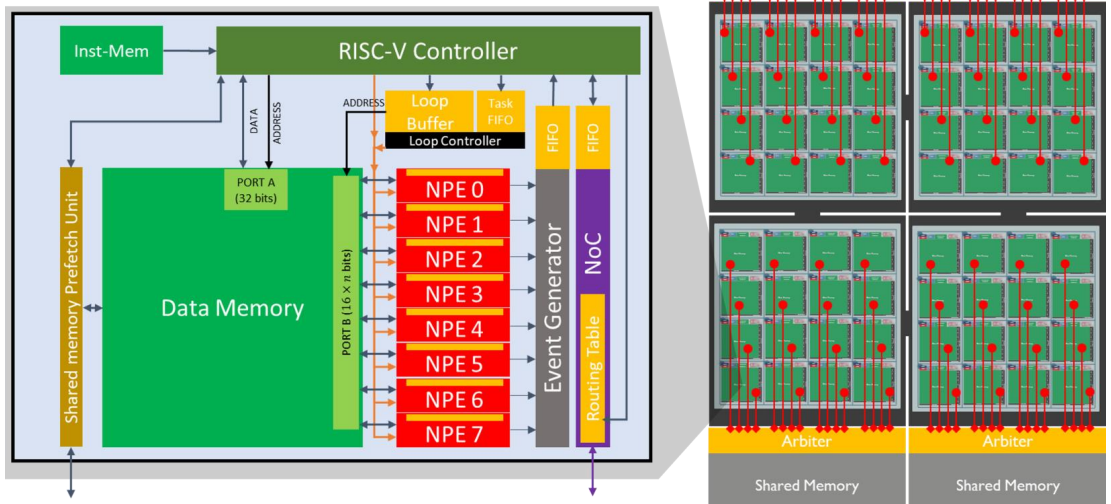
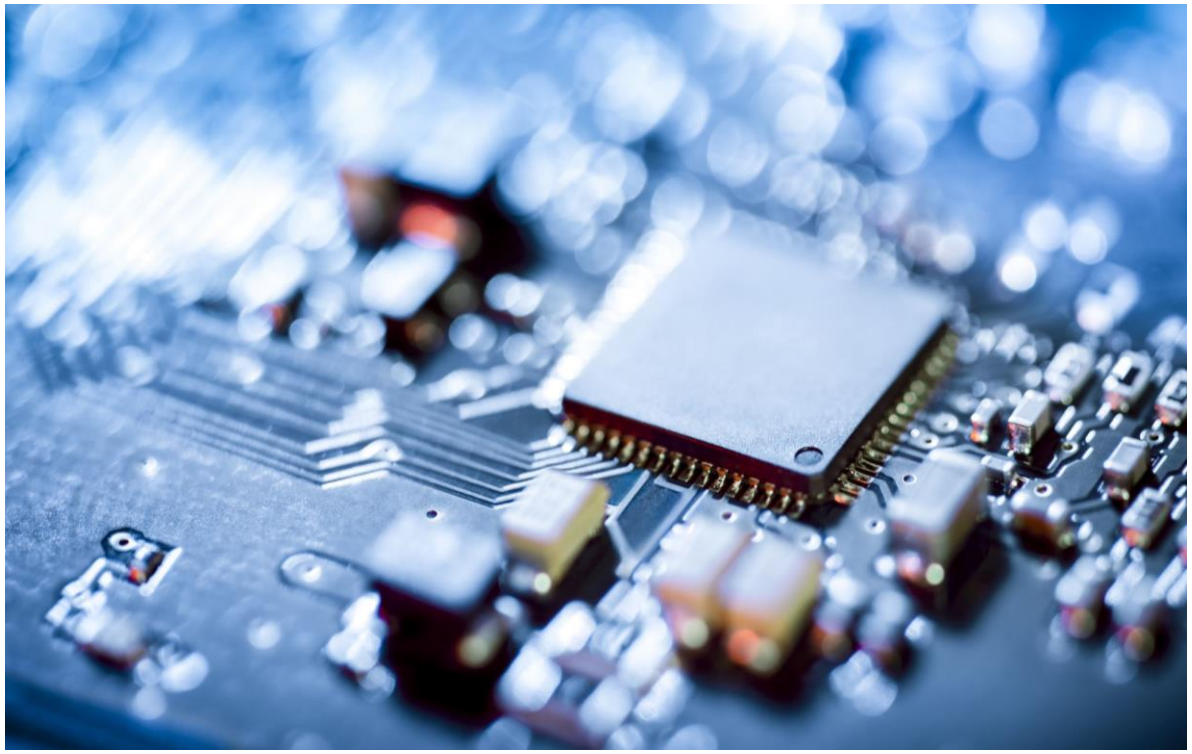




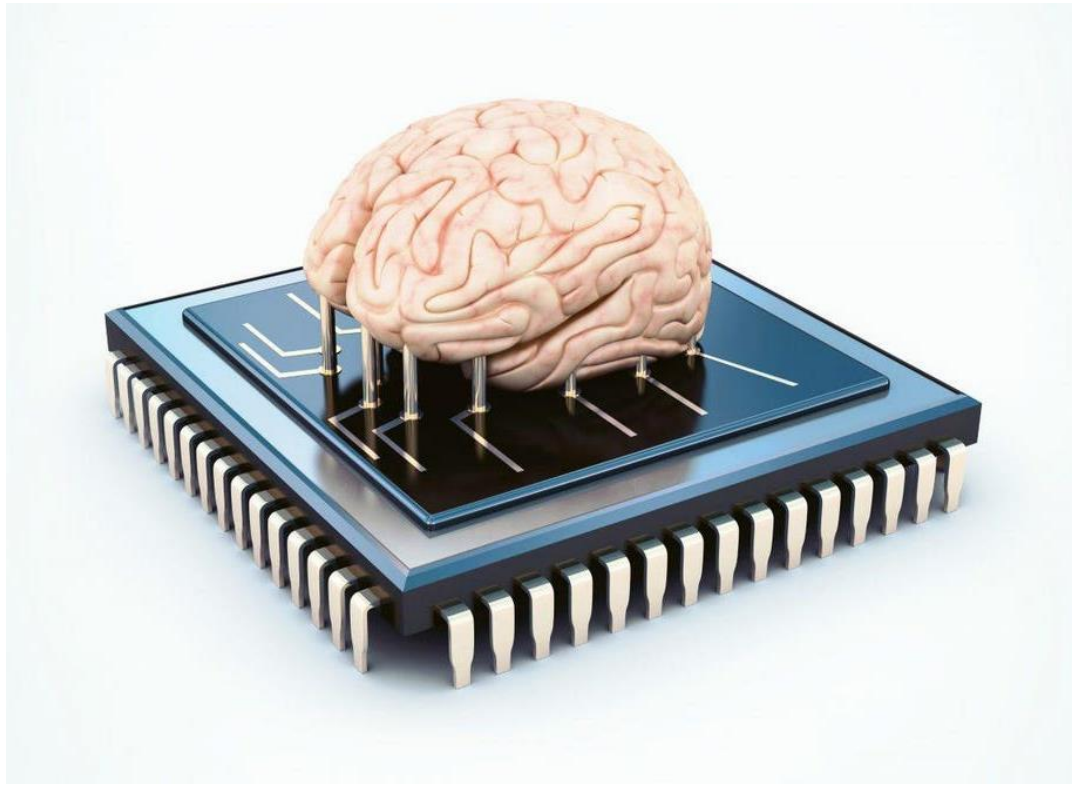
SENECA PROJECT (2020 → 2024)
RISC-V BASED NEUROMORPHIC PROCESSOR

AMIRREZA YOUSEFZADEH (AMIRREZA.YOUSEFZADEH@IMEC.NL)

DEC 2023

imec

SILICON BRAIN



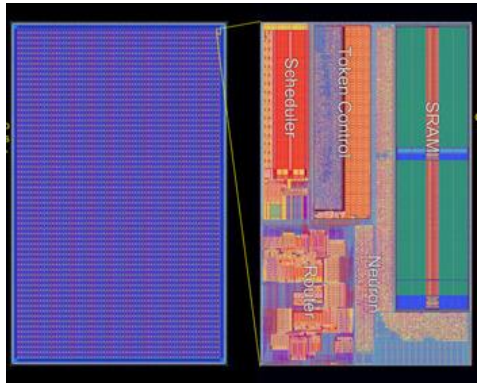
- **Low power**
  - Human brain consumes **10** to **20 Wat**
  - Sensor processing, Sensor fusion, Generative output, Memory and Learning
- **Low latency**
  - While ions are 1M times slower than electrons

Some representative features of the brain

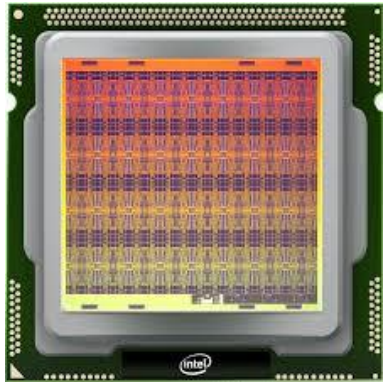
- Billions of interconnected tiny processing elements
- Highly sparse processing in a vast 3D area (low power density)
- Co-optimized with the algorithm while Turin complete

SOME DIGITAL NEUROMORPHIC PROCESSORS

IBM TrueNorth 2014/2023

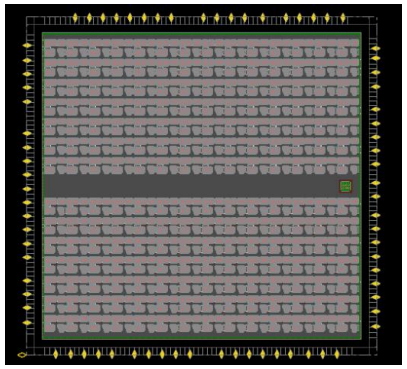


Intel Loihi 2018/2022

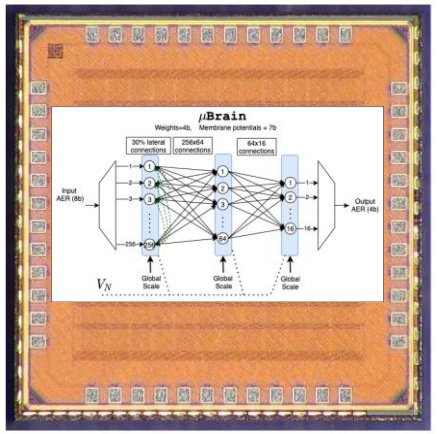


BrainChip AKIDA 2021

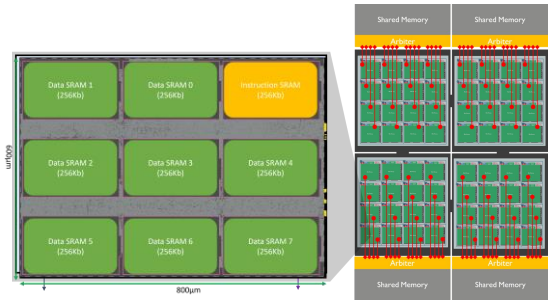
GML NeuronFlow 2020/2022



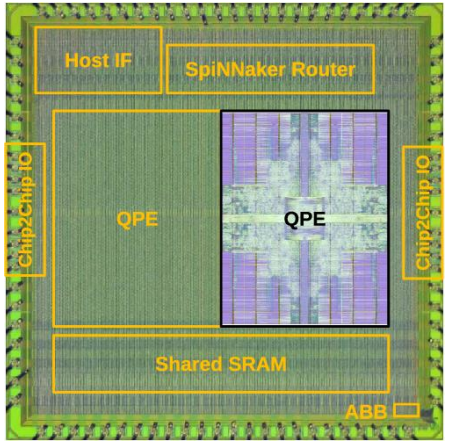
Imec µBrain 2020



Imec SENECA 2022

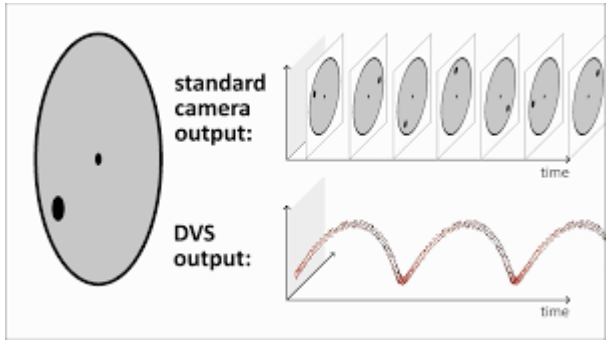


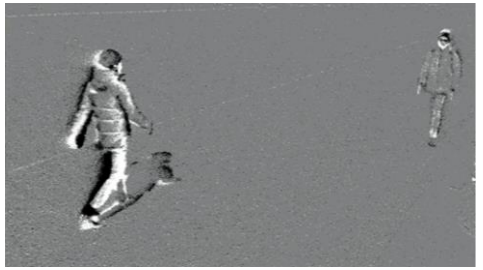
SpiNNaker 2013/2020



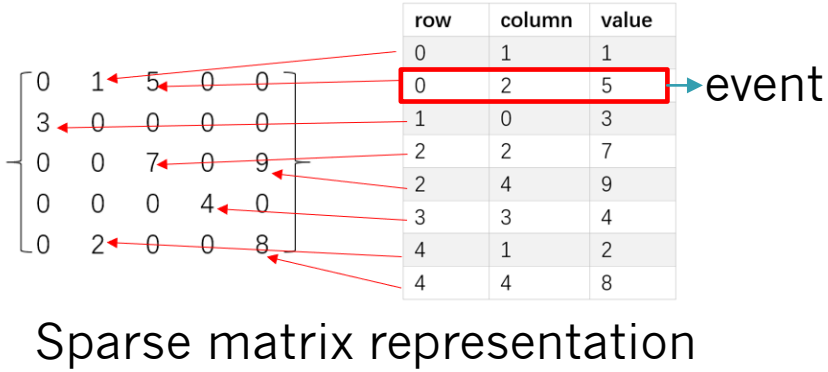
NEUROMORPHIC PROCESSORS

- **Low power**
  - Exploiting unstructured sparsity
- **Low latency**
  - Exploiting high temporal precision





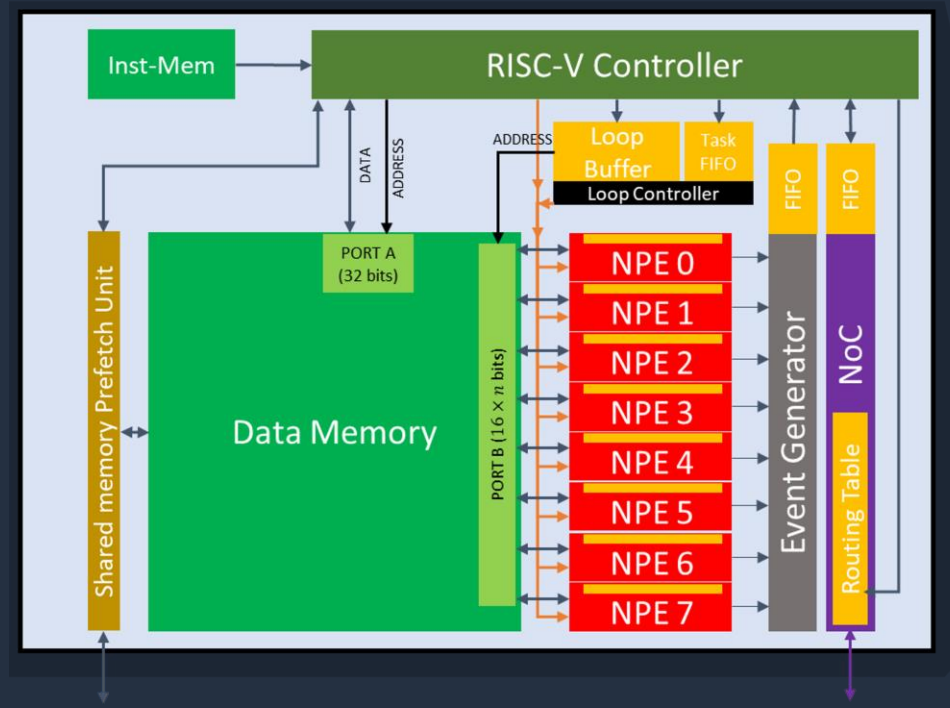
Unstructured
Sparse matrix

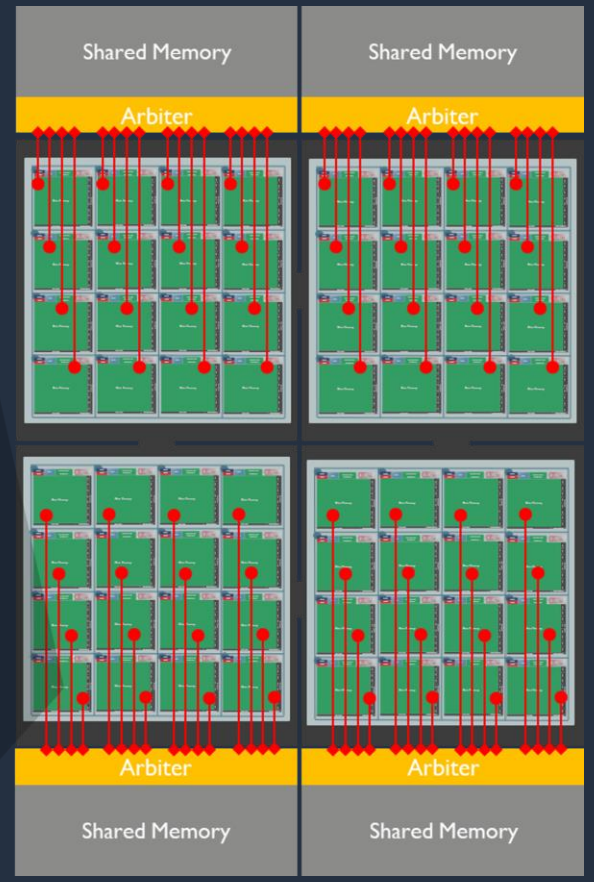


Sparse matrix representation

EVOLUTION OF SENECA

- Array of RISC-V
- NoC
- Neuron Processing Elements
- Loop Controller
- Event Generator
- Spike Grouping
- Depth-First Processing
- Task FIFOS (Asynchronization)
- Shared Memory
- Synaptic Delay accelerator
- Open-source SENECA Lite





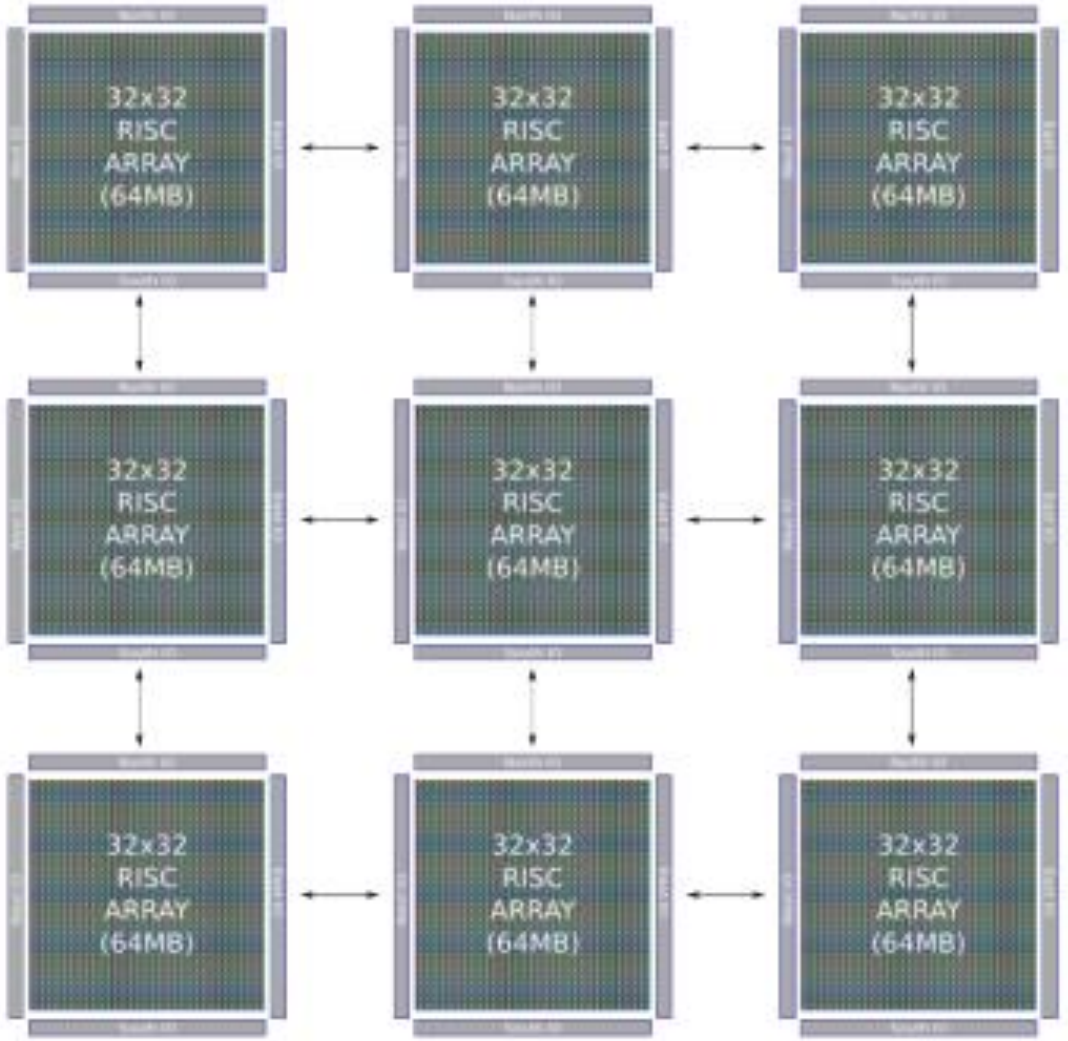
ARRAY OF TINY PROCESSORS

The idea was to connect many tiny processors trough an extendable network on chip:

- Distributed memory-processing
- Event driven Data flow architecture

Why choosing RISC-V?

- Fast development
- Flexible enough to run all types of SNN models
- Excellent open-source repositories
- Low area overhead



FIRST STEP

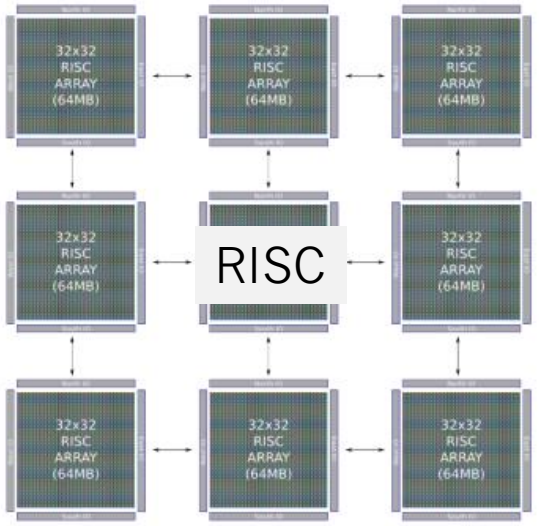
Studying similar architectures

- ARM-based: SpiNNaker
- RISC-V based: Epiphany, Esperanto, many others
- Custom: Loihi, TrueNorth, NeuronFlow, AKIDA, many others

Performance Comparison of Time-Step-Driven versus Event-Driven Neural State Update Approaches in SpiNNaker

Publisher: IEEE Cite This PDF

Amirreza Yousefzadeh ; Mikel Soto ; Teresa Serrano-Gotarredona… All Aut



Benchmarking the Epiphany Processor as a Reference Neuromorphic Architecture

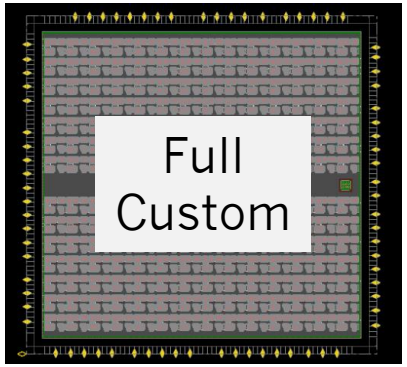
Maarten Molendijk[1,2], Kanishkan Vadivel[2], Federico Corradi[2,1], Gert-Jan van Schaik[1], Amirreza Yousefzadeh[1], and Henk Corporaal[2]

[1]imec, Netherlands
[2]Technical University of Eindhoven, Netherlands

NeuronFlow: A Hybrid Neuromorphic – Dataflow Processor Architecture for AI Workloads

Publisher: IEEE Cite This PDF

O. Moreira ; A. Yousefzadeh ; F. Chersi ; A. Kapoor ; R.-J. Zwartenkot ; P. Qiao ; G. Cinserin.

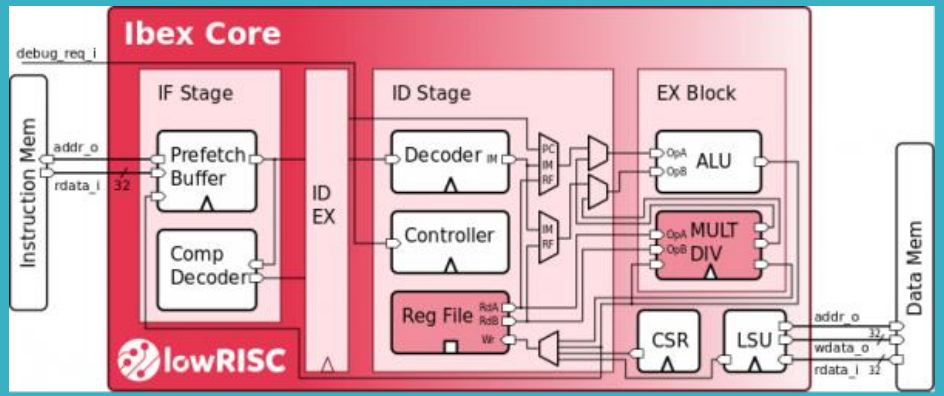


BASIC BUILDING BLOCKS

Tiny RISC-V processor

Choosing one of the smallest RV available

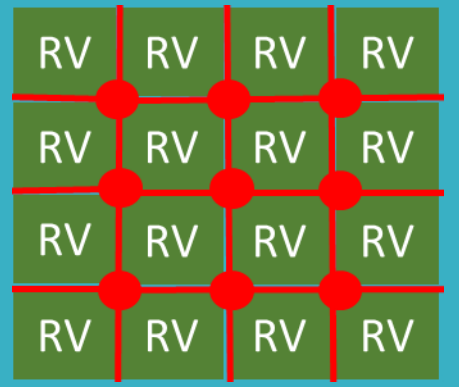
(0.01mm² @22nm)



Network on Chip

Simplest NoC available

small footprint, low performance, low area consumption

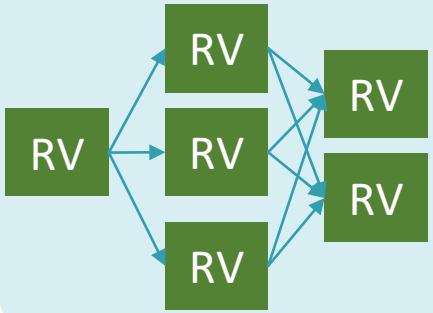


NOC EVOLUTION

Simplest form of NOC

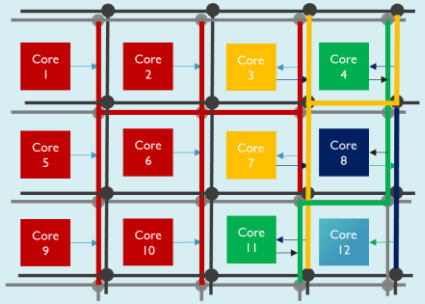
No evidence that this is a bad NoC, however
- Multicasting perceive as important element in other neuromorphic chips
- We had enough time to spend for this

Hardwire connections



- **Fast development**
- **Enough for FPGA prototype**
- **Allows multicasting**
- **Only a temporary solution**
- **Inflexible for PnR exploration**

Circuit switched NOC (Segmented BUS)

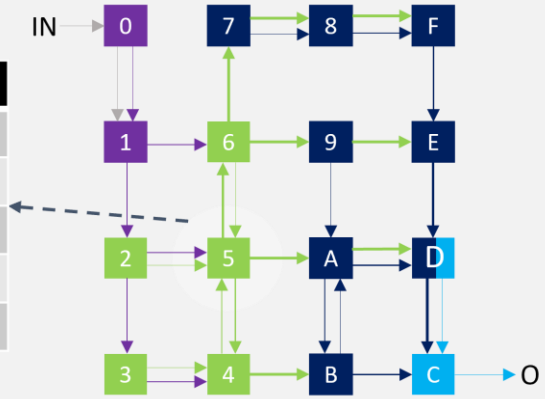


- **Allows for multicasting**
- **Asynchronous**
- **Low power (not pipelined)**
- **FPGA prototyping challenges**
- **Not flexible enough**

Packet Switched Multicasting NOC

Routing table

Input	Label	Outputs
West	1	C
West	2	N+S+E
North	2	S+E
South	2	N+E
Core	2	N+S+E



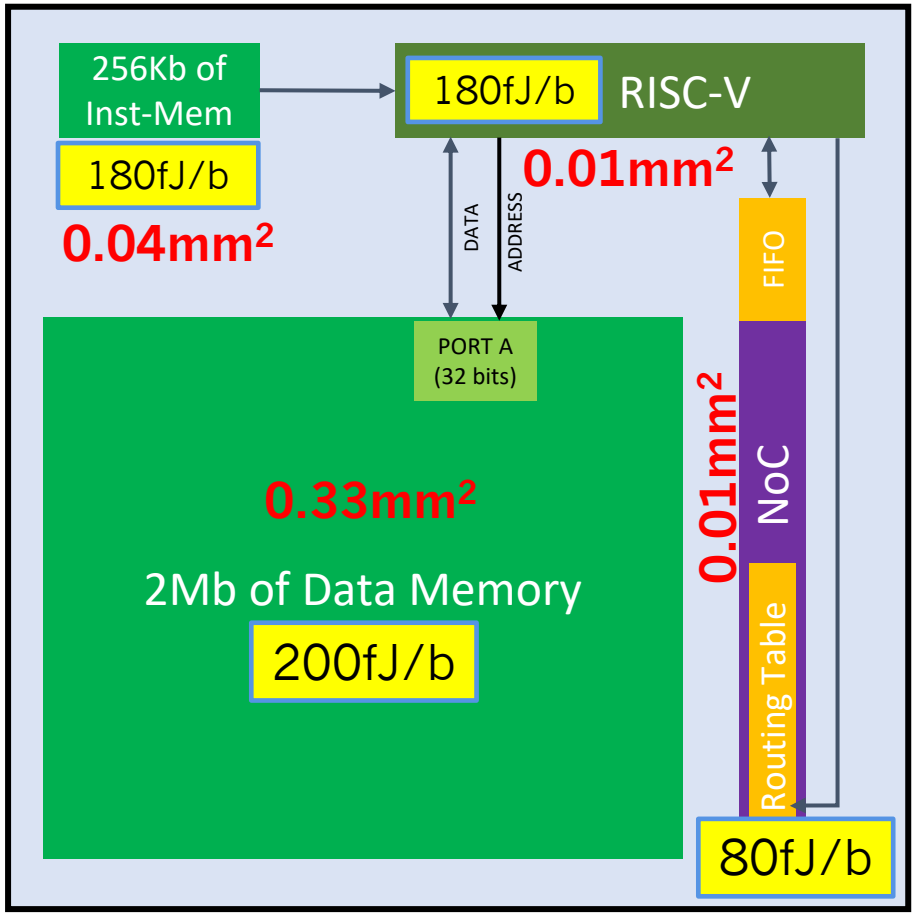
- **Low area footprint**
- **Multi-casting (source-based addressing)**
- **Multi-flit routing**
- **Less flexible than X,Y routing**
- **Challenges yet to be seen**
- **Yet there is no concrete evidence that this NoC can improve performance**
  - **Simulation showed 50% less data movement because of multicasting**

FIRST VERSION READY

Area/Power benchmarking:

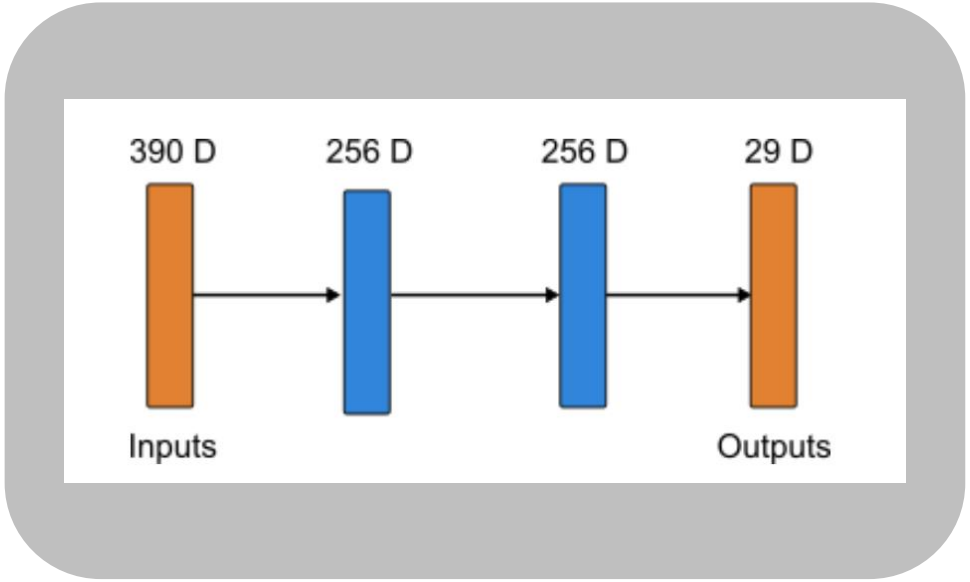
- Cadence Tools
- 22nm FDSOI technology node

Processor	Memory (Mb) per core	Area(mm^2) per core	Number of neurons per core	Technology
IMEC SENECA	2.3	0.40	Flex	22nm
Intel Loihi 1	2.0	0.41	1024	14nm
Intel Loihi 2	1.5	0.21	Flex	Intel4
SpiNNaker2	1.0	1.09	Flex	28nm
IMEC µBrain	0.15	1.42	336	40nm
GML NeuronFlow	0.12	0.1	1024	28nm
SpiNNaker1	0.12	5.6	Flex	130nm
IBM TrueNorth	0.1	0.1	256	28nm



One SENECA Core **0.4mm²**

BENCHMARKING



Processor	Inference Time (µS)	Energy (µJ)
Loihi 1	3378	372
SpiNNaker2	1000	7.1
SENECA	**7000**	**34**

No worries, there is still money and time ☺

NEURON PROCESSING ELEMENTS

SENeCA: Scalable Energy-efficient Neuromorphic Computer Architecture

Publisher: IEEE Cite This PDF

Amirreza Yousefzadeh ; Gert-Jan van Schaik ; Mohammad Tahghighi ; Paul Detterer ; St... A

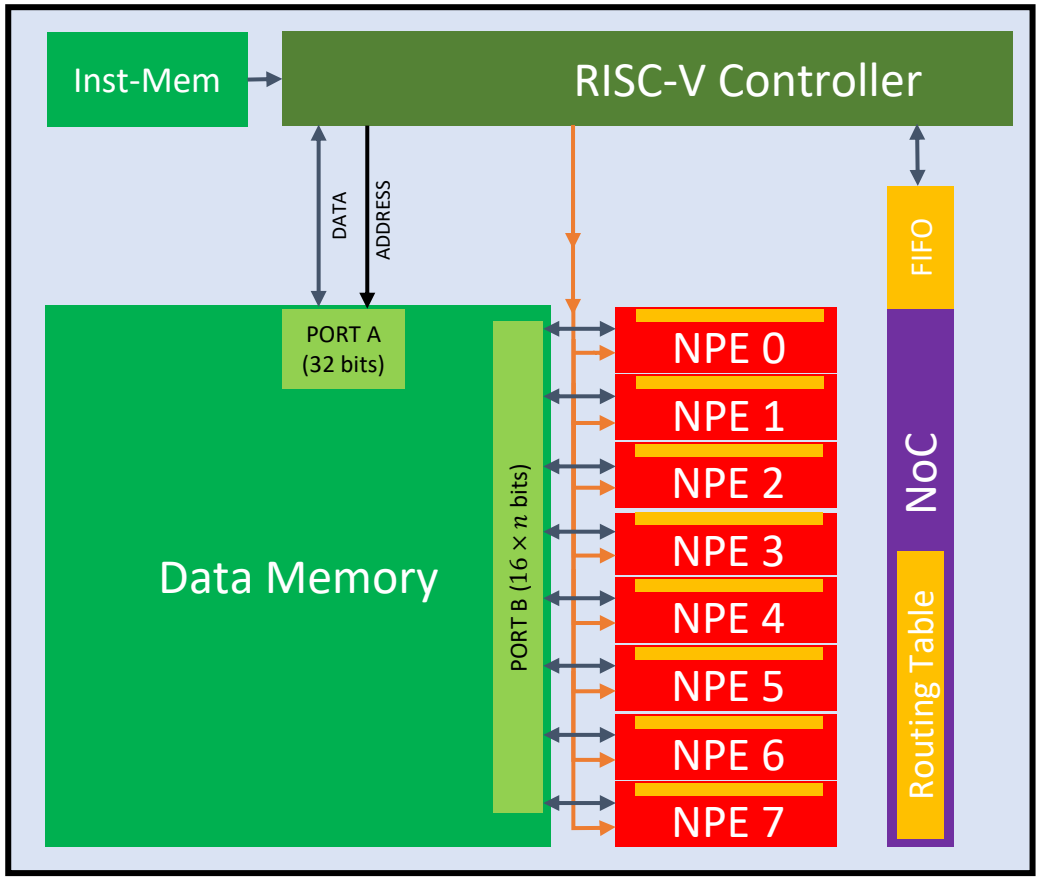
Improves

- **Performance**
  - Updating 8 neurons in parallel
  - Complex neuron operations in one cycle
- **Energy consumption**
  - BF16 instead of INT32
  - Single instruction overhead (SIMD)

Challenge:

Writing assembly micro-codes for NPES

```
MLD(R0, ADD1, 1) //load weight w_ij
MLD(R1, ADD2, 0) //load state v_i
ADD(R1, R0, R1) //v_i = v_i + w_ij
MST(ADD2, R1, 1) //store R1 in v_i
```



NEURON PROCESSING ELEMENTS

SENeCA: Scalable Energy-efficient Neuromorphic Computer Architecture

Publisher: IEEE Cite This PDF

Amirreza Yousefzadeh ; Gert-Jan van Schaik ; Mohammad Tahghighi ; Paul Detterer ; St... A

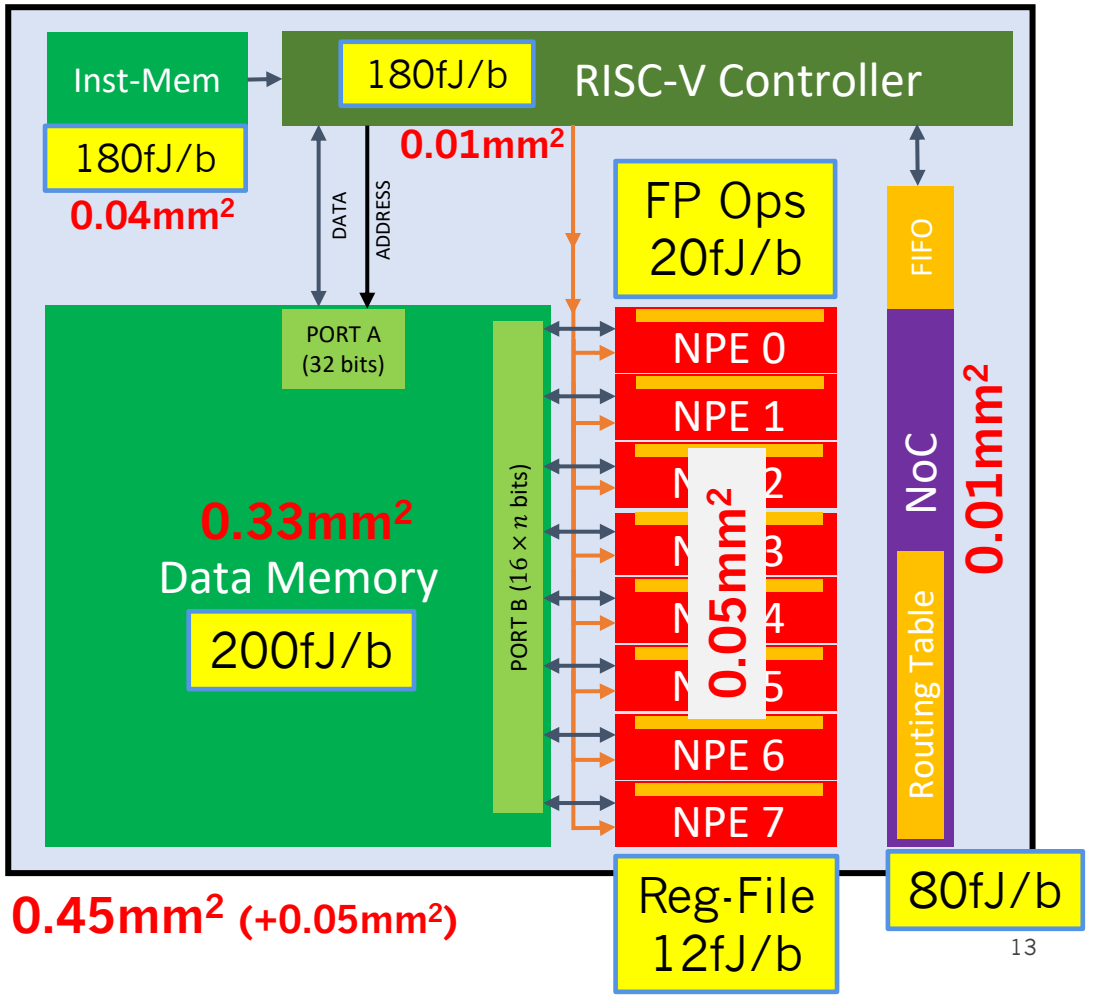
Improves

- **Performance**
  - Updating 8 neurons in parallel
  - Complex neuron operations in one cycle
- **Energy consumption**
  - BF16 instead of INT32
  - Single instruction overhead (SIMD)

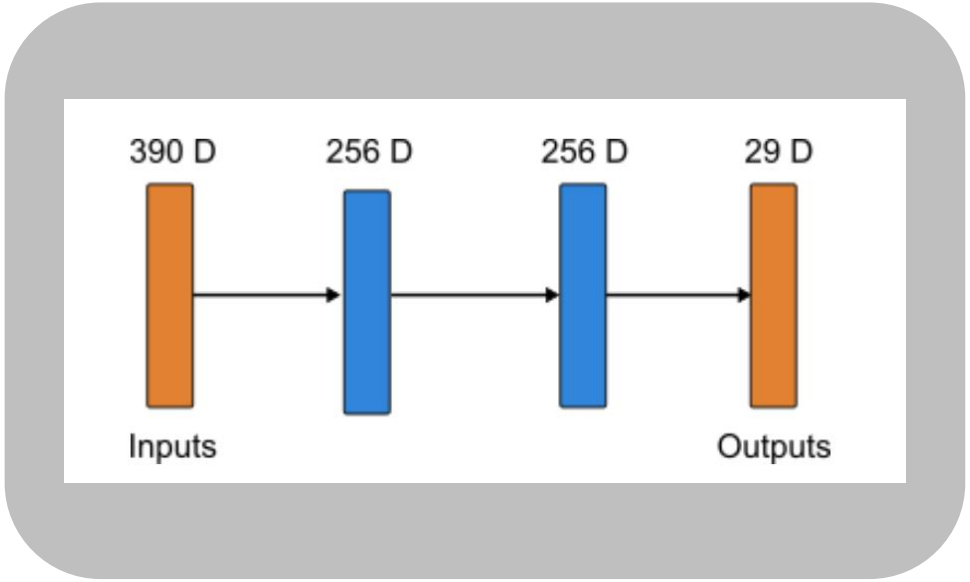
Challenge:

Writing assembly micro-codes for NPES

```
MLD(R0, ADD1, 1) //load weight w_ij
MLD(R1, ADD2, 0) //load state v_i
ADD(R1, R0, R1) //v_i = v_i + w_ij
MST(ADD2, R1, 1) //store R1 in v_i
```



BENCHMARKING

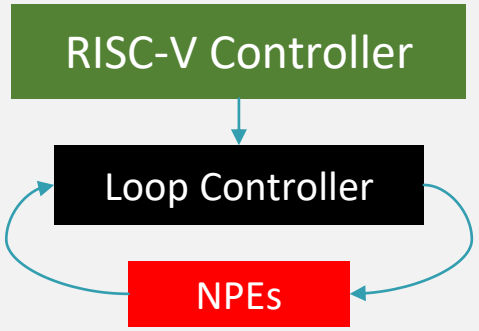


Processor	Inference Time (µS)	Energy (µJ)
Loihi 1	3378	372
SpiNNaker2	1000	7.1
SENECA1	7000	34
SENECA2	**1100**	**7**

Experiment	Inference Time (µS)	RISC-V Energy (µJ)	NPEs Energy (µJ)	Dmem Energy (µJ)	Total Energy (µJ)
RISC-V only	7000	30	0	2	34
Using NPEs	1100	3	2	1.5	7

LOOP CONTROLLER

Hierarchical controlling system

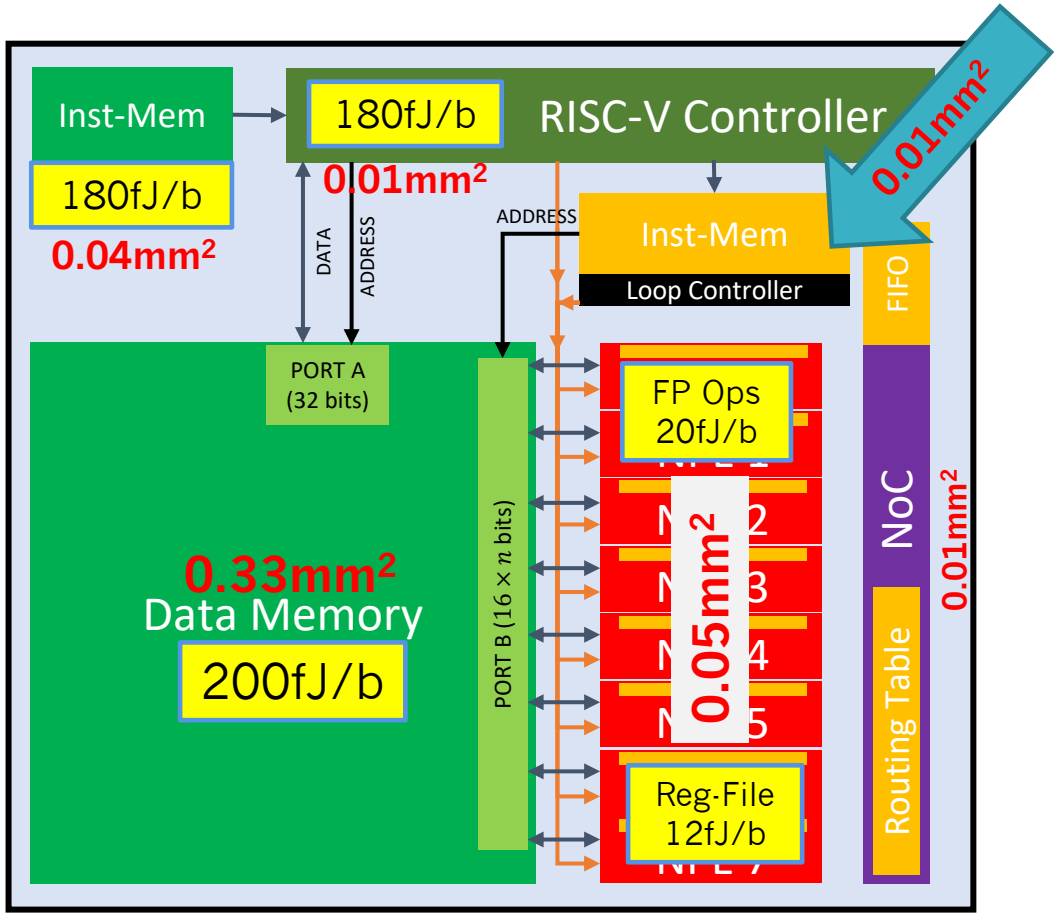


Highly Flexible controller

Custom controller

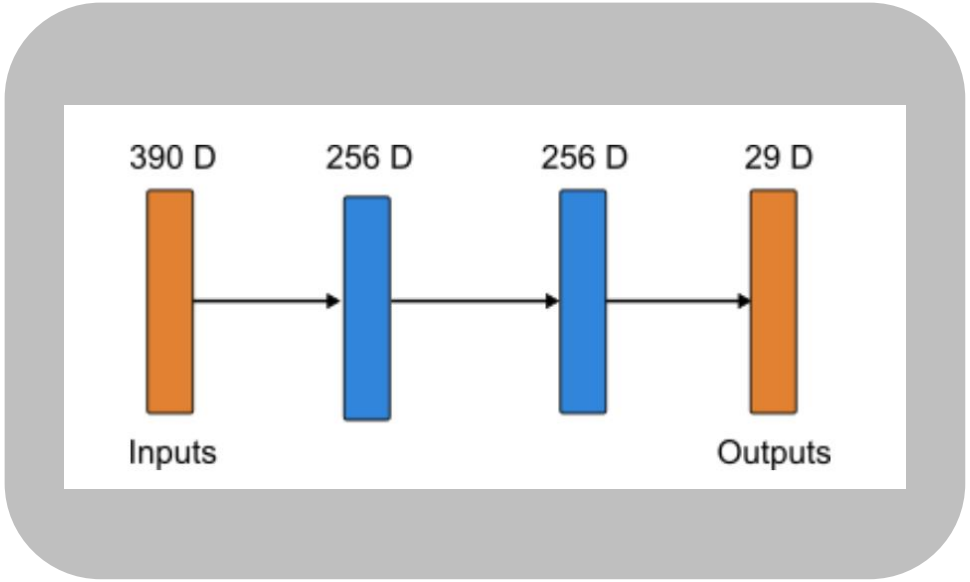
10x more power efficient

- Instructions to NPEs
- Memory Address calculation
- Loop index calculation



0.46mm² (+0.01mm² for loop controller and its memory)

BENCHMARKING



Processor	Inference Time (µS)	Energy (µJ)
Loihi 1	3378	372
SpiNNaker2	1000	7.1
SENECA1	7000	34
SENECA2	1100	7
SENECA3	**550**	**3**

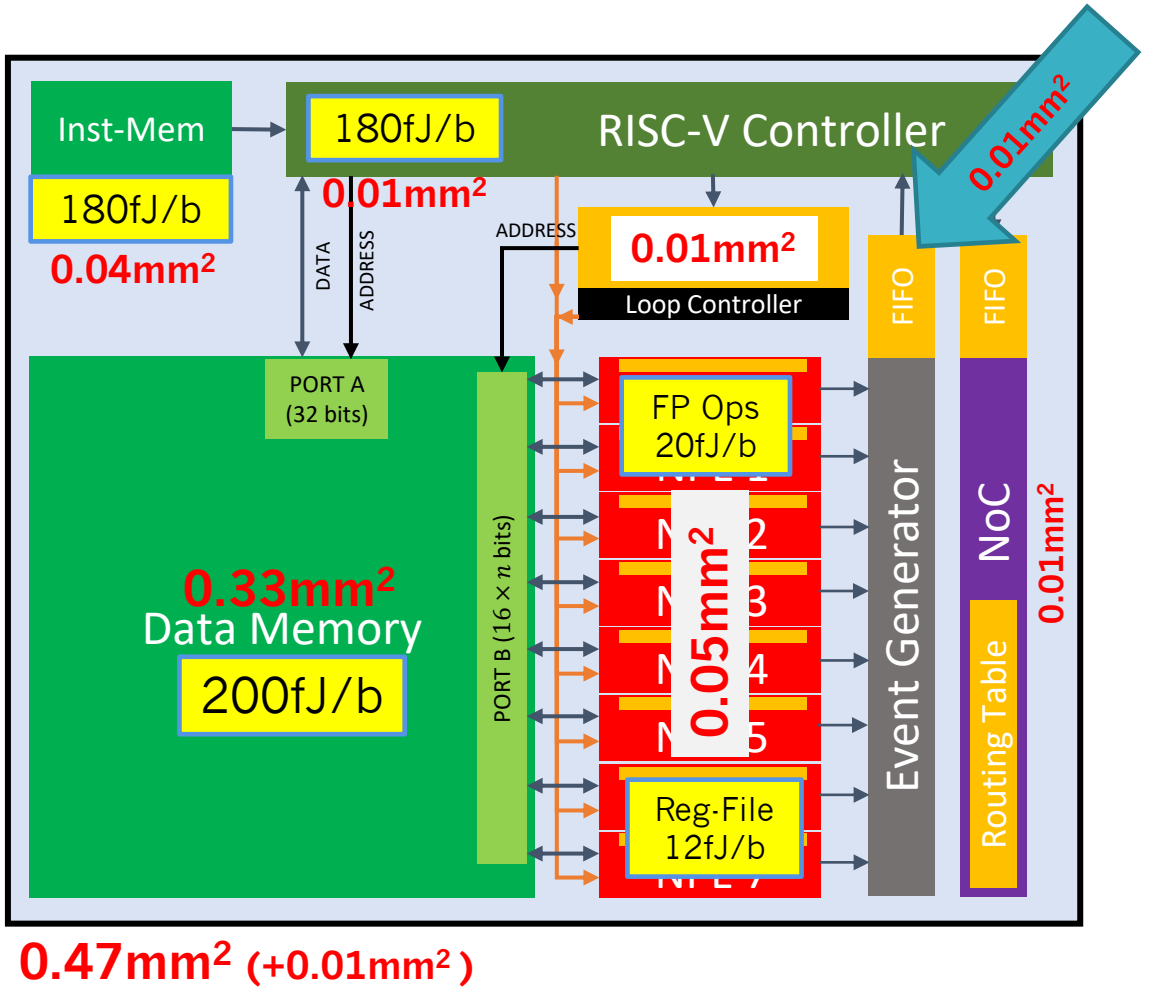
Experiment	Inference Time (µS)	RISC-V Energy (µJ)	NPEs Energy (µJ)	Dmem Energy (µJ)	Total Energy (µJ)
RISC-V only	7000	30	0	2	34
Using NPEs	1100	3	2	1.5	7
Loop Contrl	550	0.2	1.3	1.2	3

EVENT GENERATOR

Accelerating Spike Packetization:

- Threshold comparison
- Address calculation

Processor	Inference Time (µS)	Energy (µJ)
Loihi 1	3378	372
SpiNNaker2	1000	7.1
SENECA1	7000	34
SENECA2	1100	7
SENECA3	550	3
SENECA4	**400**	**2.1**

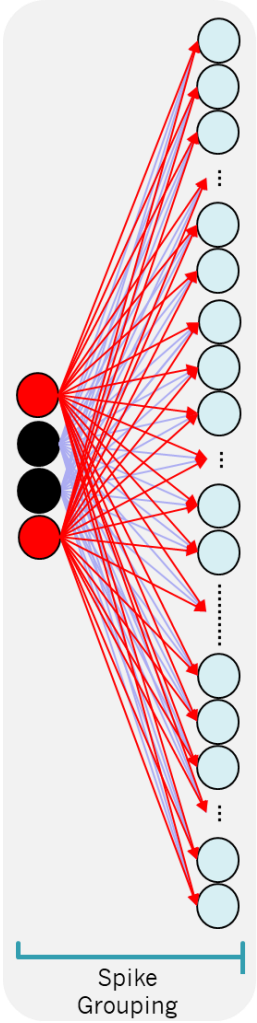


SW TECH: SPIKE GROUPING

A Software technique similar to Batch processing

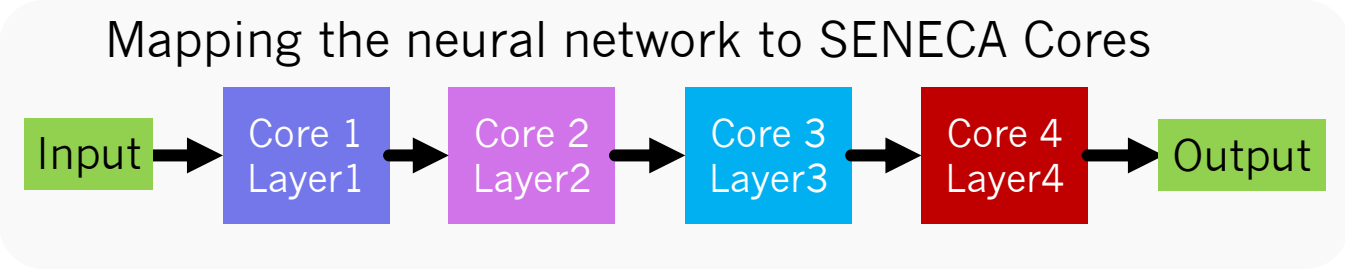
→ Re-using the parameters read from SRAM

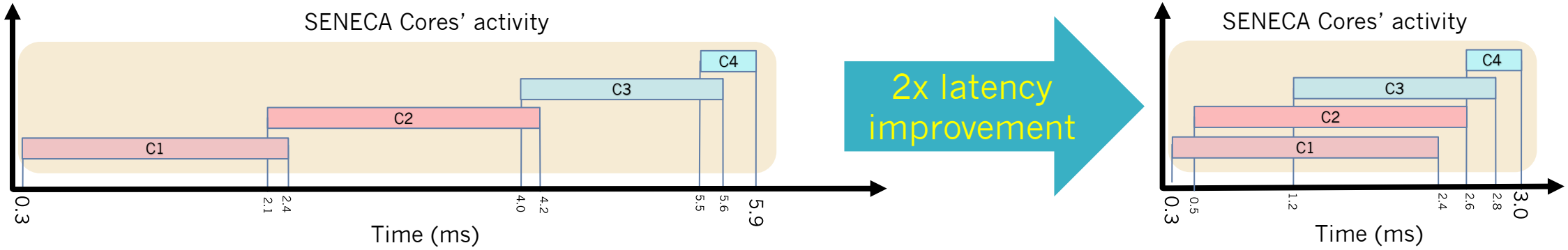
Processor	Inference Time (µS)	Energy (µJ)
Loihi 1	3378	372
SpiNNaker2	1000	7.1
SENECA1	7000	34
SENECA2	1100	7
SENECA3	550	3
SENECA4	400	2.1
SENECA5	**200**	**1.2**



SW TECH: DEPTH-FIRST LAYER FUSION

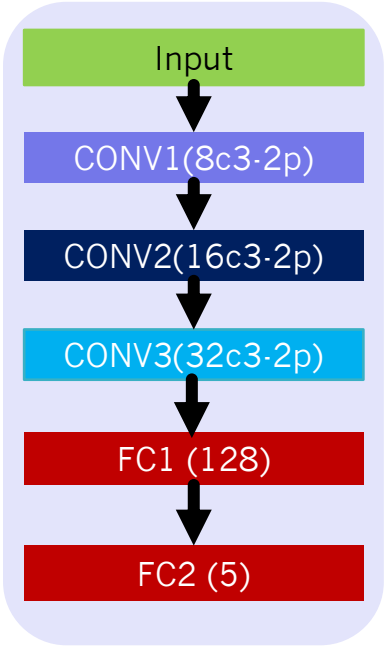
- **Special mapping increases latency** (layer-wise processing)
- **Depth First layer Fusion**:
  - Only in CNN
  - Reduction of Latency / memory usage





BENCHMARKING

Convolutional Neural Network

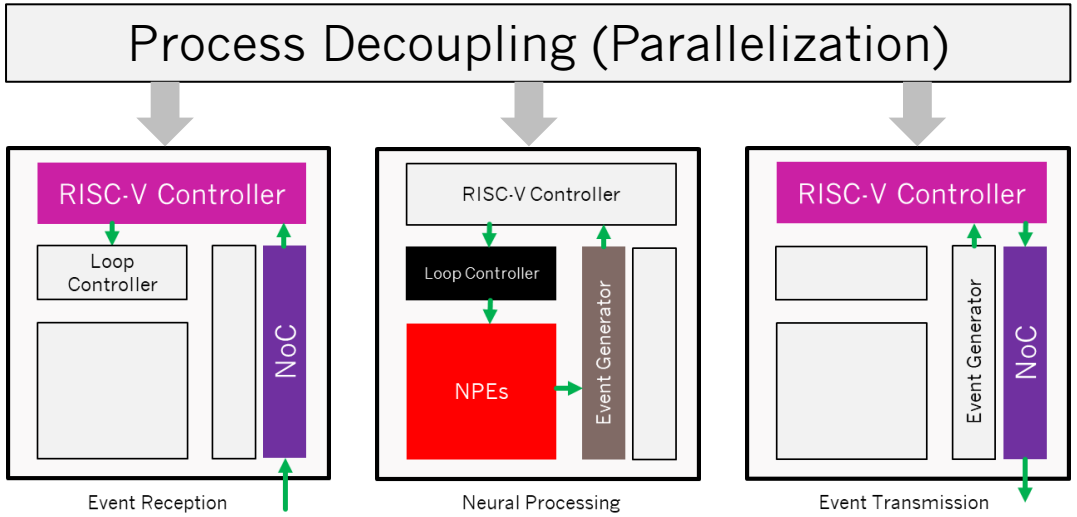


Processor	Inference Time (mS)	Energy (µJ)	Area* (mm^2)
Loihi 1	6.6	660	5.74
TrueNorth	4	108	192
MAX78000	8.3	215	NA
SENECA	**1.1**	**12**	**1.88**

**Area = Number of core used x Area of a core*
Task : MNIST with 99.4% accuracy

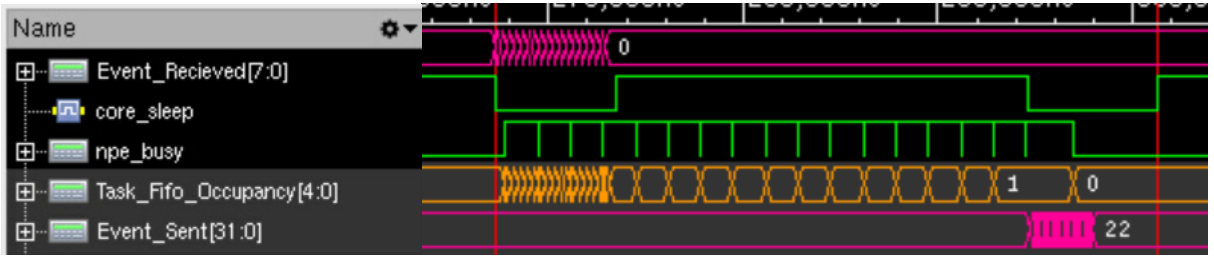
TASK FIFOS

- **RV waiting for Loop controller to finish its task**
- **Task FIFOs allows event reception in parallel to neural processing**



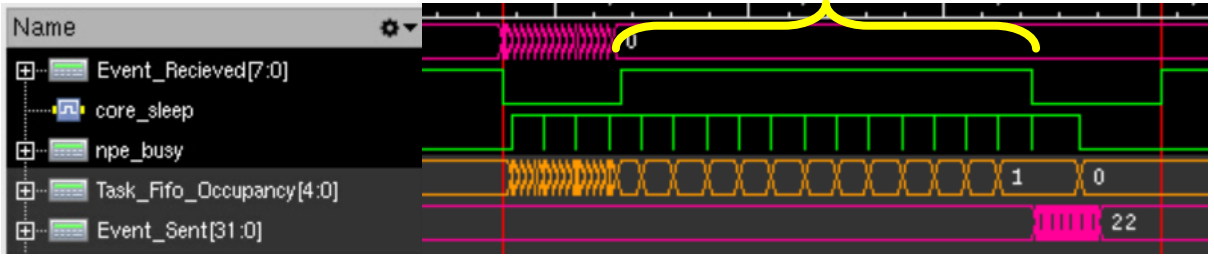
- When RV is slower: improving the performance
- When RV is faster: improving the energy consumption

227nJ

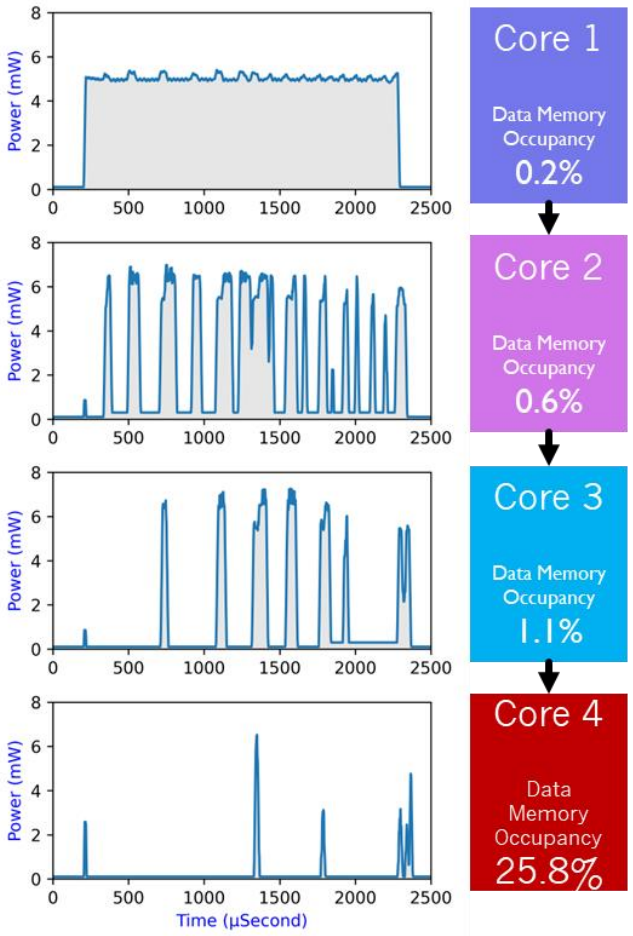


RV core sleeping

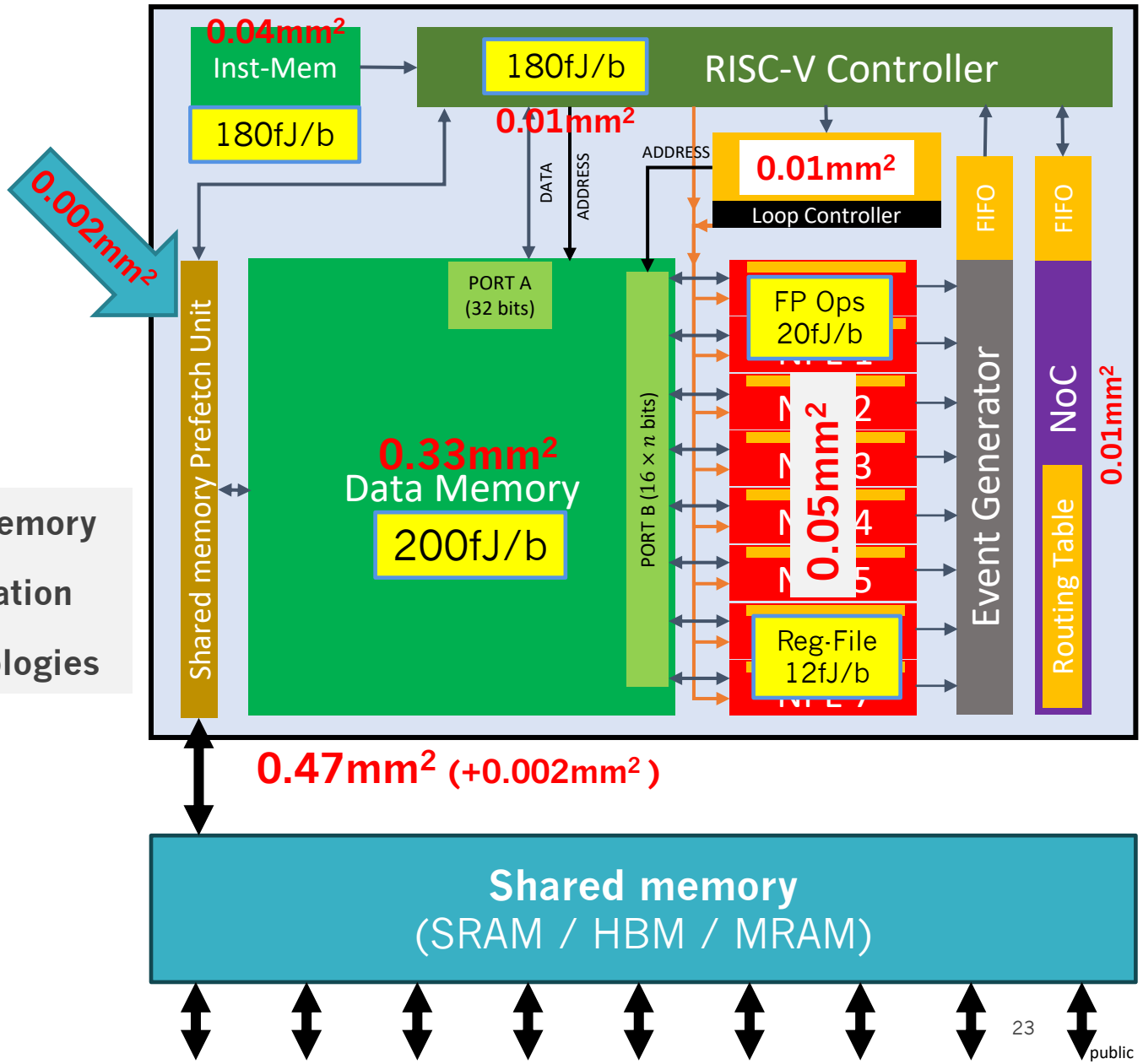
133nJ



SHARED MEMORY



- Extending the core memory
- On the go Reconfiguration
- Other memory technologies

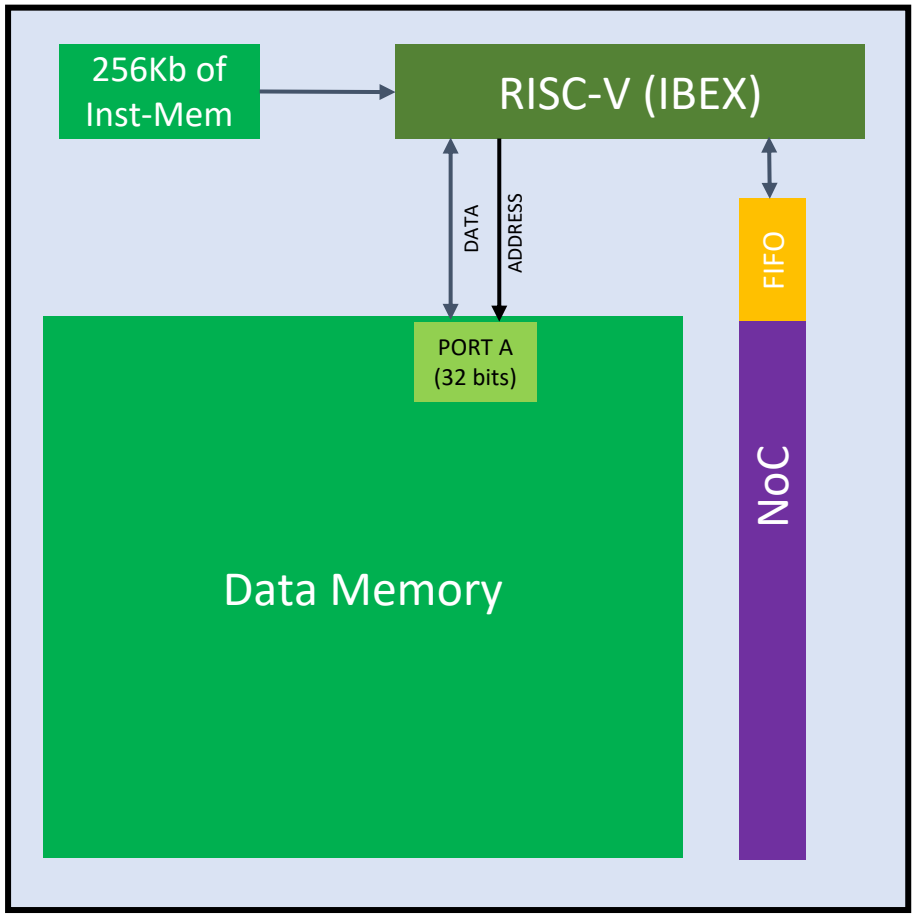


SENECA LITE

Open-source

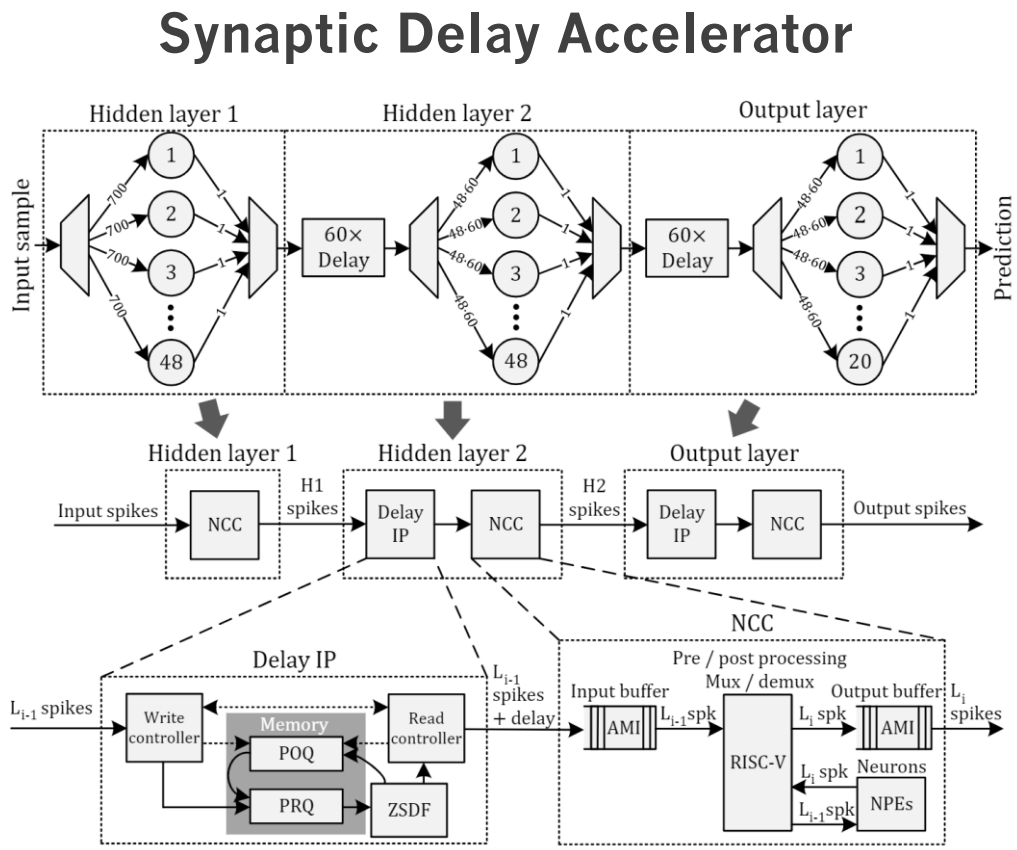
- Using open-source elements (RV + NoC)
- No accelerator

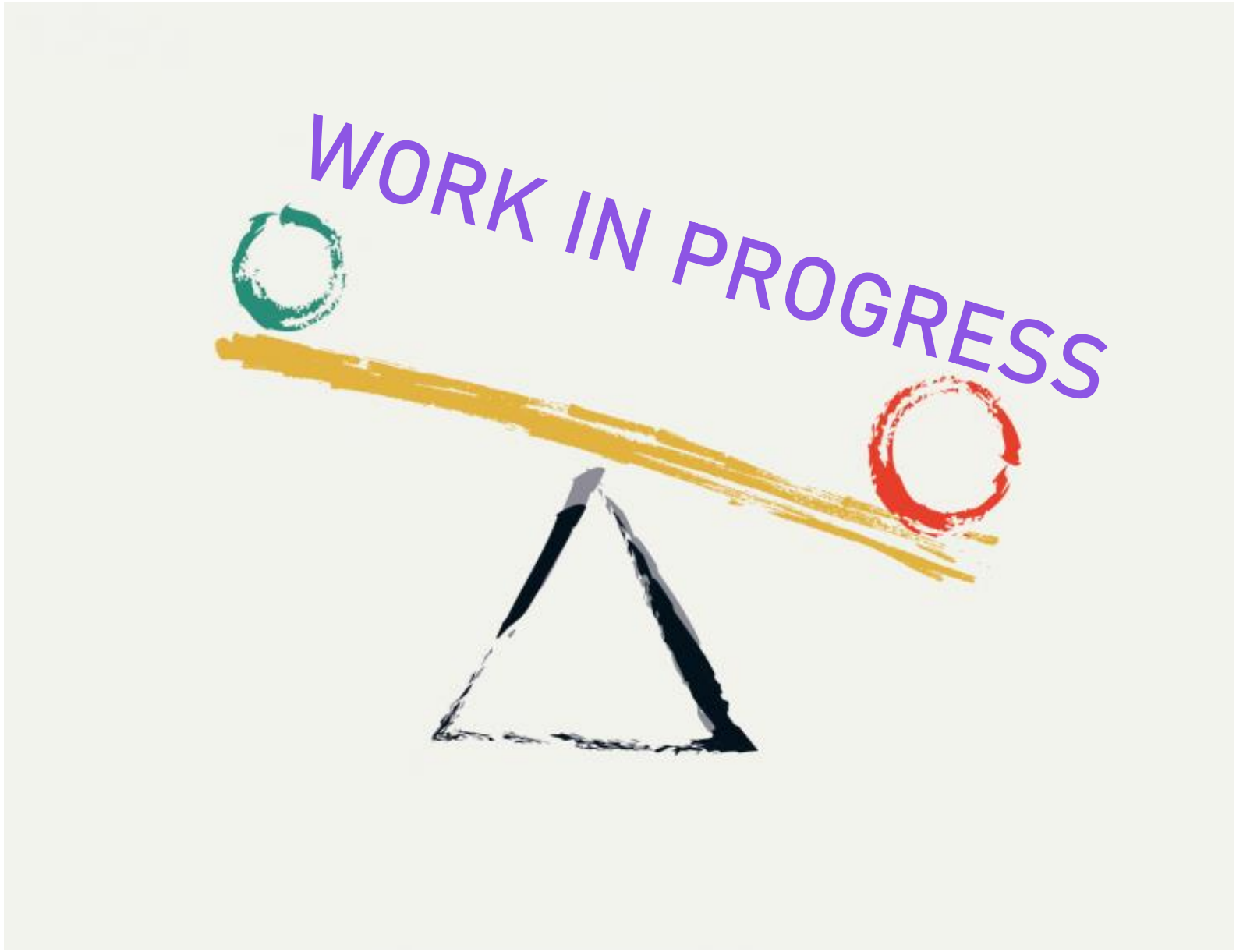
- **Can be deployed in off-the-shelf FPGA boards**
- **Can be used as the test-bench platform for exploring new technologies**



SENECA AS THE REFERENCE PLATFORM

Adding accelerators and measure performance is easily possible

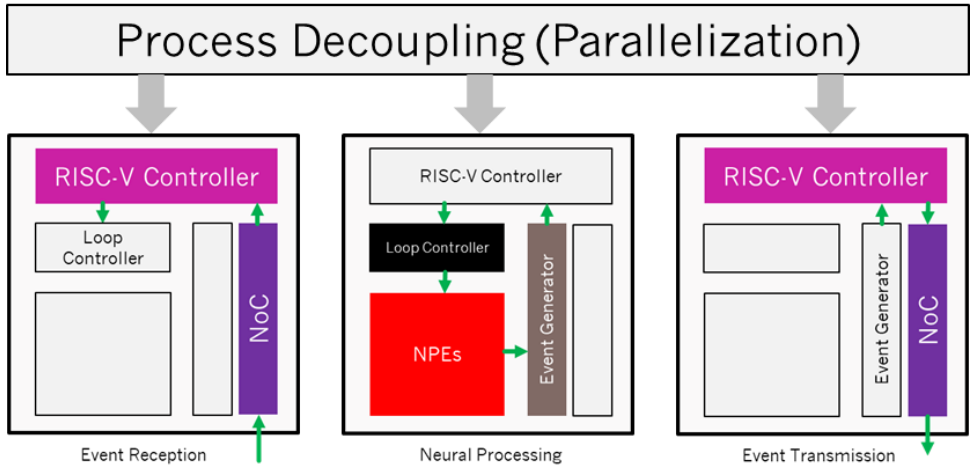




RISC-V CHALLENGE

Problem for layers with small Channel dimensions

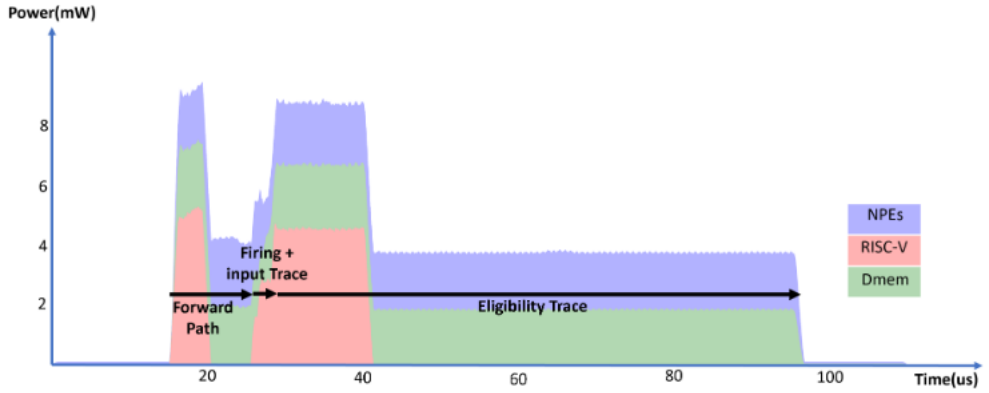
Energy consumption	Performance
• Instruction memory	• Pre-processing • Post-processing



Layer dimension	Total Energy (uJ)	RISC-V Energy (uJ)	Inst-RAM (uJ)
512	133	14 (10%)	12 (9%)
128	69	14 (20%)	12 (17%)

Processing 16 spikes
Network is 4x smaller, but consumes only 2x less 😐

ON DEVICE LEARNING



Algorithm phase	Time (μS)	RISC-V Energy (nJ)	NPEs Energy (nJ)	Dmem Energy (nJ)	Total Core Energy (nJ)	Normalized Energy (pJ)
Forward Path	10.5	13.4	18.8	20.7	56.2	18.3/SOp
Firing	0.9	1.3	2.0	0.8	4.5	35.0/Output
Input trace	1.1	2.3	0.9	1.1	4.7	29.3/Input
Eligibility trace	68.2	21.8	117.7	127.8	289.7	14.1/Weight

Table 11. Experimental Results for e-prop with RSNN. RISC-V energy includes RISC-V and its instruction memory. NPEs energy includes all NPEs, Loop buffer and Event-generator blocks.

FUTURE WORKS

- Neuron Processing Data type
- Interfacing to outside
- Internal interconnect (for programming)
- Advance neural networks architectures
- Asynchronous inference
- Memory technologies
- In-memory processing
- In-material processing
- Benchmarking with SOTA

THANKS!

Q&A