Turing or Non-Turing ?
That Is The Question

how the BrainScaleS 2nd generation architecture proposes some answers that support the quest of bio-inspired artificial intelligence

Johannes Schemmel

Electronic Vision(s) Group
Kirchhoff Institute for Physics
Heidelberg University, Germany
in collaboration with
TU Dresden, Frauenhofer IZM Berlin and EPFL Lausanne

Bio-inspired artificial intelligence (Bio-AI)

REALIZING future computing based on biological information processing ⟷ understanding biological information processing

artificial system inspired by neuroscience

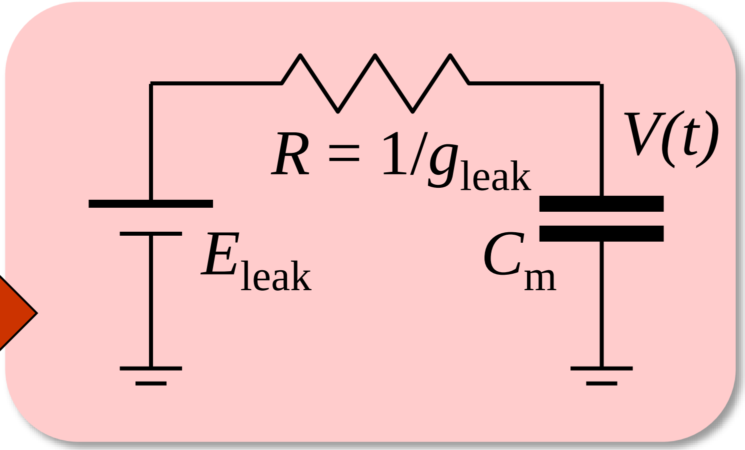
Bio-AI hardware based on spike-based neuromorphic computing

- model imprinted into hardware (rather than being simulated)
- goal: overcoming the power wall of Turing-based computing
- find local learning rules
- a lot of unknowns:
  - classical AI (DCNN) heavily relies on numerical precision for training
  - novel devices not yet available
  - CMOS best option, but still very-expensive for research groups
  - no spike-based algorithms for application-level performance (hen-and-egg problem)

Neuromorphic computing with physical model systems

Consider a simple physical model for the neuron's cell membrane potential *V:*

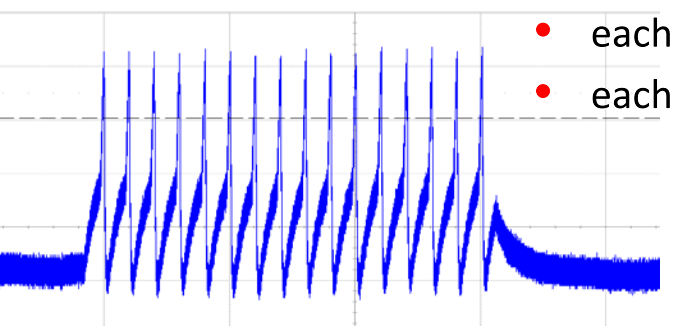
$$C_{\mathrm{m}} \frac{dV}{dt} = g_{\mathrm{leak}} \left(E_{\mathrm{leak}} - V \right)$$

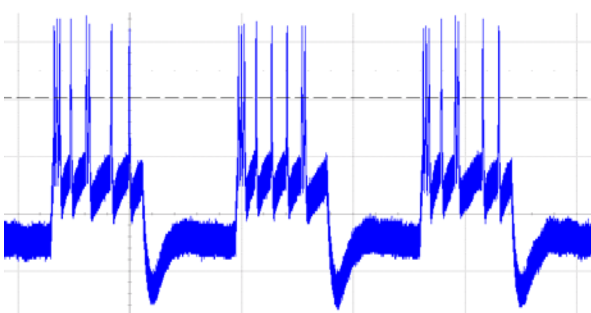


$$\left.\frac{dV}{dt}\right|_{bio} \ll \left.\frac{dV}{dt}\right|_{VLSI}$$

→ accelerated neuron model

- continuous time
  - fixed acceleration factor (we use 10^3 to 10^5)
- no multiplexing of components storing model variables
  - each neuron has its membrane capacitor
  - each synapse has a physical realization





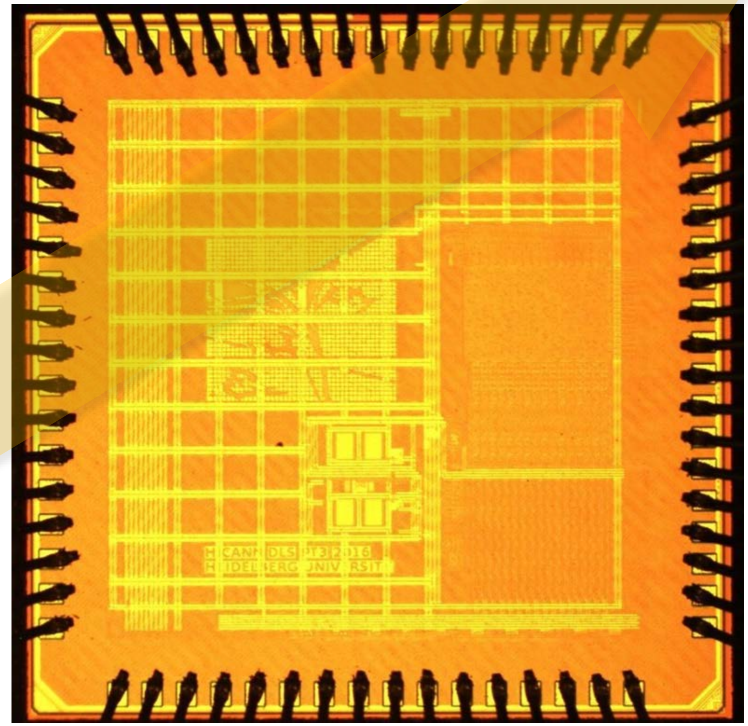
The Heidelberg BrainScaleS physical model systems

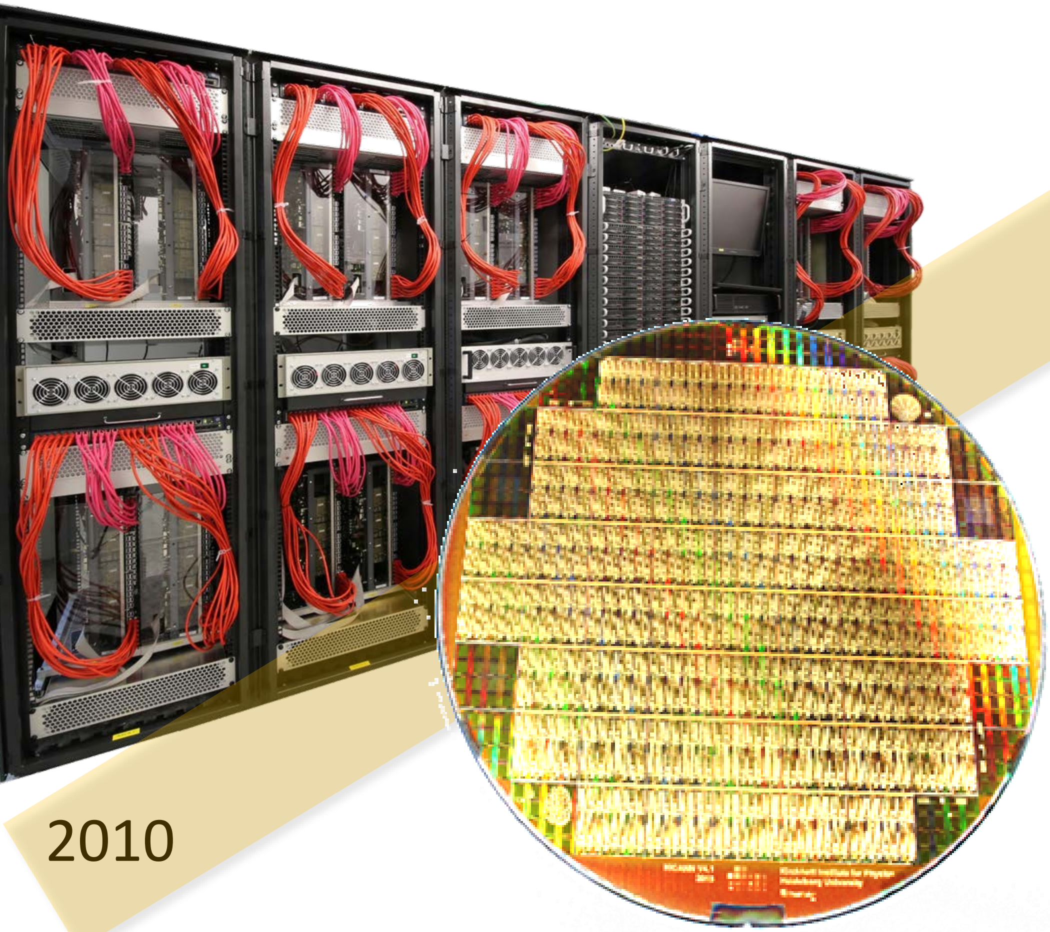
BrainScaleS 1: wafer-scale Neuromorphic system

introduced:

- wafer-scale event-communication
- AdEx neuron with >10k inputs

2020





2010

BrainScaleS 2: hybrid plasticity

introduced:

- software-controlled local plasticity
- non-linear dendrites and structured neurons

Human Brain Project – Benchmarking NMC

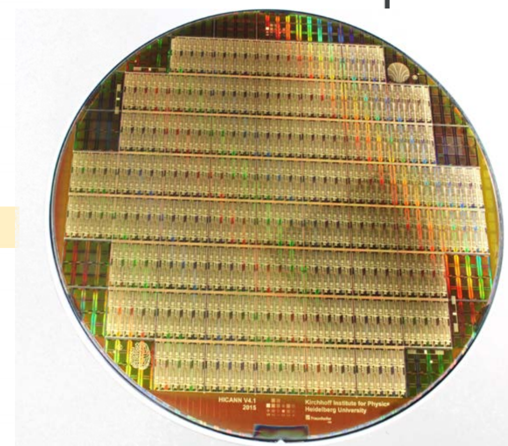
solving constraint satisfaction problems with spiking neurons

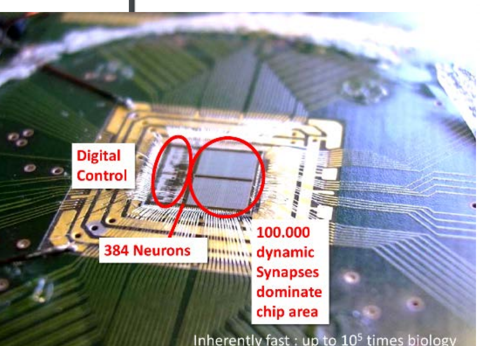
Universität Bielefeld

Benchmarking Spiking Sudoku Solver

CITEC

Platform	#Solved Sudokus	Bio-time to sol. in ms	Real-time to sol. in s	Power in W	Energy to Solution in J
		4 × 4 Sudokus using `architecture#1`			
NEST	100	214.6 ± 263.1	0.03	17	0.5
SpiNN-5	97	357.1 ± 688.9	3.57	23.3	83.2
BrainScaleS	86	3241.9 ± 4573.1	$3.24 \cdot 10^{-4}$	NA	†0.0059
		4 × 4 Sudokus using `architecture#2`			
NEST	100	214.6 ± 263.1	0.03	17	0.5
SpiNN-3	99	241.2 ± 250.0	2.41	2.7	6.5
		4 × 4 Sudokus using `architecture#3`			
NEST	100	286.0 ± 377.6	0.12	17	2.0
SpiNN-3	100	319.0± 437.3	3.19	2.8	8.9
Spikey	75	3745.8± 6041.11	$3.75 \cdot 10^{-4}$	5.6	0.0021
		6 × 6 Sudokus using `architecture#1`			
NEST	98	1769.2 ± 1909.1	0.62	17	10.5
SpiNN-5	99	2084.8 ± 2703.3	20.85	23.5	490.0
		6 × 6 Sudokus using `architecture#2`			
NEST	98	1769.2 ± 1909.1	0.62	17	10.5
SpiNN-3	91	1641.1 ± 1463.0	16.41	2.7	44.3

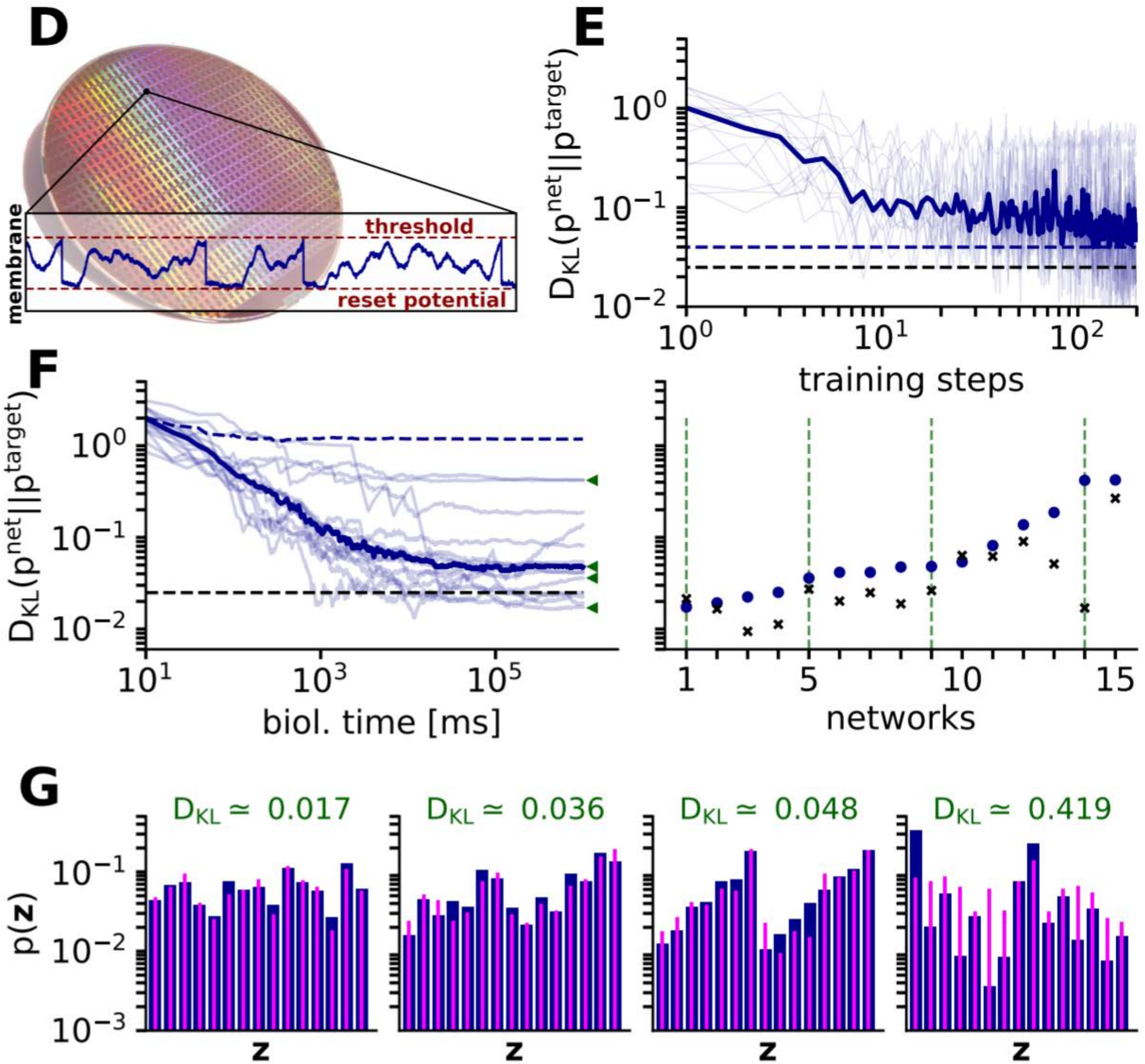




- NEST: 4 threads on i7-4710MQ; simulation power – idle power
- BrainScaleS: calculation assumes 5pJ per pre-synaptic event
- Spikey/SpiNNaker: measure at 5V/12V supply lane

slide taken from University Bielefeld presentation by Christoph Ostrau at SP9 meeting, Graz ´19

Stochastic model example: sampling from multiple neural Boltzmann machines



D. Dold et.al., https://arxiv.org/abs/1809.08045v1

Observations

- Non-Turing physical model can autonomously, fast and power-efficient replicate learned distributions
- As previously demonstrated (NICE 17), same is true for DCNN-inference

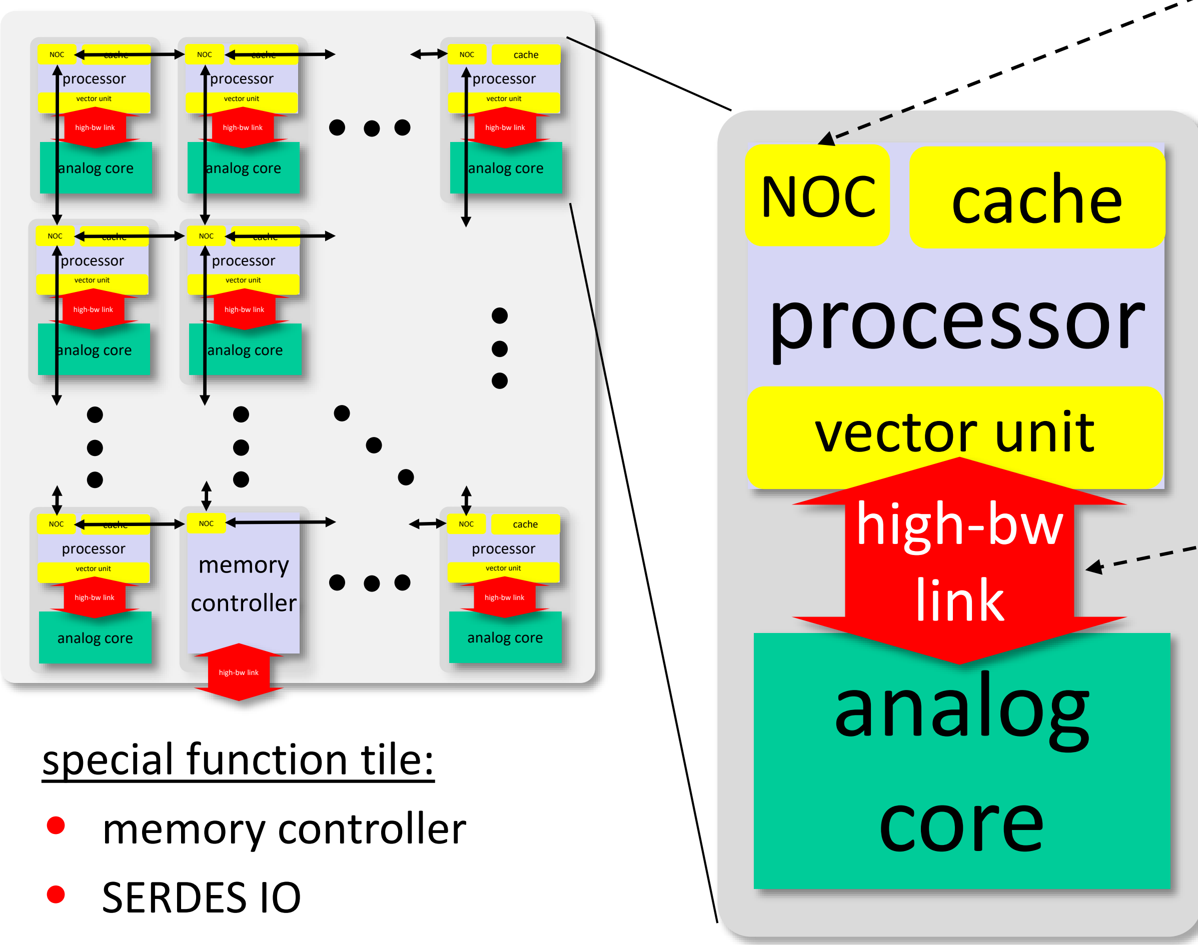
Turing-based computing is used in multiple places in these experiments

- training
- system initialization
- hardware calibration
- runtime control
- input/output data handling

Add classical, Turing-based system to analog NM core?

Why not the other way round?

Analog neuromorphic system as co-processor



Network-on-chip:

- prioritize event data
- unused bw for CPU
- common address space for neurons and CPUs

high-bandwidth link:

vector unit ←→ NM core

- weights
- correlation data
- routing topology
- event (spikes) IO
- configuration

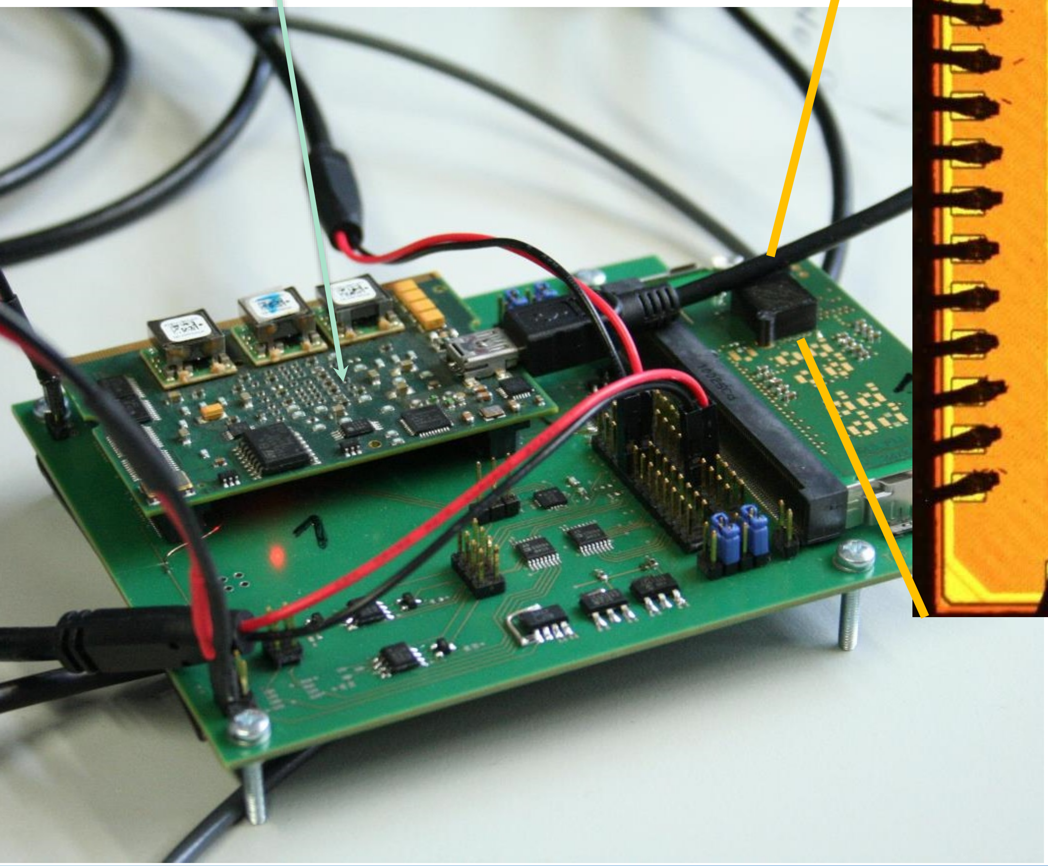
special function tile:

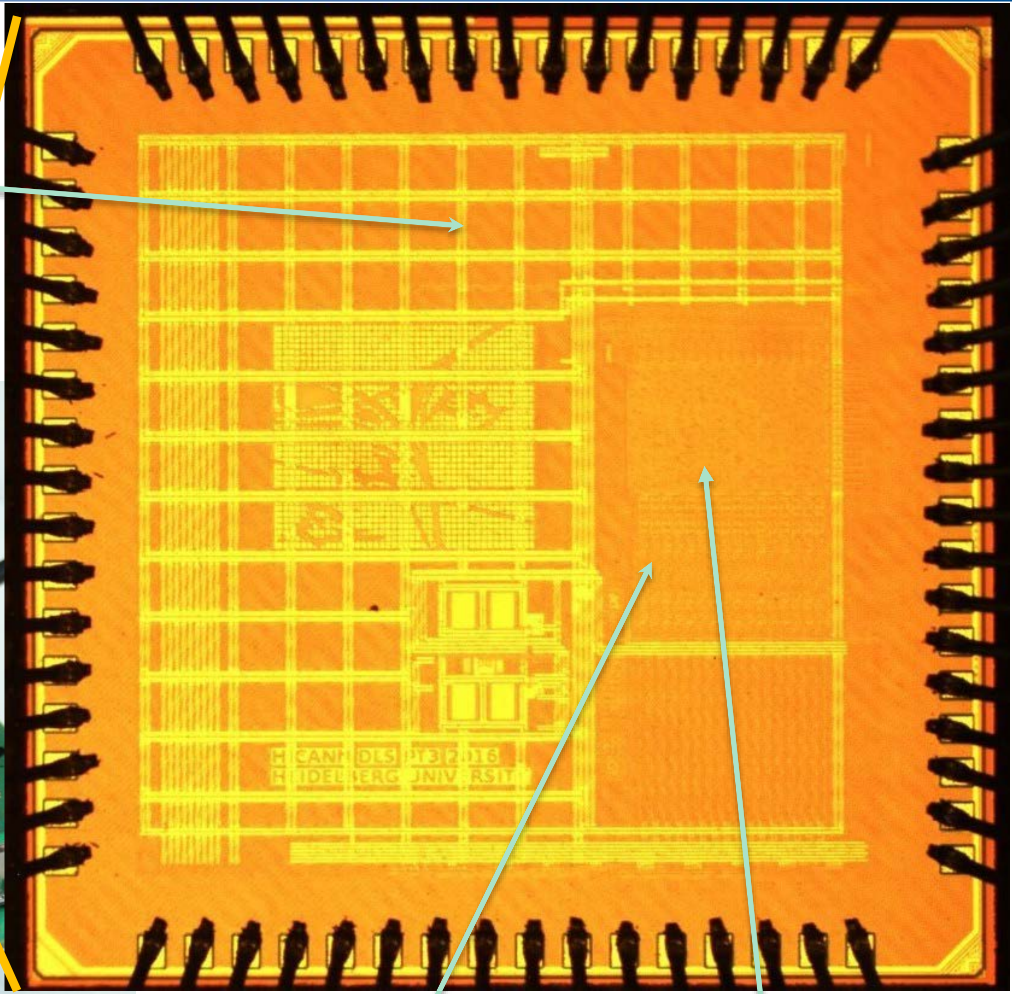
- memory controller
- SERDES IO
- purely digital function unit

BrainScaleS 2 (BSS-2): 2nd generation prototype chip

FPGA based controller board

plasticity processor

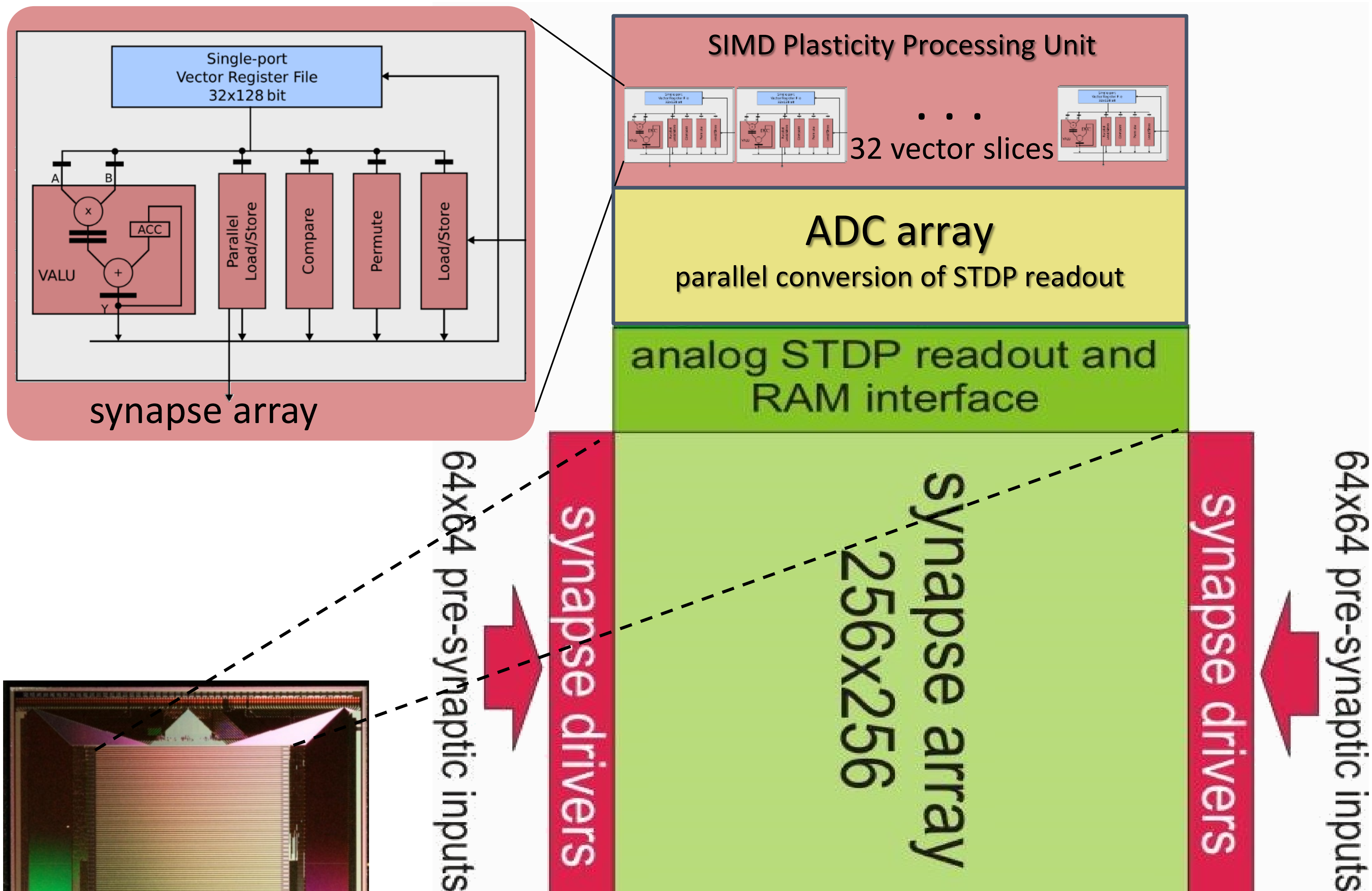




neuron circuits

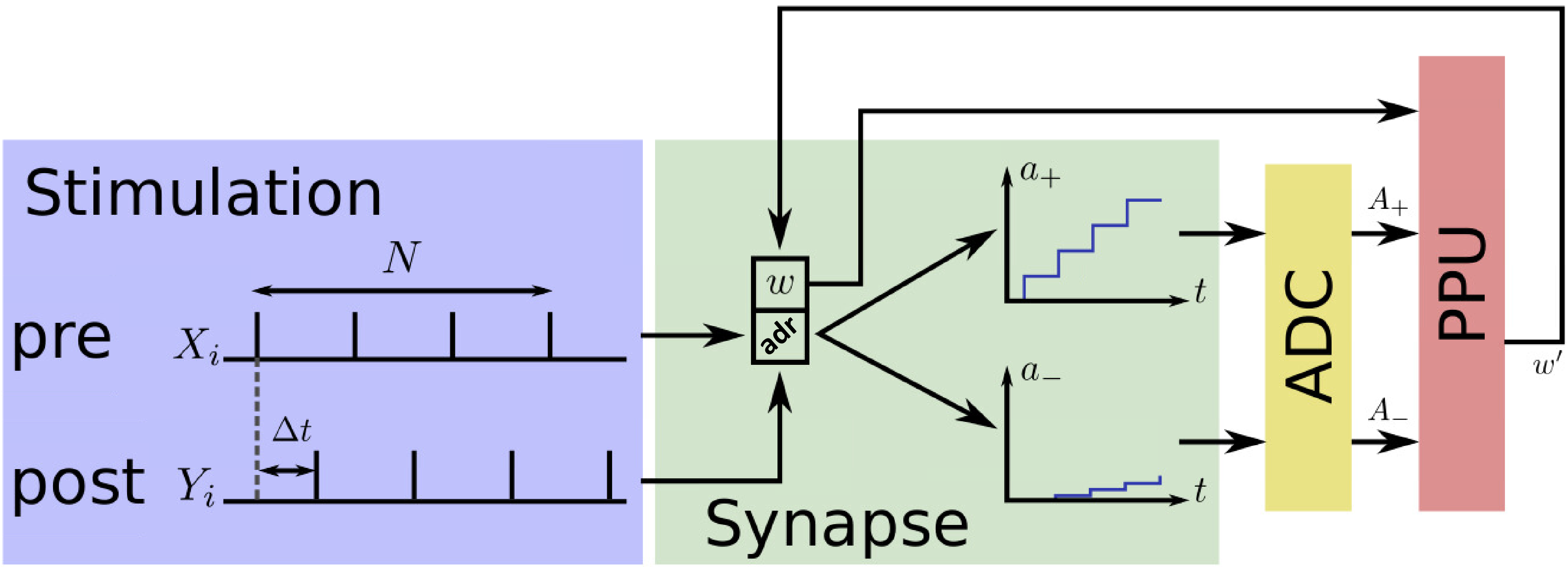
synapse array

BSS-2 uses tightly coupled Turing and Non-Turing compute parts for hybrid plasticity



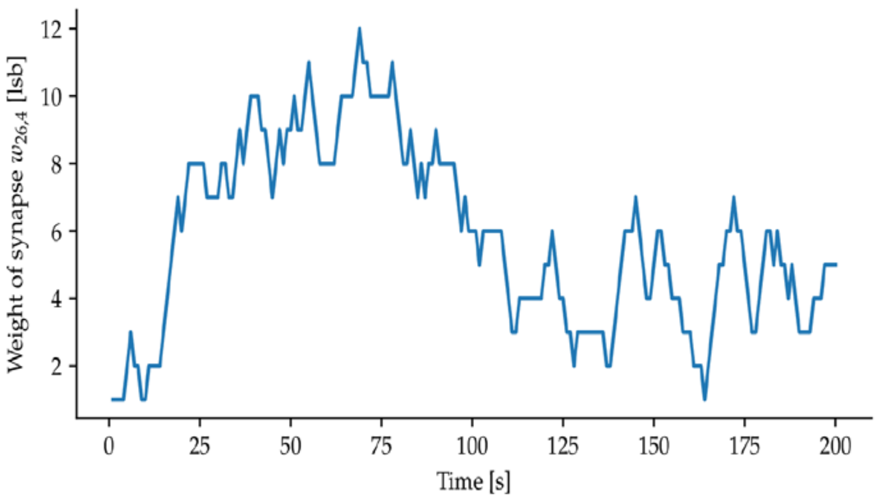
Concept of hybrid plasticity operation

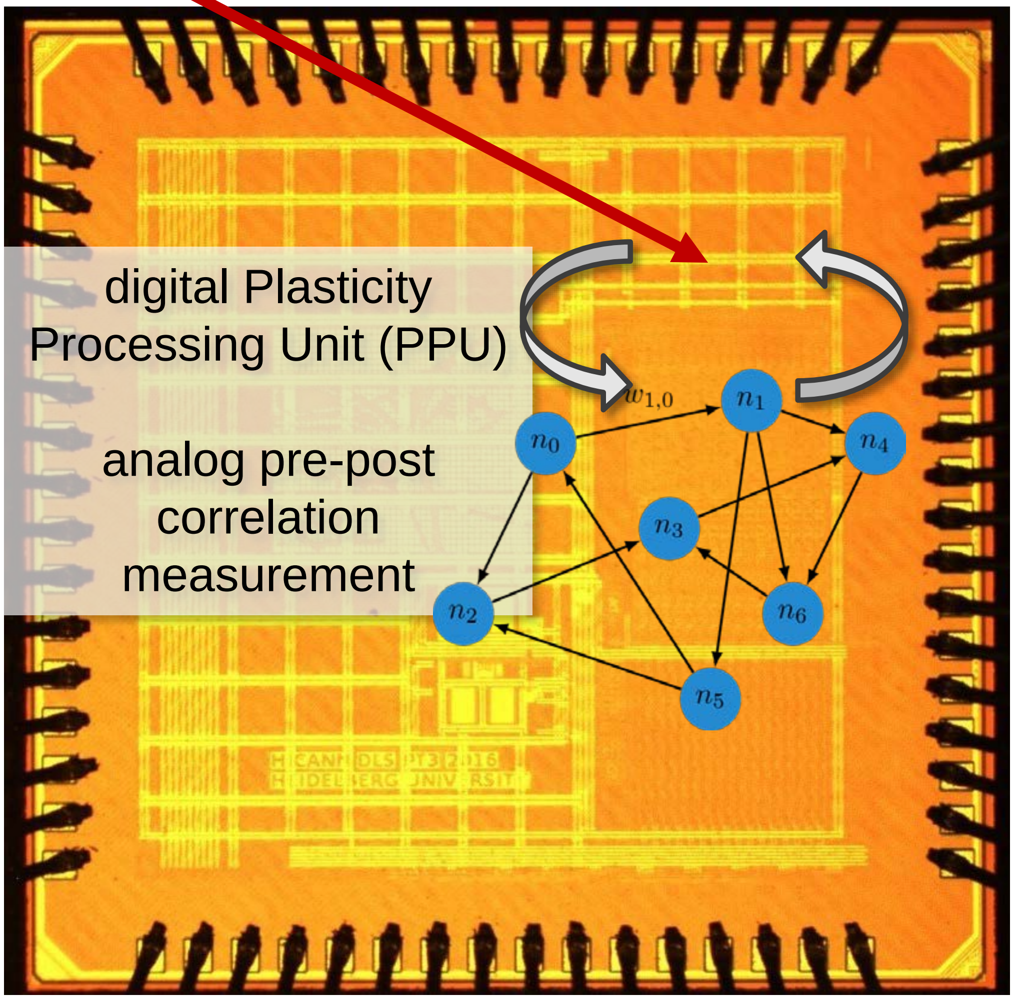
- analog correlation measurement in synapses
- A/D conversion by parallel ADC
- digital Plasticity Processing Units
  → full access to synaptic weights (ω)
  → full access to configuration data (adr)



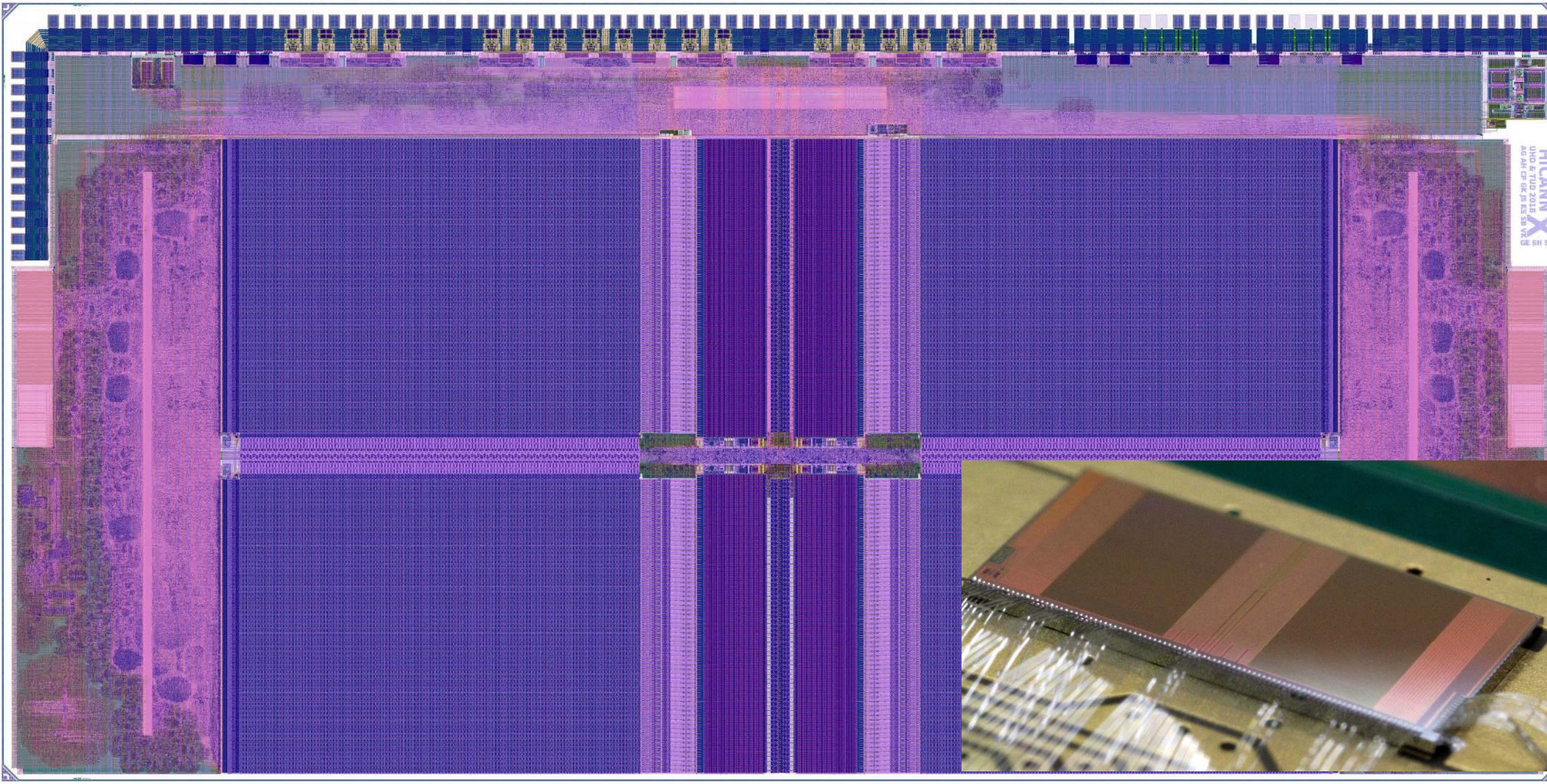
Summary: learning and plasticity with hybrid plasticity

- local plasticity loop **on the chip**
- continuous weight update during network operation
- algorithm can use
  - neuron firing rates
  - compartmental voltages
  - temporal correlations
  - neuromodulatory signals



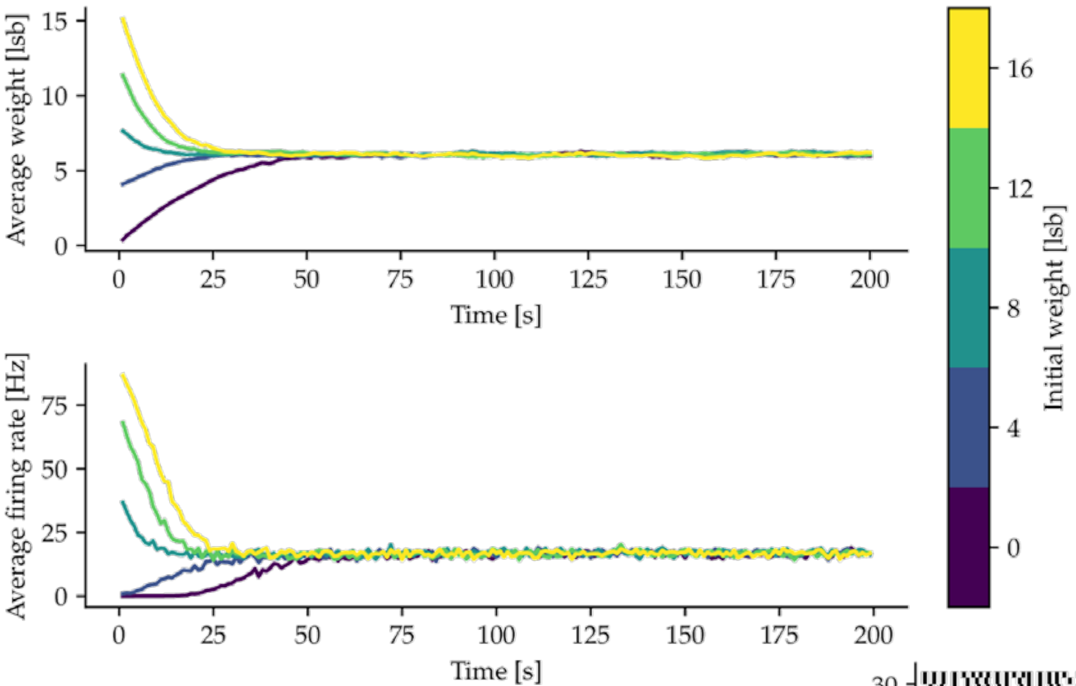


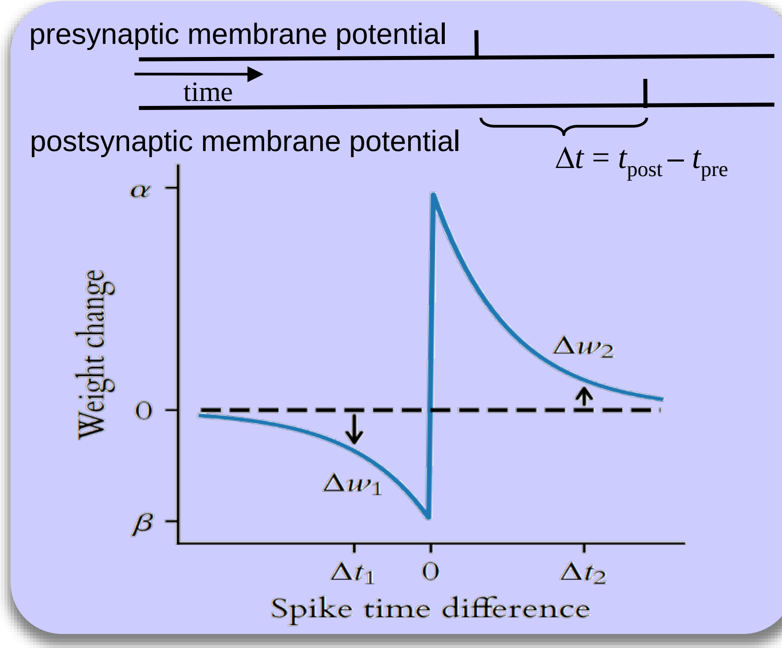
BrainScaleS 2 full-size prototype chip



- 65nm LP-CMOS, power consumption O(10 pJ/synaptic event)
- 128k synapses
- 512 neural compartments
- two SIMD plasticity processing units
- fast ADC for membrane voltage monitoring
- 256k correlation sensors with analog storage (> 10 Tcorr/s max)
- 1024 ADC channels for plasticity input variables
- 32 Gb/s neural event IO
- 32 Gb/s local entropy for stochastic neuron operation
- current prototype not operational due to incomplete production database checking at the manufacturer
  → rerun pending

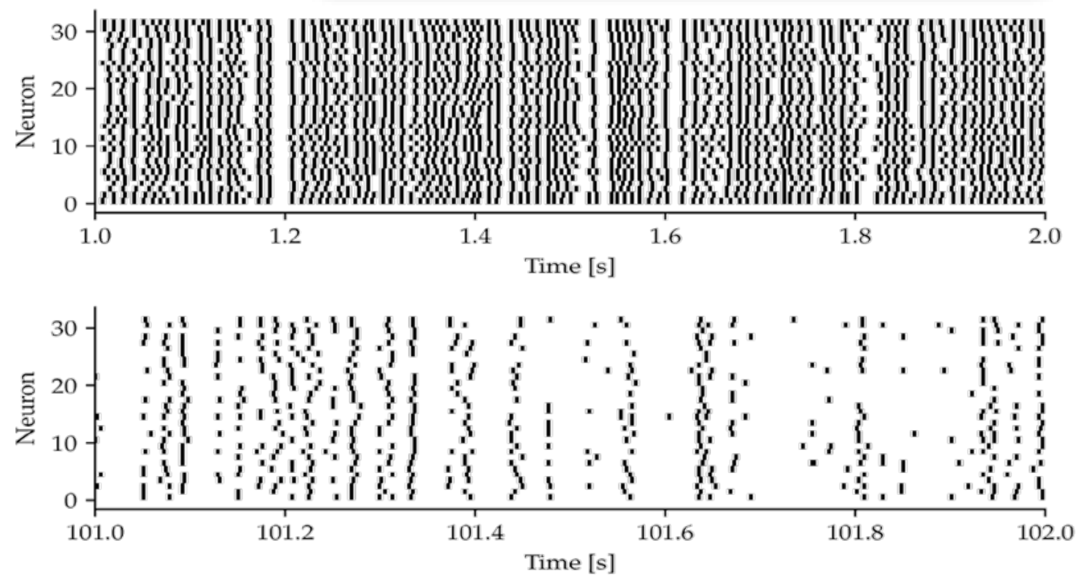
Stabilizing firing rates with spike time dependent plasticity





Wall-time per trace: 200ms

→ acceleration factor of 1000

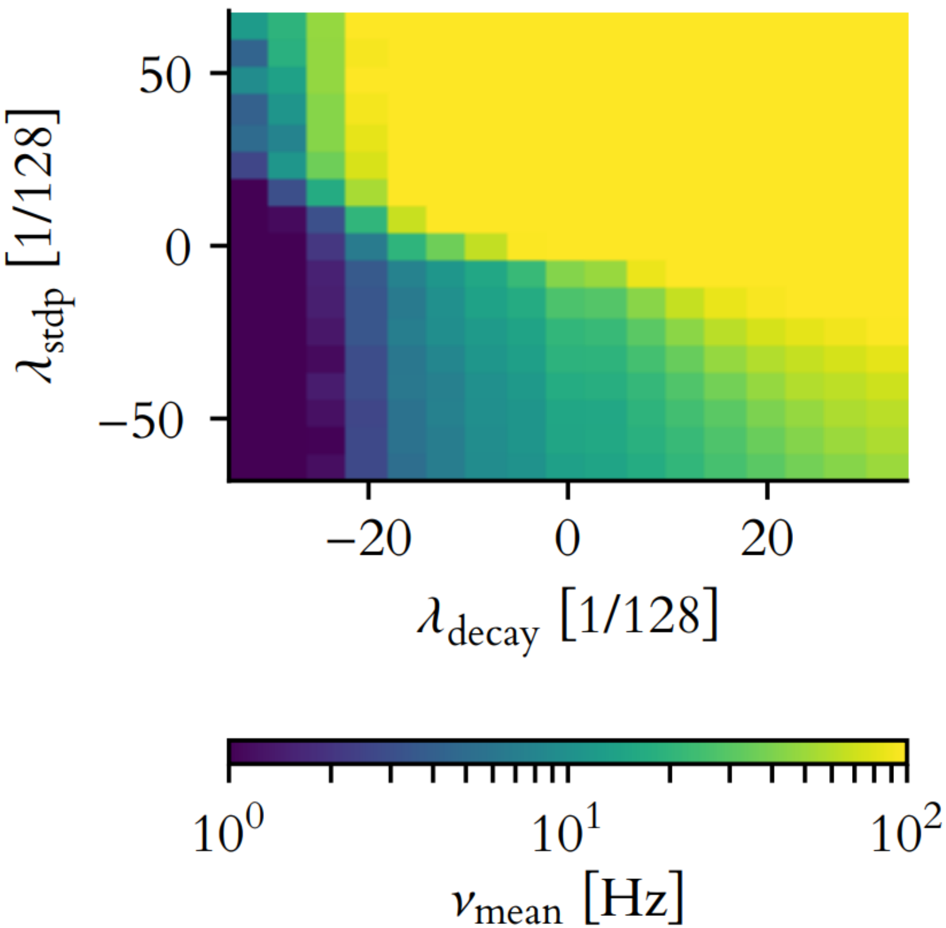


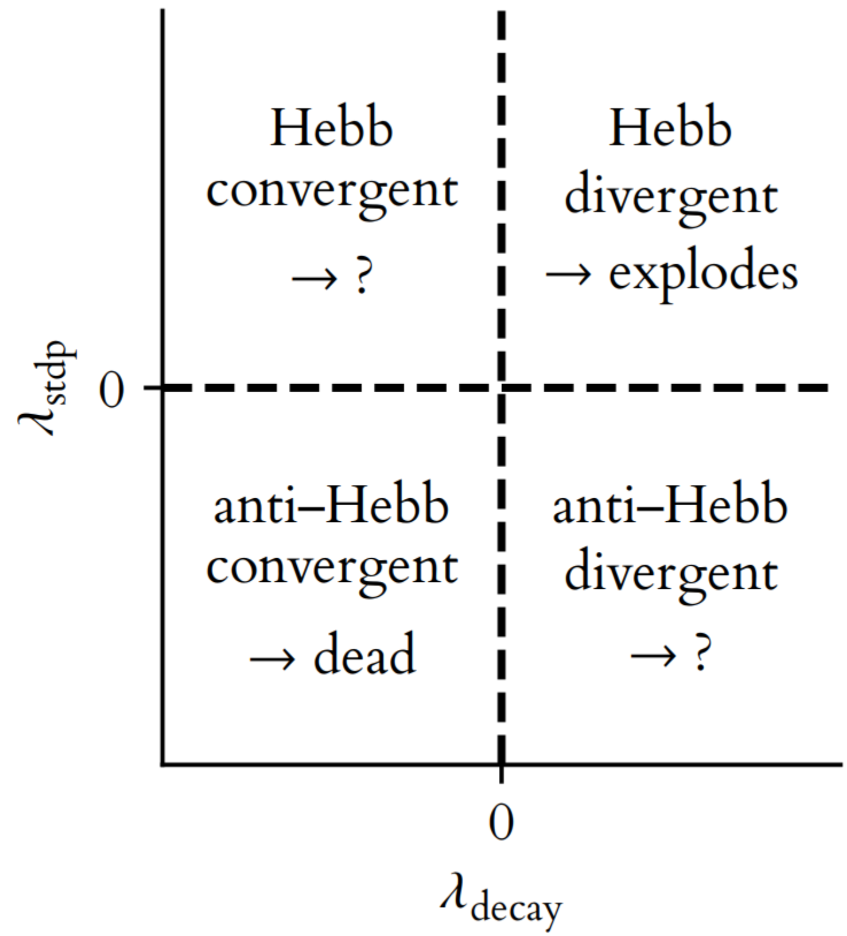
David Stöckel, Master Thesis, Heidelberg University, 2017

Stability analysis for plasticity rules

David Stöckel, Master Thesis, Heidelberg University, 2017

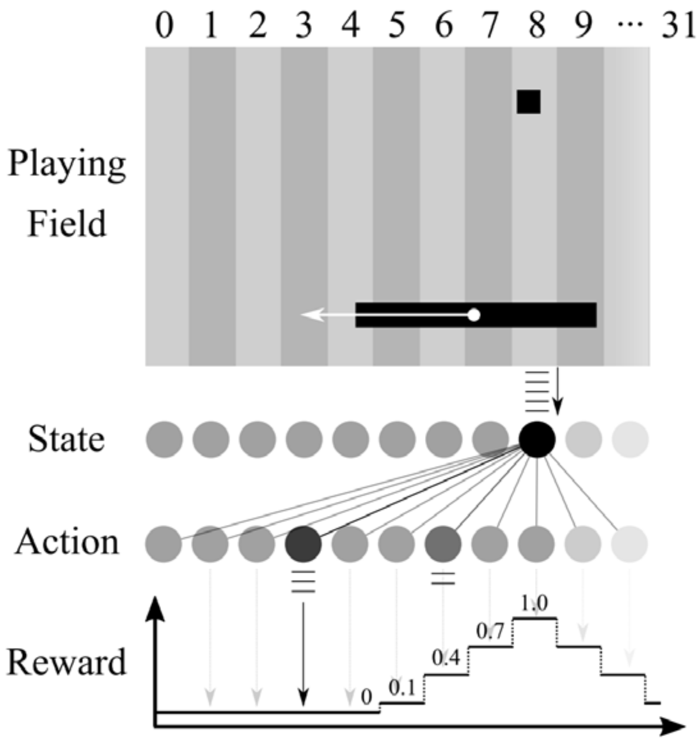
Measure the plasticity parameter phase space

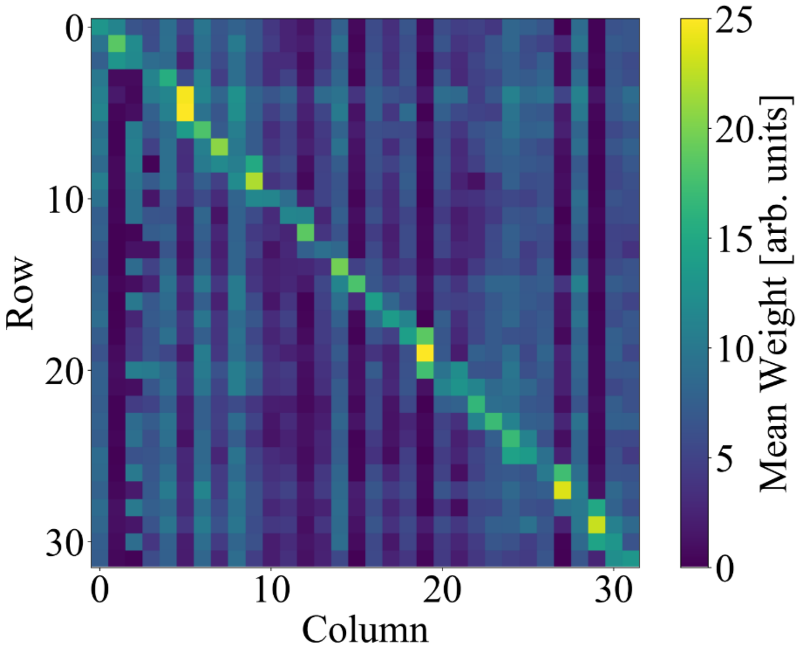


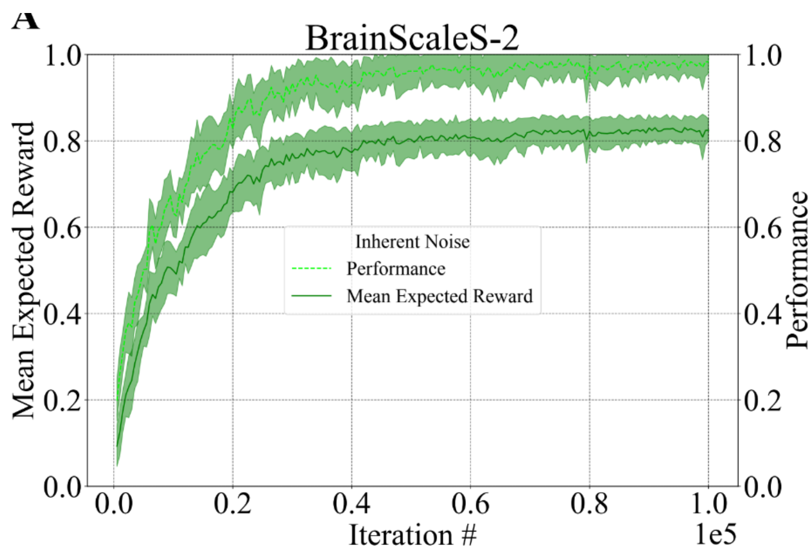


each data point is full plasticity experiment covering 200s biological real time

Learning Pong – tech demo using internal PPU only





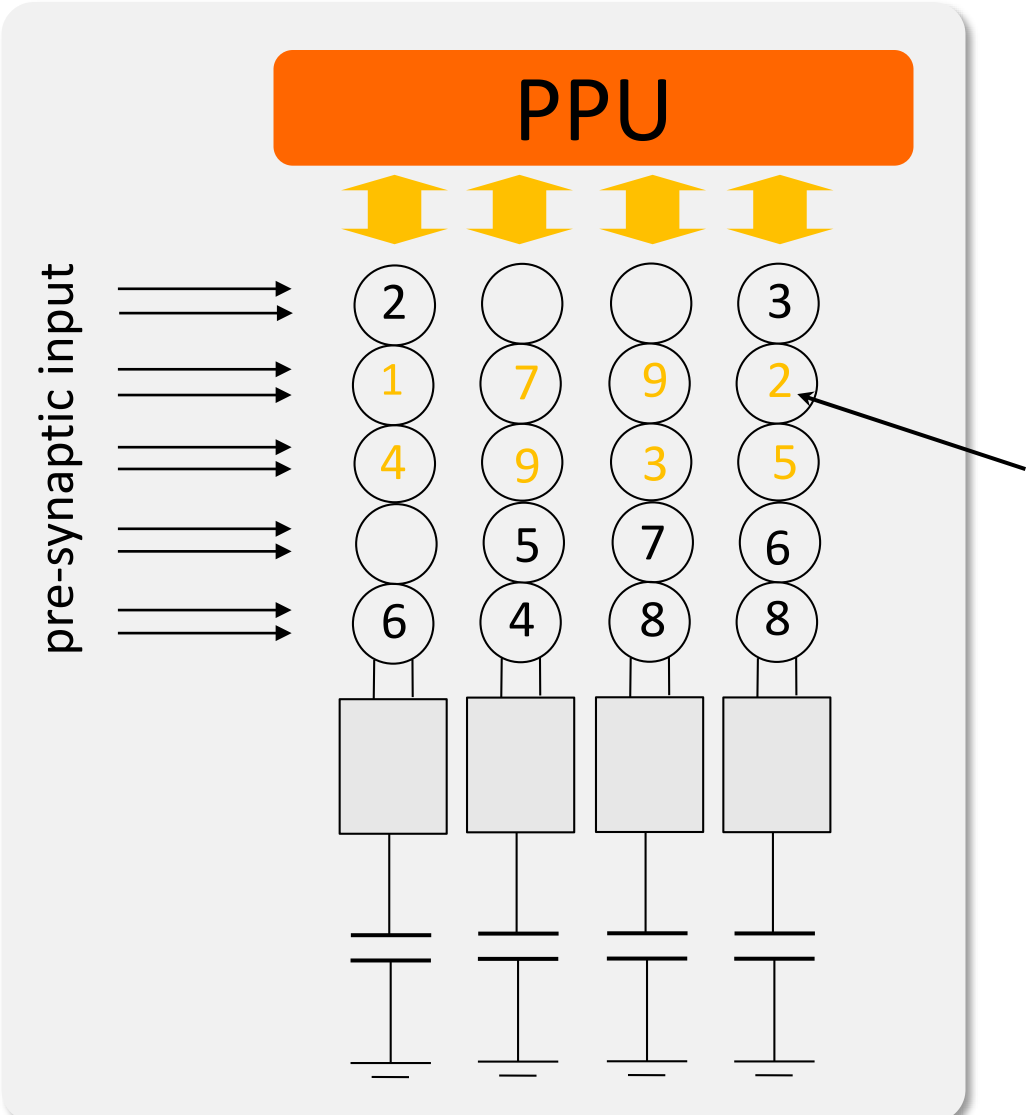


- reinforcement learning rule
- learning is calibration
- experiment runs completely on internal PPU
- 5s for 10k iterations

network time 0.4ms/iteration
23 µJ total chip energy

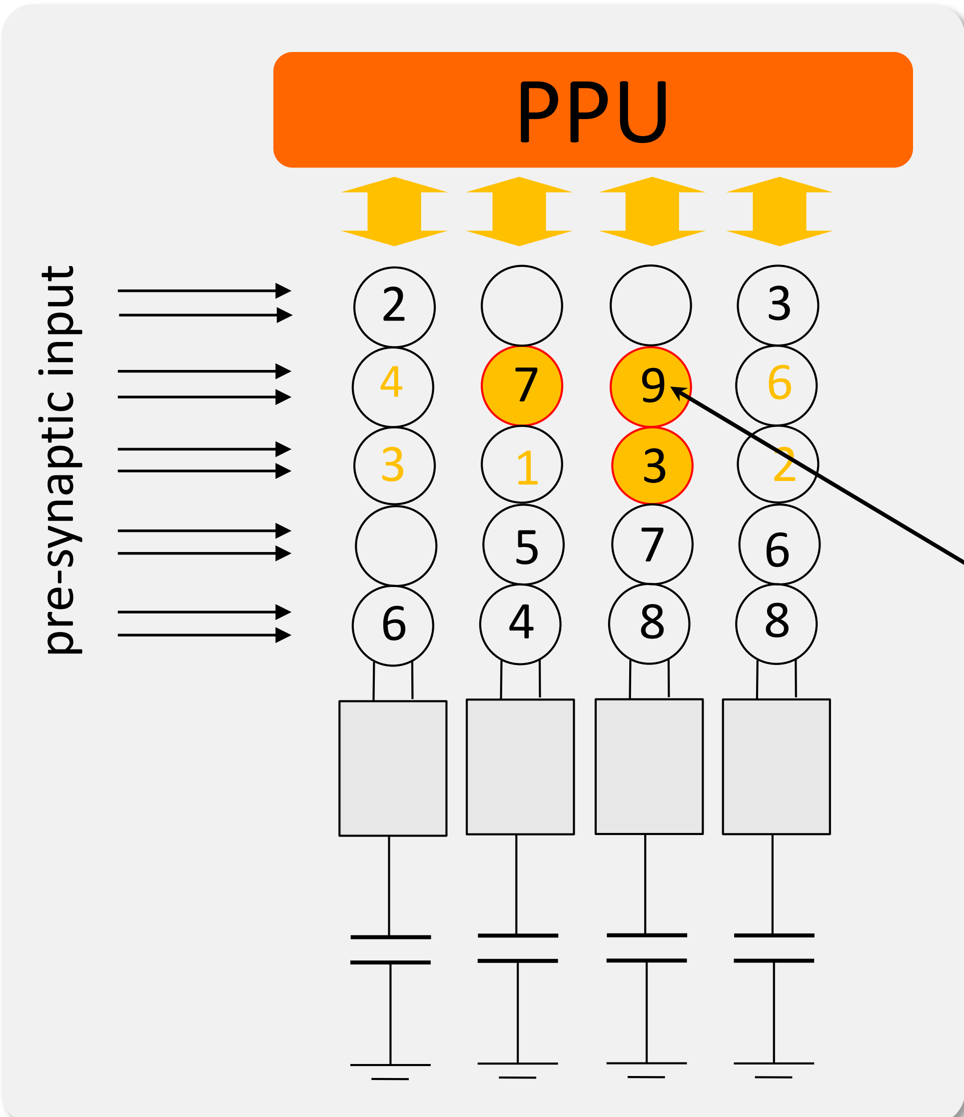
Wunderlich et.al., Demonstrating Advantages …, Front. Neurosci., 2019

Structural Plasticity



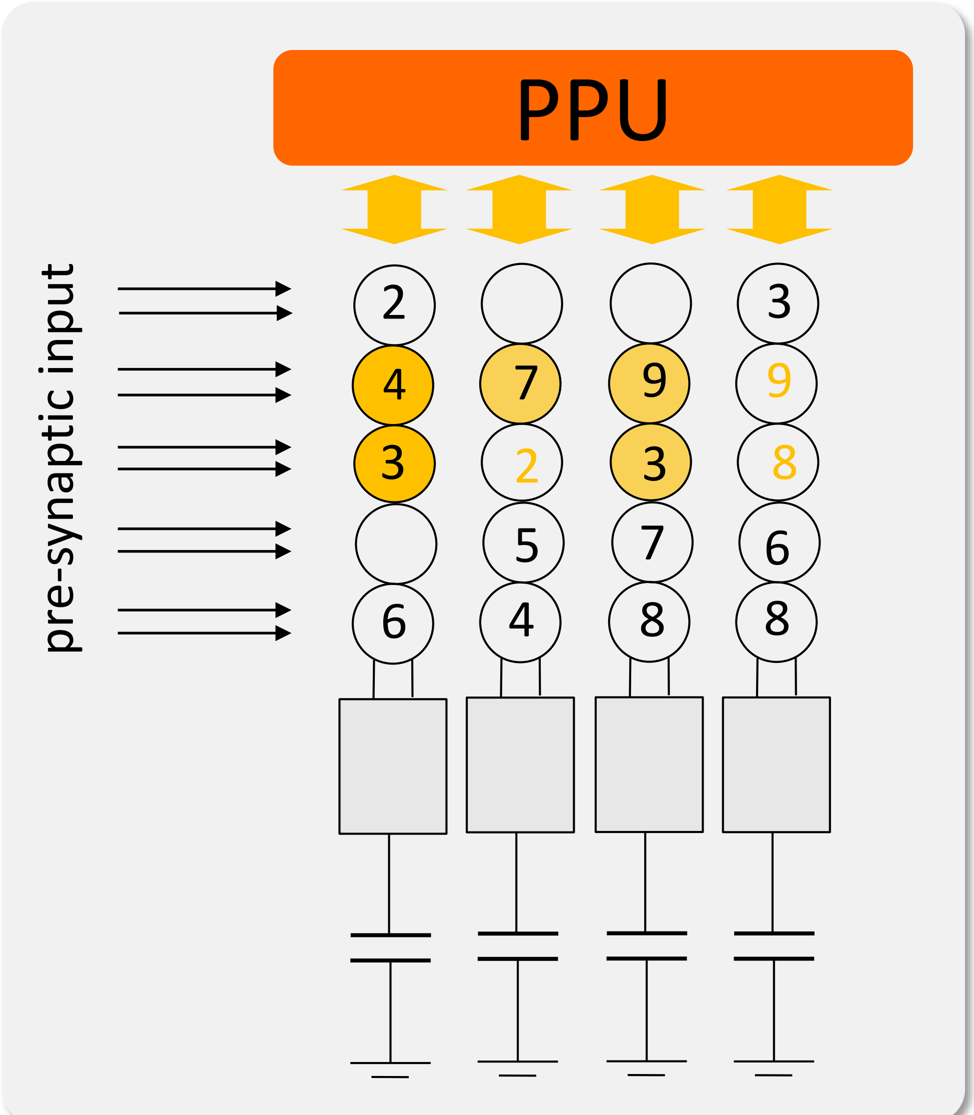
- assign random pre-synaptic neurons
- evaluate correlation

Structural Plasticity



- assign random pre-synaptic neurons
- evaluate correlation
- keep the best

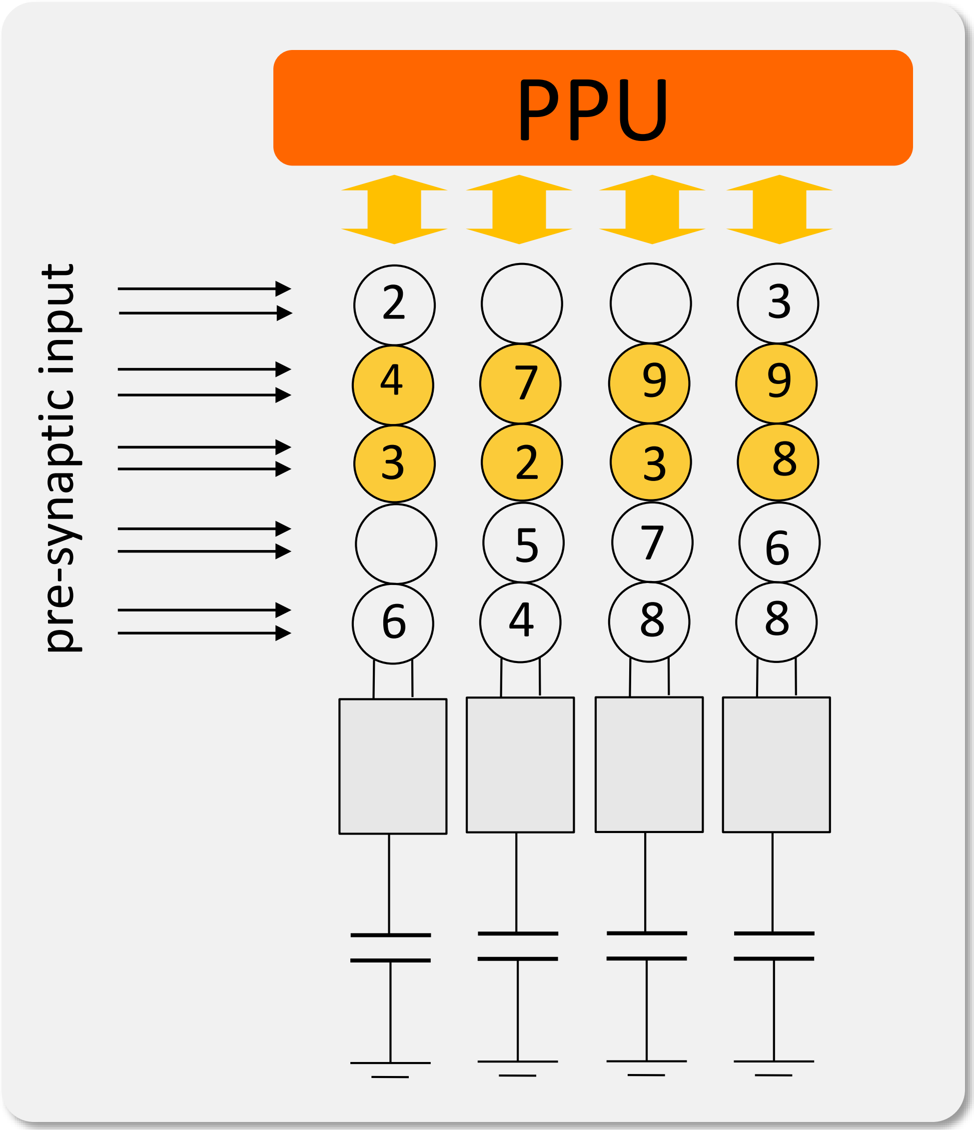
Structural Plasticity



- assign random pre-synaptic neurons
- evaluate correlation
- keep the best

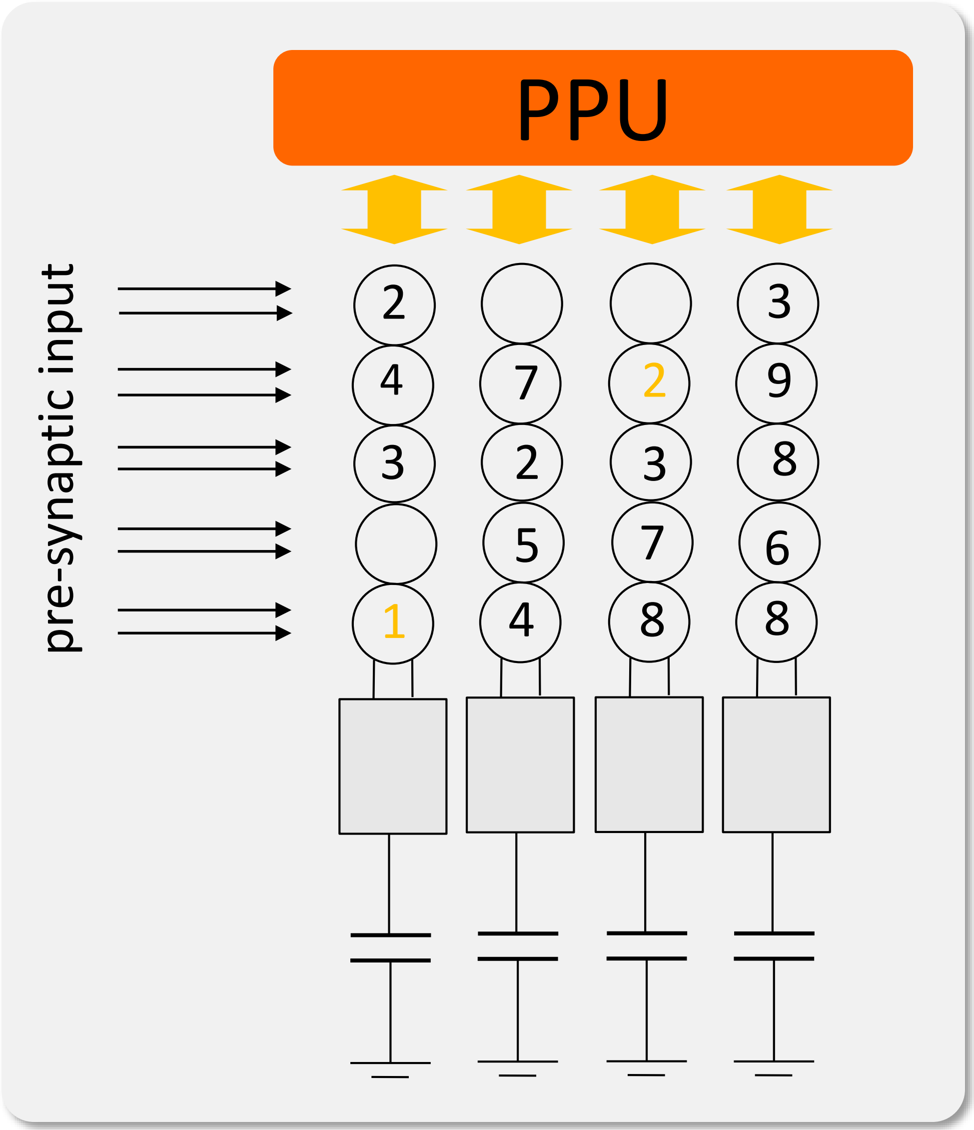
repeat

Structural Plasticity



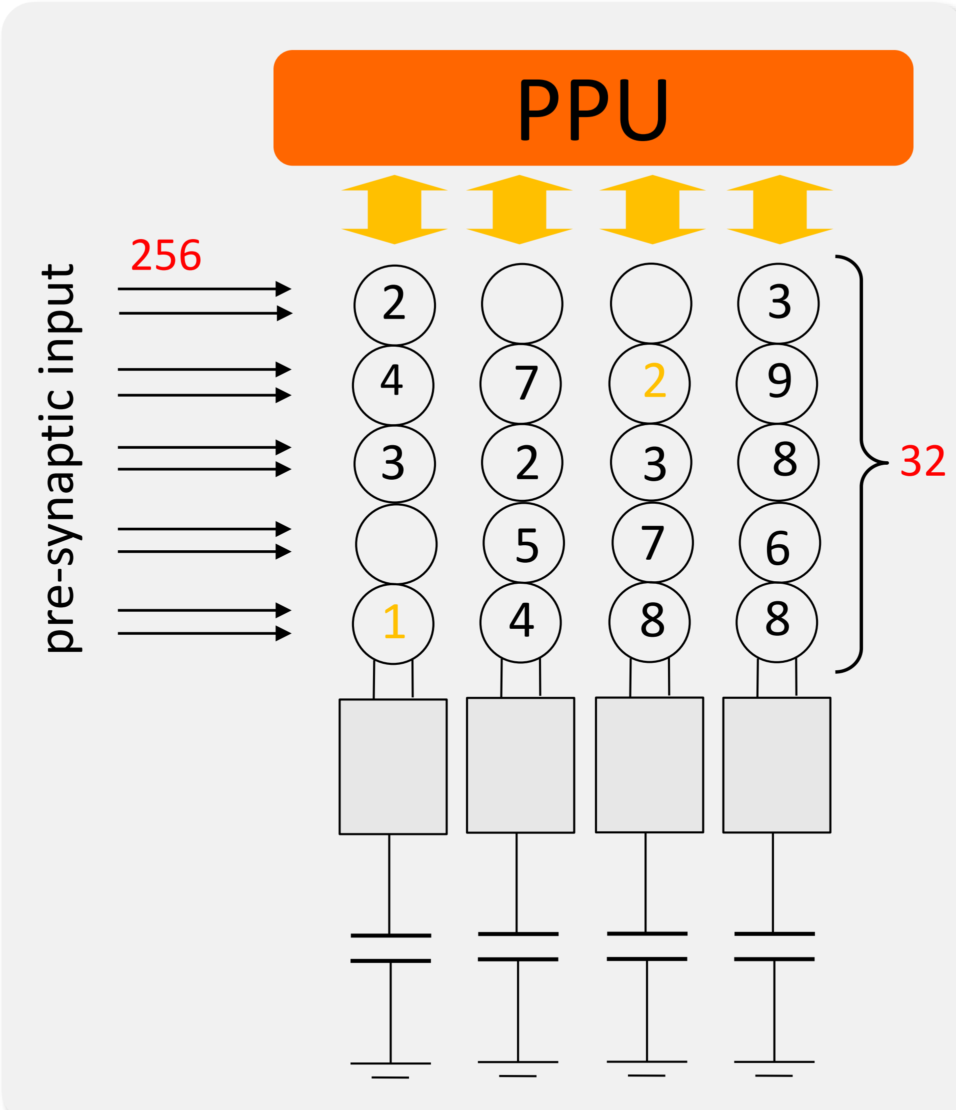
- assign random pre-synaptic neurons
- evaluate correlation
- keep the best

Structural Plasticity



- assign random pre-synaptic neurons
- evaluate correlation
- keep the best
- replace weakly correlating synapses constantly against random new ones

Experimental Example : Structural Plasticity



- 256 pre-synaptic inputs mapped to single dendrite with 32 active synapses
- plasticity rule combines structural, STDP and homeostatic terms:

if $\omega \geq \theta_{\text{rand}}$:

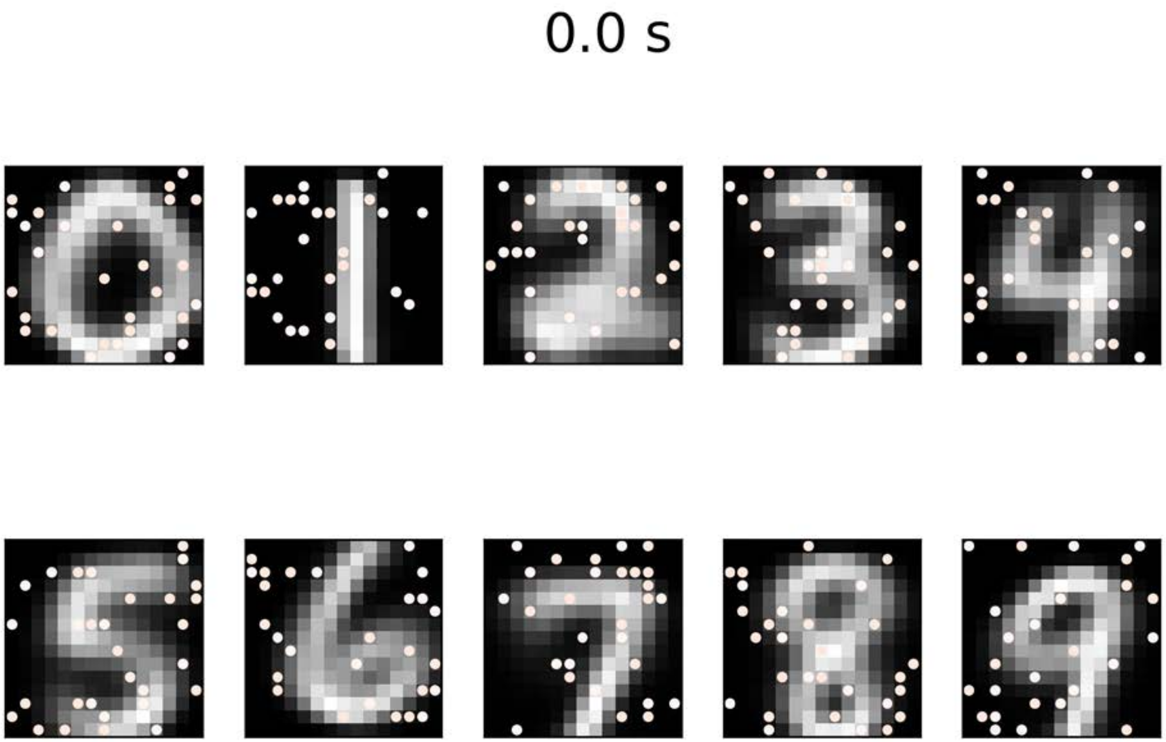
$$\omega' \leftarrow \omega$$
$$+\lambda_{\text{STDP}}(c_+ + c_-)$$
$$-\lambda_{\text{hom}}(\nu + \nu_{\text{target}})$$
$$a' \leftarrow a$$

else:

$$\omega' \leftarrow \omega_{\text{init}}$$
$$a' \leftarrow \text{rand}(0{,}8)$$

B. Cramer and S. Billaudelle, unpublished work, 2018

Supervised learning



- 256 pre-synaptic inputs mapped to single dendrite with 32 active synapses
- plasticity rule combines structural, STDP and homeostatic terms:

if $\omega \geq \theta_{\text{rand}}$:

$$\omega' \leftarrow \omega$$
$$+\lambda_{\text{STDP}}(c_+ + c_-)$$
$$-\lambda_{\text{hom}}(\nu + \nu_{\text{target}})$$
$$a' \leftarrow a$$

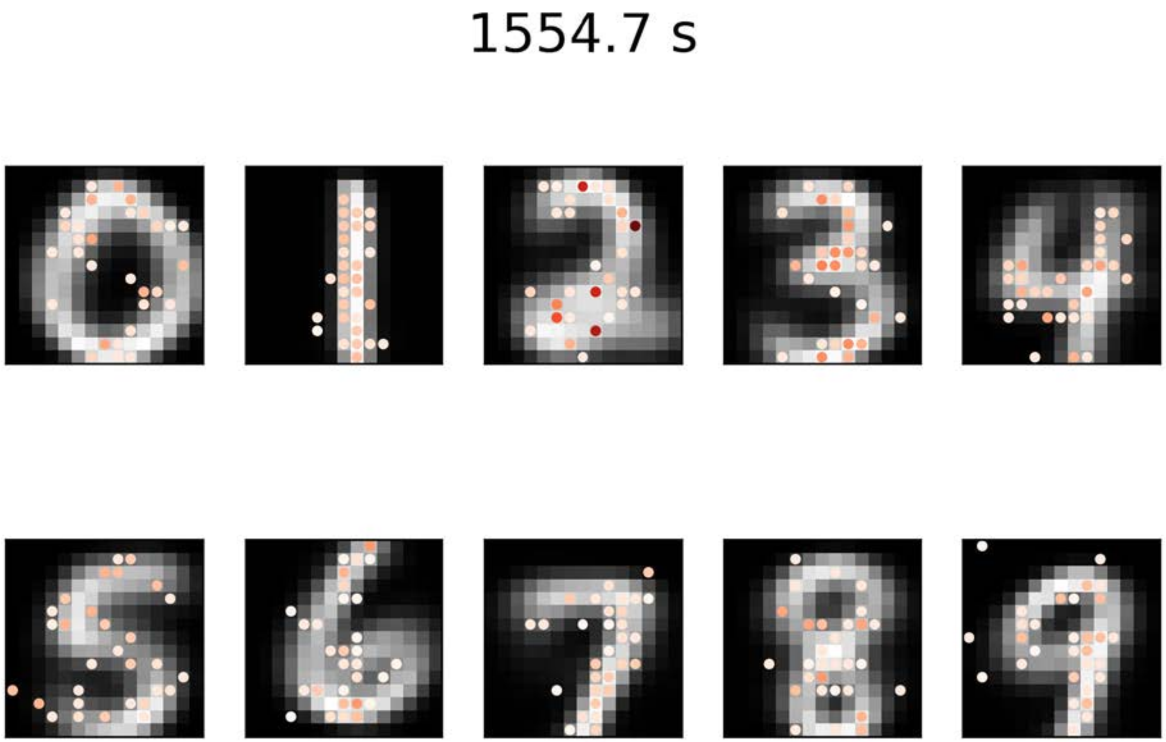
else:

$$\omega' \leftarrow \omega_{\text{init}}$$
$$a' \leftarrow \text{rand}(0,8)$$

B. Cramer and S. Billaudelle, unpublished work, 2018

- dots represent realized (active) synapses
- ten target groups (with three dendrites each) trained simultaneously
- 1.5 s wall time needed for emulation

Supervised learning



- dots represent realized (active) synapses
- ten target groups (with three dendrites each) trained simultaneously
- 1.5 s wall time needed for emulation

- 256 pre-synaptic inputs mapped to single dendrite with 32 active synapses
- plasticity rule combines structural, STDP and homeostatic terms:

if $\omega \geq \theta_{\text{rand}}$:

$$\omega' \leftarrow \omega + \lambda_{\text{STDP}}(c_+ + c_-) - \lambda_{\text{hom}}(\nu + \nu_{\text{target}})$$

$$a' \leftarrow a$$

else:

$$\omega' \leftarrow \omega_{\text{init}}$$

$$a' \leftarrow \text{rand}(0,8)$$

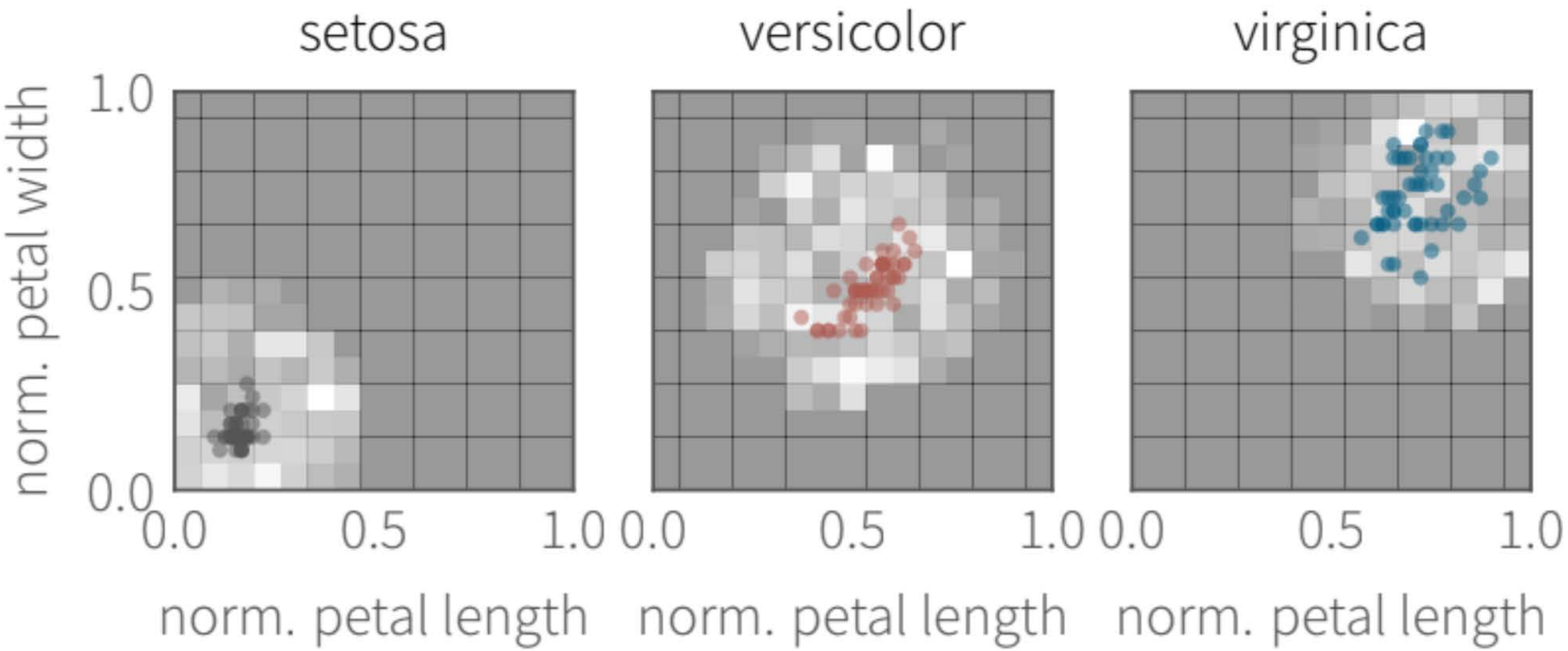
B. Cramer and S. Billaudelle, unpublished work, 2018

Formation of receptive fields with structural plasticity

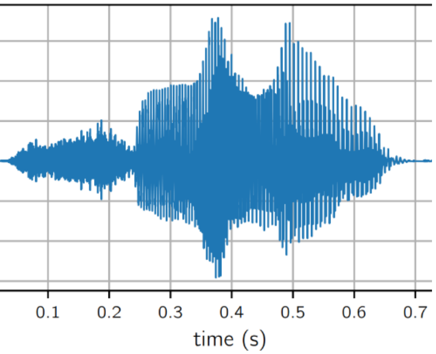
B. Cramer and S. Billaudelle, unpublished work, 2018

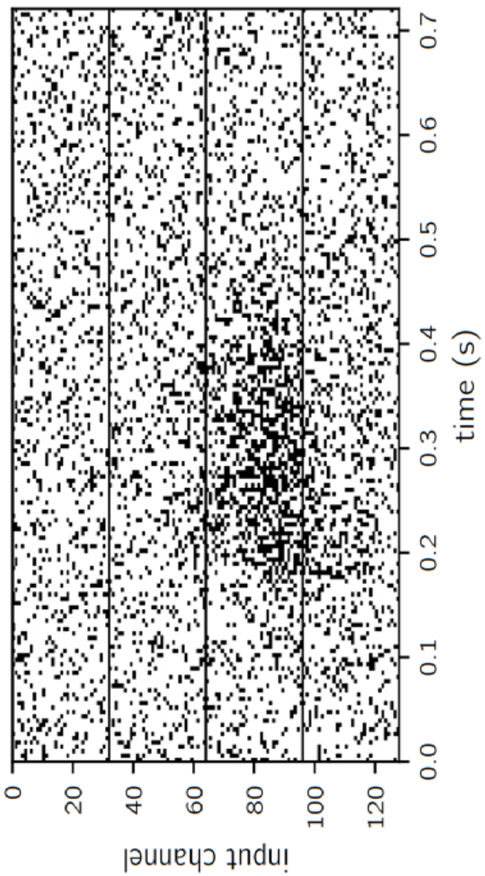
- Iris dataset
- Simple feed-forward network
- only a small fraction of all possible synapses realized
- Synapses are rewired to cover relevant receptor locations
- Self-organized development of receptive fields

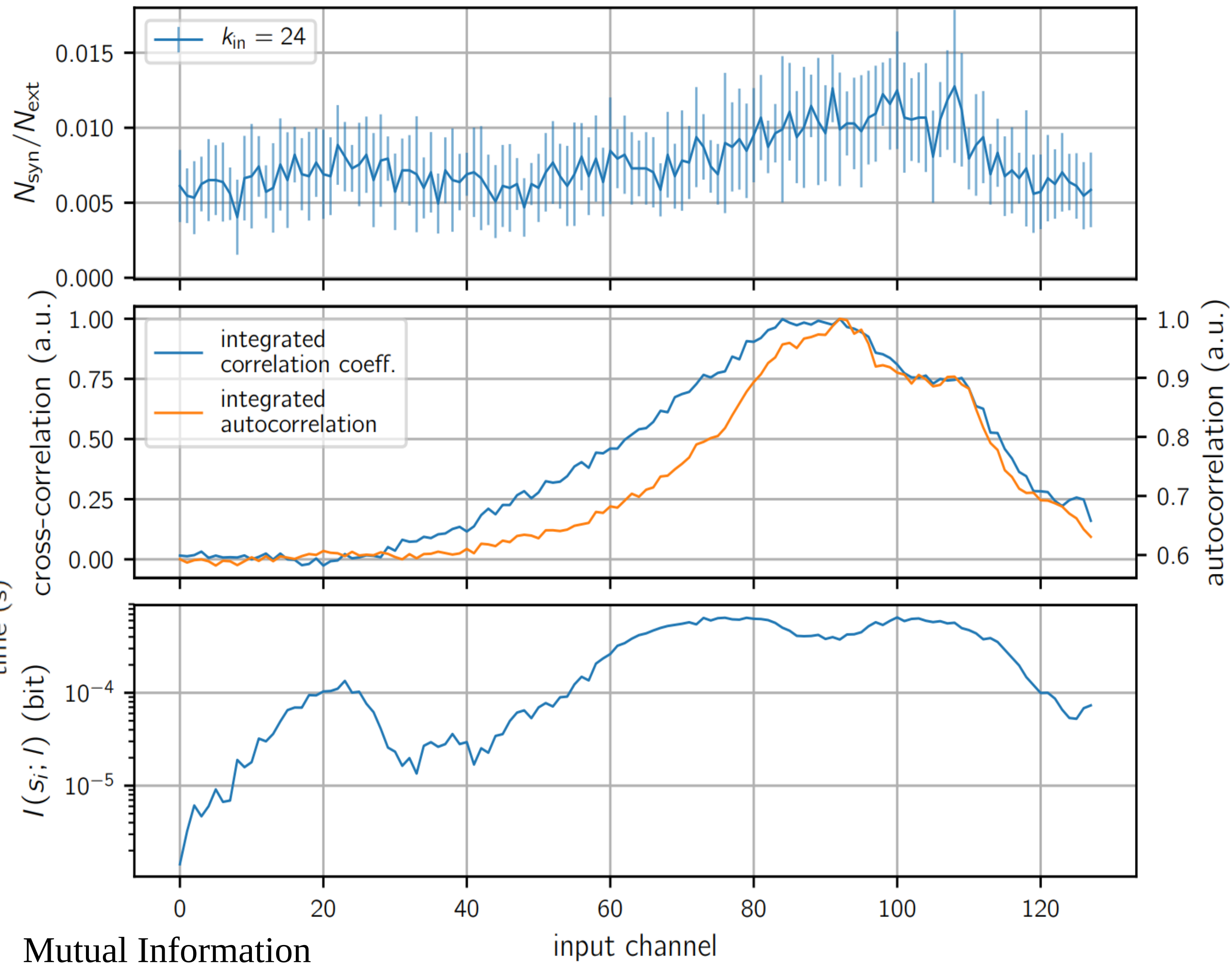
Two of four features shown:



Auditory stimulus: learning input channel distribution







Mutual Information

Markus Kreft, Bachelor Thesis, Heidelberg University, 2019

Conclusions and Outlook

BrainScaleS neuromorphic principles:

- physical model for fast, energy efficient neural network emulation of
  - structured neurons
  - nonlinear effects of dendrites
  - time-continuous emulation of different ion-channels
  - correlation measurement
- closely coupled to SIMD u
  - training
  - initialisation
  - configuration
  - debugging
  - calibration
- shared system-wide netwo
  - action potentials
  - memory access for neural routing CPUs
  - message passi
    inputs)

for many, many years . . .


Thank you!