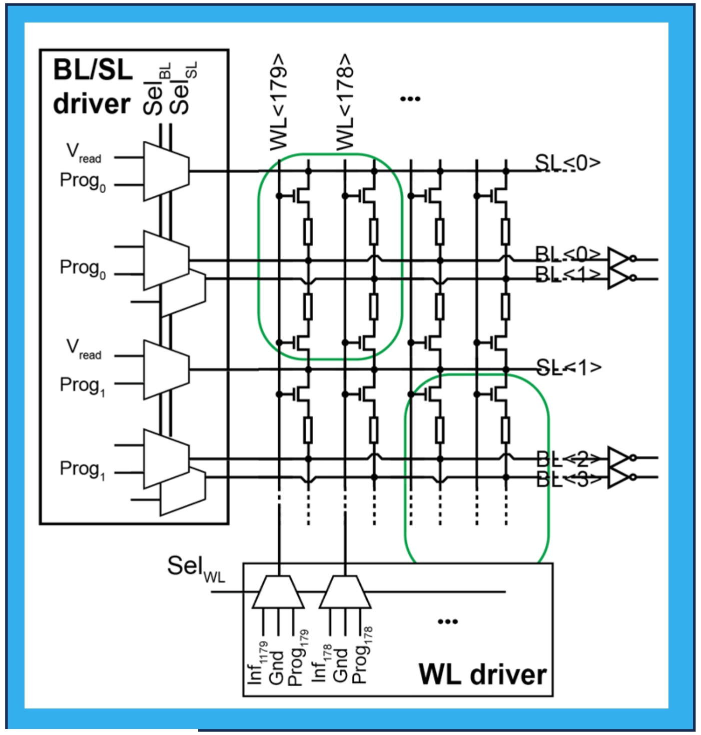
Dalgaty*, Moro*, Demirag*, et al, Nat. Comm. 2024

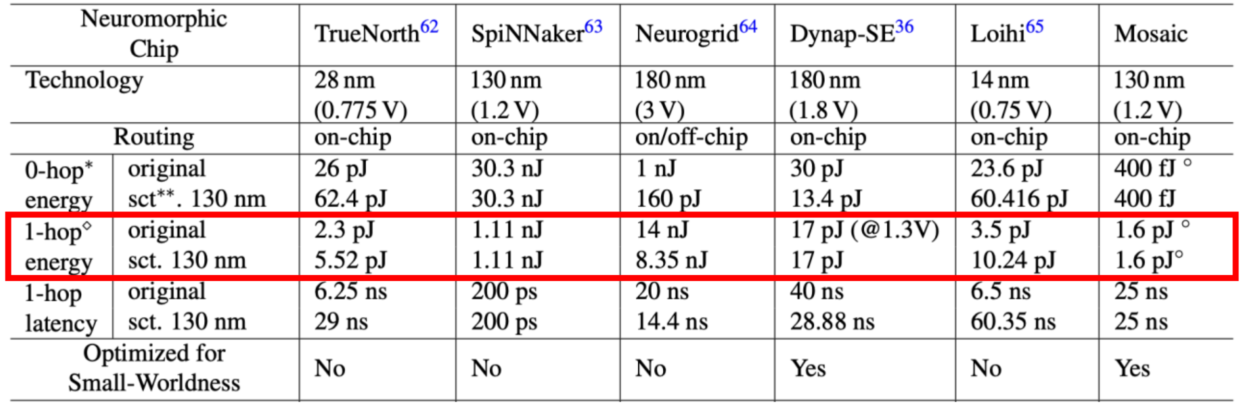
Local connectivity reduces the communication distance.



Mosaic architecture



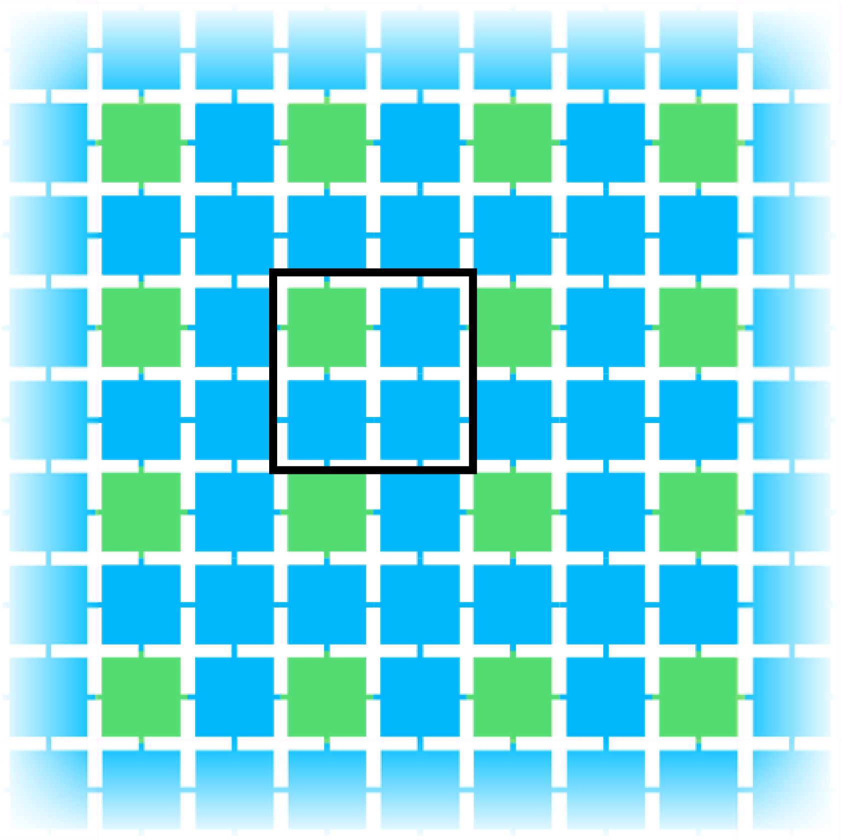




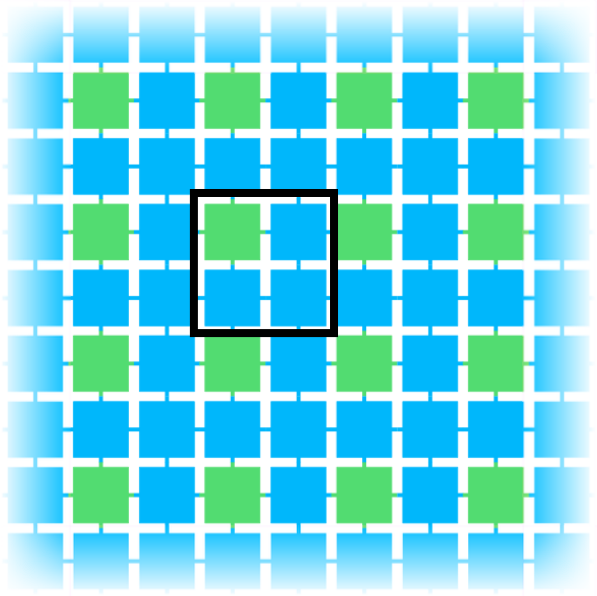
Neuromorphic Chip		TrueNorth[62]	SpiNNaker[63]	Neurogrid[64]	Dynap-SE[36]	Loihi[65]	Mosaic
Technology		28 nm (0.775 V)	130 nm (1.2 V)	180 nm (3 V)	180 nm (1.8 V)	14 nm (0.75 V)	130 nm (1.2 V)
Routing		on-chip	on-chip	on/off-chip	on-chip	on-chip	on-chip
0-hop* energy	original	26 pJ	30.3 nJ	1 nJ	30 pJ	23.6 pJ	400 fJ °
	sct**. 130 nm	62.4 pJ	30.3 nJ	160 pJ	13.4 pJ	60.416 pJ	400 fJ
1-hop◇ energy	original	2.3 pJ	1.11 nJ	14 nJ	17 pJ (@1.3V)	3.5 pJ	1.6 pJ °
	sct. 130 nm	5.52 pJ	1.11 nJ	8.35 nJ	17 pJ	10.24 pJ	1.6 pJ°
1-hop latency	original	6.25 ns	200 ps	20 ns	40 ns	6.5 ns	25 ns
	sct. 130 nm	29 ns	200 ps	14.4 ns	28.88 ns	60.35 ns	25 ns
Optimized for Small-Worldness		No	No	No	Yes	No	Yes

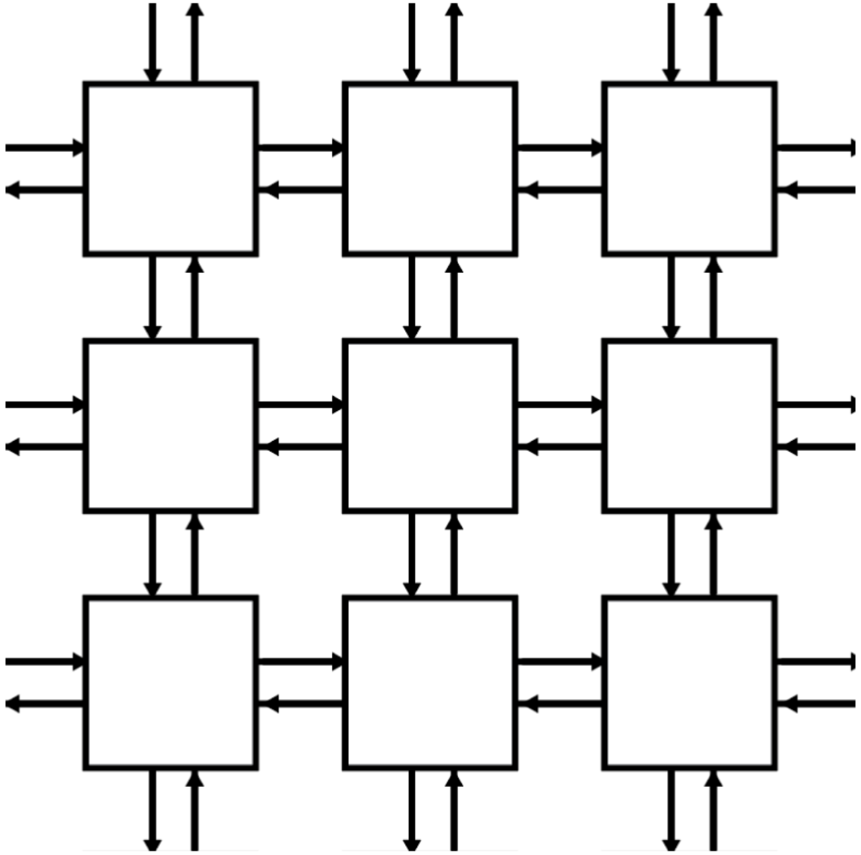
Dalgaty*, Moro*, Demirag*, et al, Nat. Comm. 2024

Mosaic: Scalable fully RRAM in-memory compute and routing chip

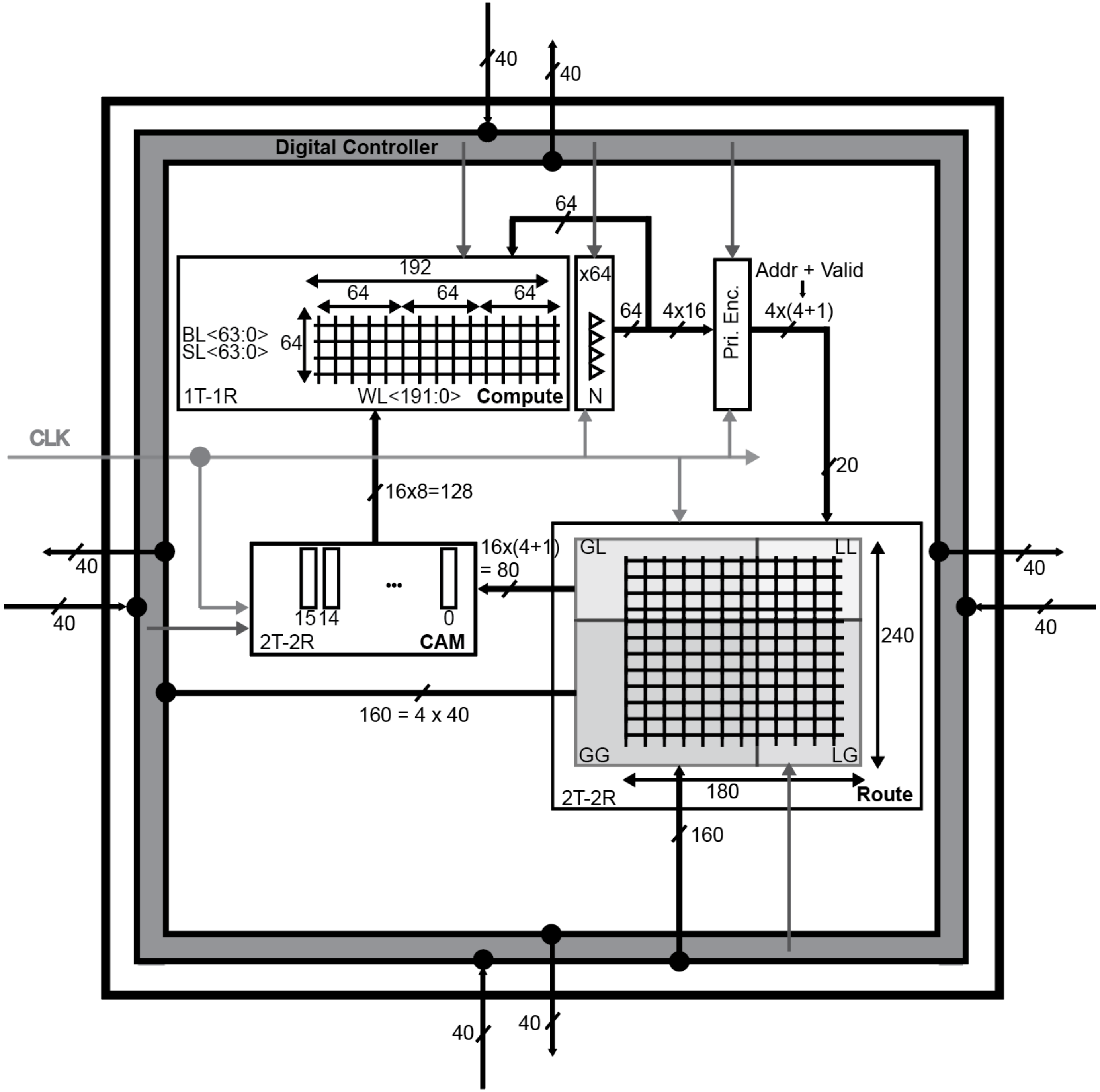


Mosaic: Scalable fully RRAM in-memory compute and routing chip





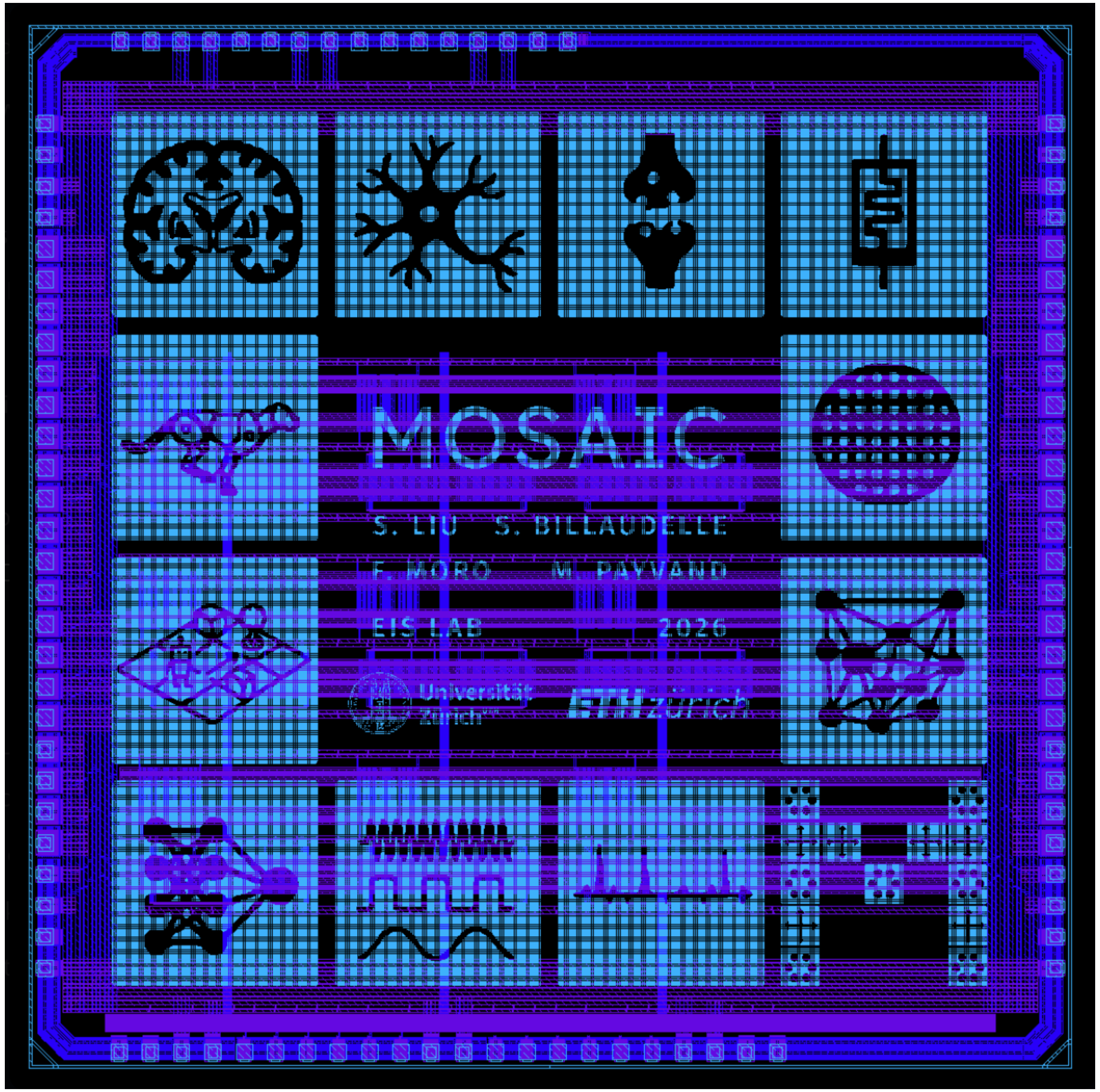
One Mosaic tile



Mosaic: First fully-RRAM in-memory compute AND route

Archicted by:

EIS -LAB



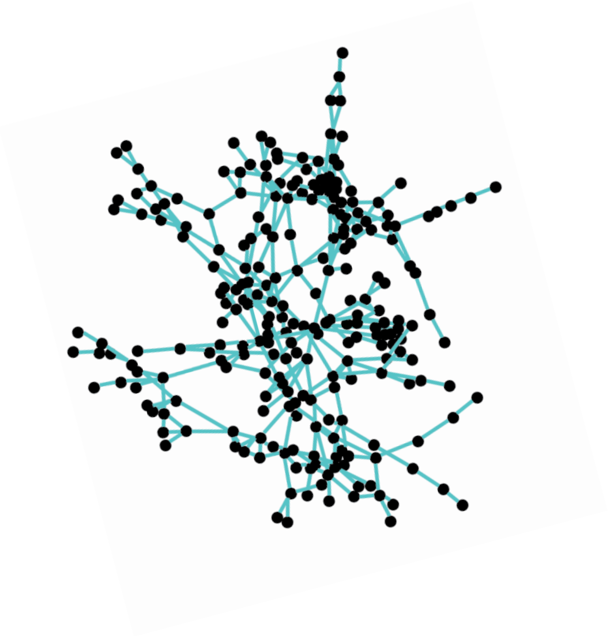
Built by:

Sebastian
Billaudelle

Siqi
Liu

Acknowledgement: The amazing DZ and Chenxi Wen for help w/ the tape-out! ❤️

Part II: Summary



Local connectivity enables scalable efficient architectures.

- Localizing connectivity brings power advantage.
- Mosaic: the first fully RRAM In-Memory Compute AND route chip!!
- Stateful computation brings advantage in masked environments.

Compilation and placement becomes important.
Co-design is important.

Acknowledgements

University of Zurich^UZH

ETH zürich

Funding

FNSNF
FONDS NATIONAL SUISSE
DE LA RECHERCHE SCIENTIFIQUE

METASPIN

VolkswagenStiftung

Support

Design Center, IIS, ETHZ

Website

mpayvand@ethz.ch
melika@ini.uzh.ch

Collaborations

Giacomo Indiveri
Yassine Taoudi

leti ceatech

Elisa Vianello
Simone D’Agostino
Thomas Dalgaty

JÜLICH FORSCHUNGSZENTRUM

Alpha Renner
Emre Neftci

UNIVERSITÄT BERN

Julian Goeltz
Mihai Petrovici

UNIVERSITÄT HEIDELBERG ZUKUNFT SEIT 1386

Johannes
Schemmel

Imperial College London

Dan Goodman
Gabriel Bena