SenseMaker

IST2001-34712

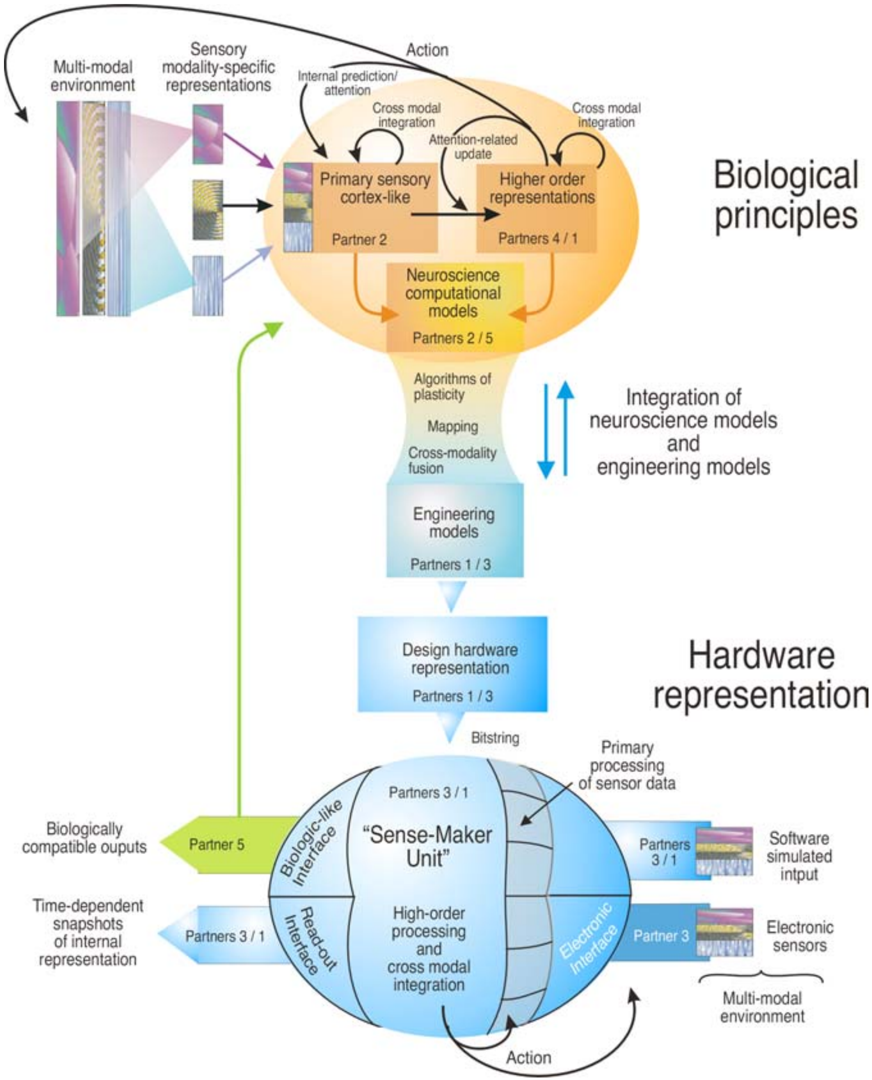
Neuro-IT Presentation

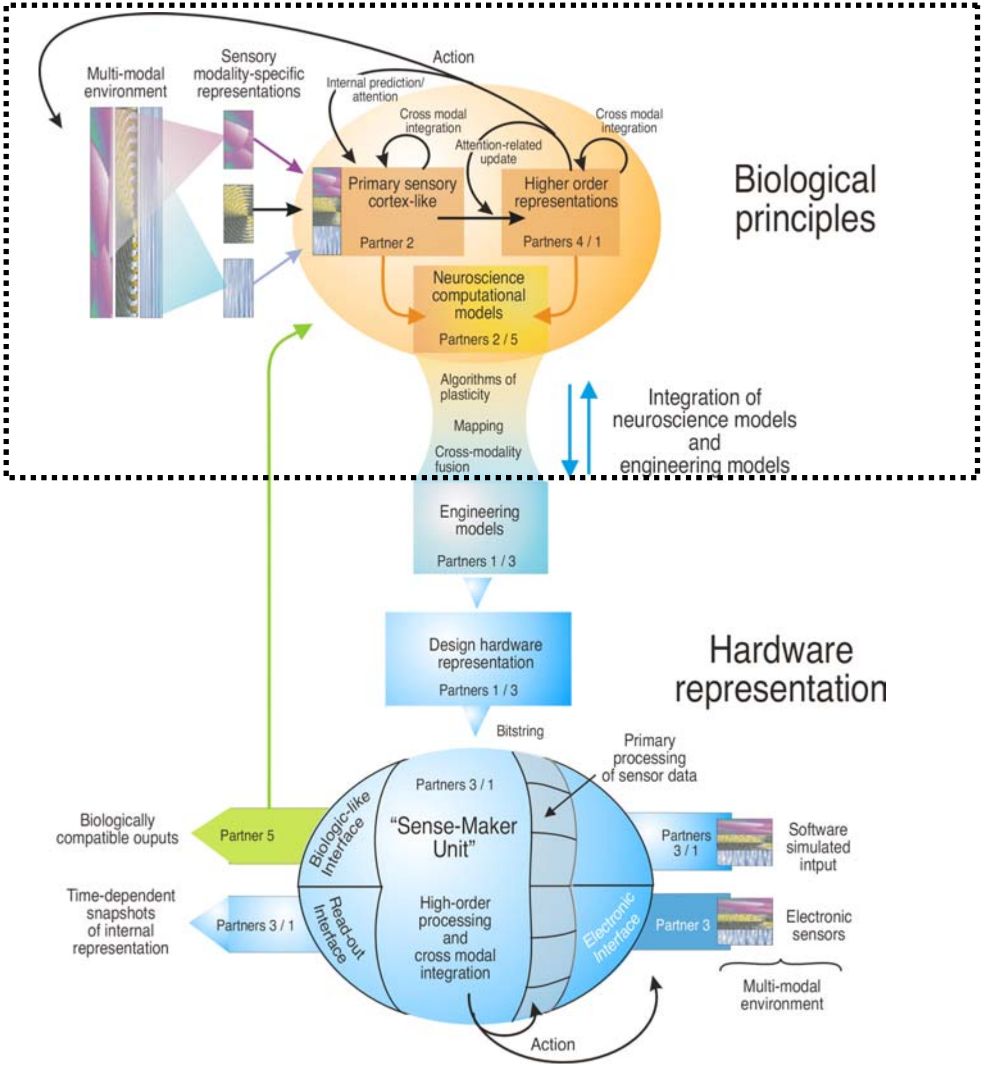
Alicante 2003

Project Partners

1	ISEL	University of Ulster (UK) - Intelligent Systems Engineering Laboratory	
2	UNIC-CNRS	Unité de Neurosciences Intégratives et Computationnelles, Centre National de la Recherche Scientifique (France)	
3	UHEI	Ruprecht-Karls-Universität Heidelberg (Germany), Kirchhoff-Institut für Physik	
4	TCD	Trinity College Dublin (Ireland) Visual Cognition Group, Institute for Neuroscience, Department of Psychology	
5	IXL-CNRS	ENSEIRB-CNRS Université Bordeaux 1 (France), IXL Laboratory	

Project Overview




Action
Multi-modal environment
Sensory modality-specific representations
Internal prediction/ attention
Cross modal integration
Attention-related update
Cross modal integration
Primary sensory cortex-like
Partner 2
Higher order representations
Partners 4 / 1
Neuroscience computational models
Partners 2 / 5
Biological principles
Algorithms of plasticity
Mapping
Cross-modality fusion
Integration of neuroscience models and engineering models
Engineering models
Partners 1 / 3
Design hardware representation
Partners 1 / 3
Hardware representation
Bitstring
Primary processing of sensor data
Partners 3 / 1
"Sense-Maker Unit"
High-order processing and cross modal integration
Biologic-like Interface
Read-out Interface
Electronic Interface
Biologically compatible ouputs
Partner 5
Time-dependent snapshots of internal representation
Partners 3 / 1
Partners 3 / 1
Software simulated intput
Partner 3
Electronic sensors
Multi-modal environment
Action

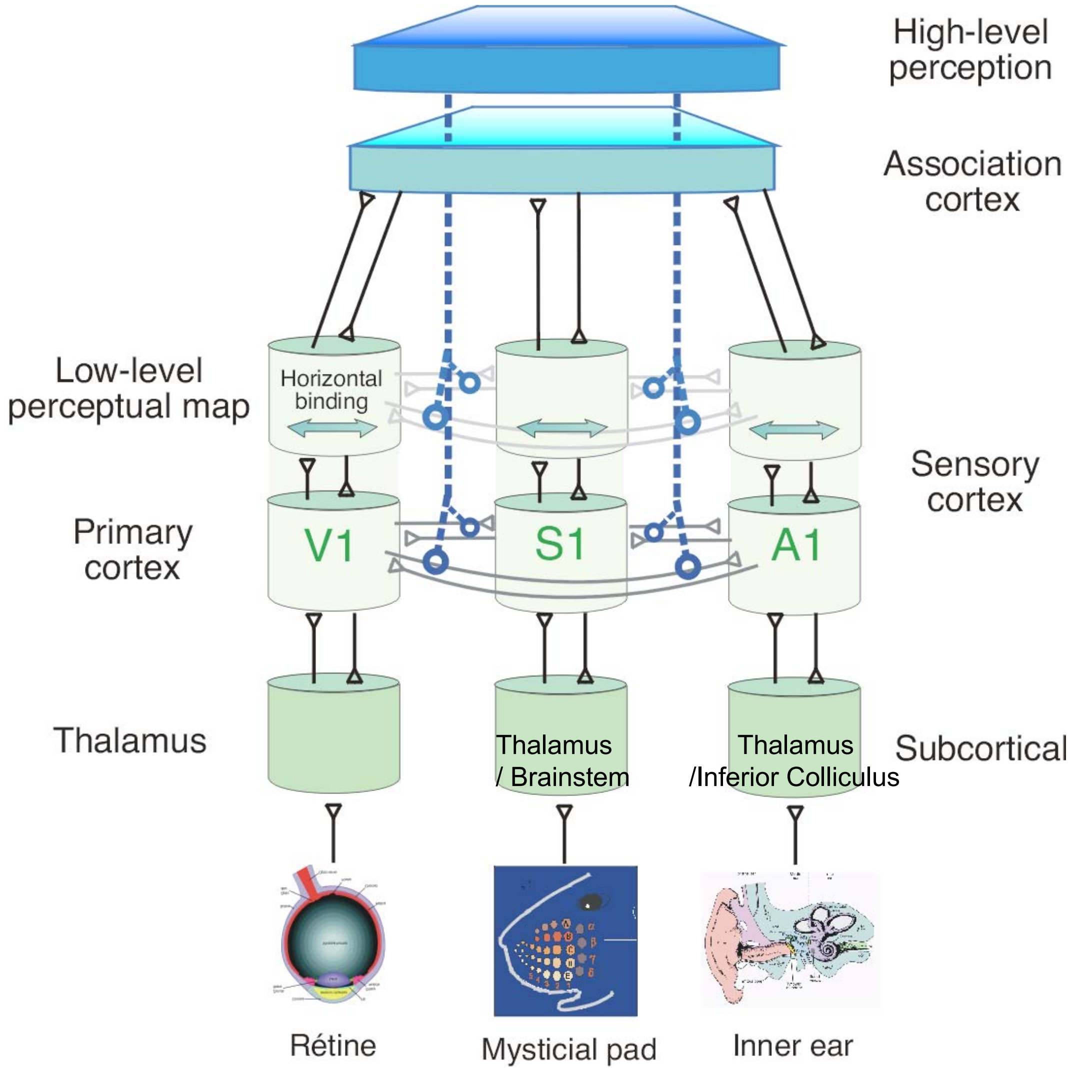
BIOLOGICAL PRINCIPLES

- ❖ SYSTEM LEVEL : BRAIN-LIKE ARCHITECTURE
- ❖ NEURAL PROCESSING LEVEL
- ❖ « NOISE » AS A COMPUTATIONAL PRINCIPLE
- ❖ CONSTRAINTS ON ARTIFICIAL NETWORK ARCHITECTURE
- ❖ LOW-LEVEL IMPLEMENTATION: THE TWO-RING PROBLEM

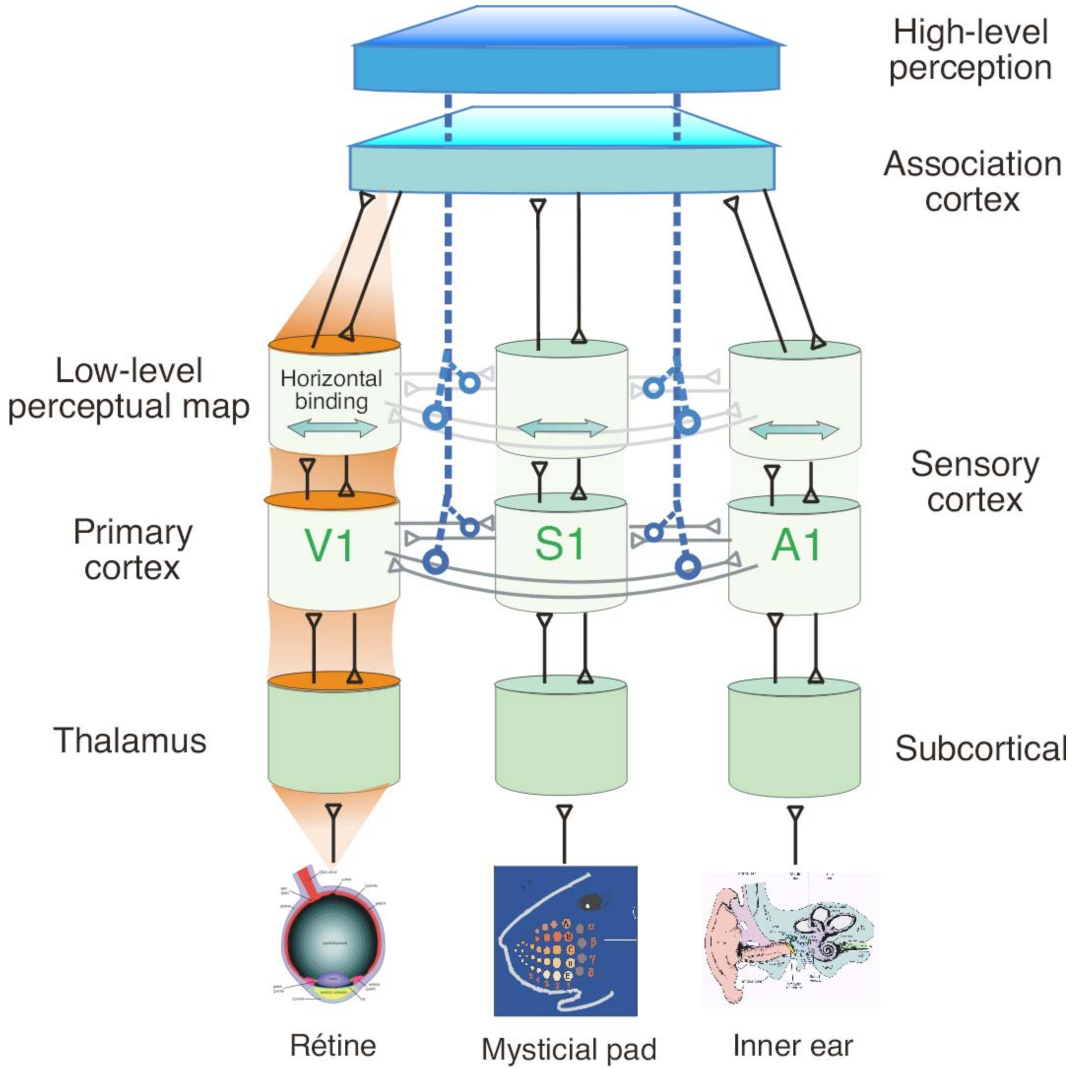
BIOLOGICAL PRINCIPLES

- **SYSTEM LEVEL : BRAIN-LIKE ARCHITECTURE**
- NEURAL PROCESSING LEVEL
- « NOISE » AS A COMPUTATIONAL PRINCIPLE
- CONSTRAINTS ON ARTIFICIAL NETWORK ARCHITECTURE
- LOW-LEVEL IMPLEMENTATION: THE TWO-RING PROBLEM

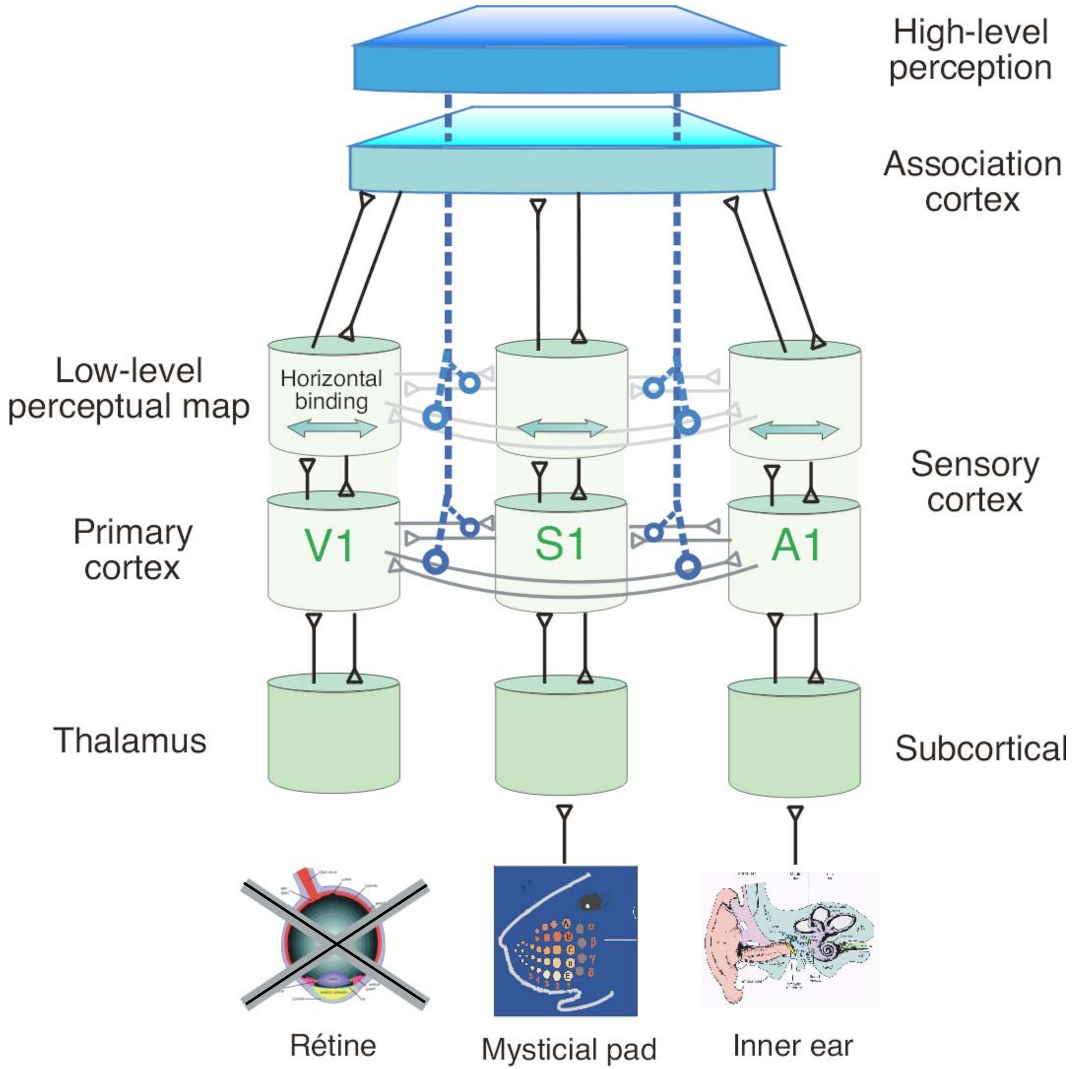
BIOLOGICAL PRINCIPLES: BRAIN-LIKE ARCHITECTURE



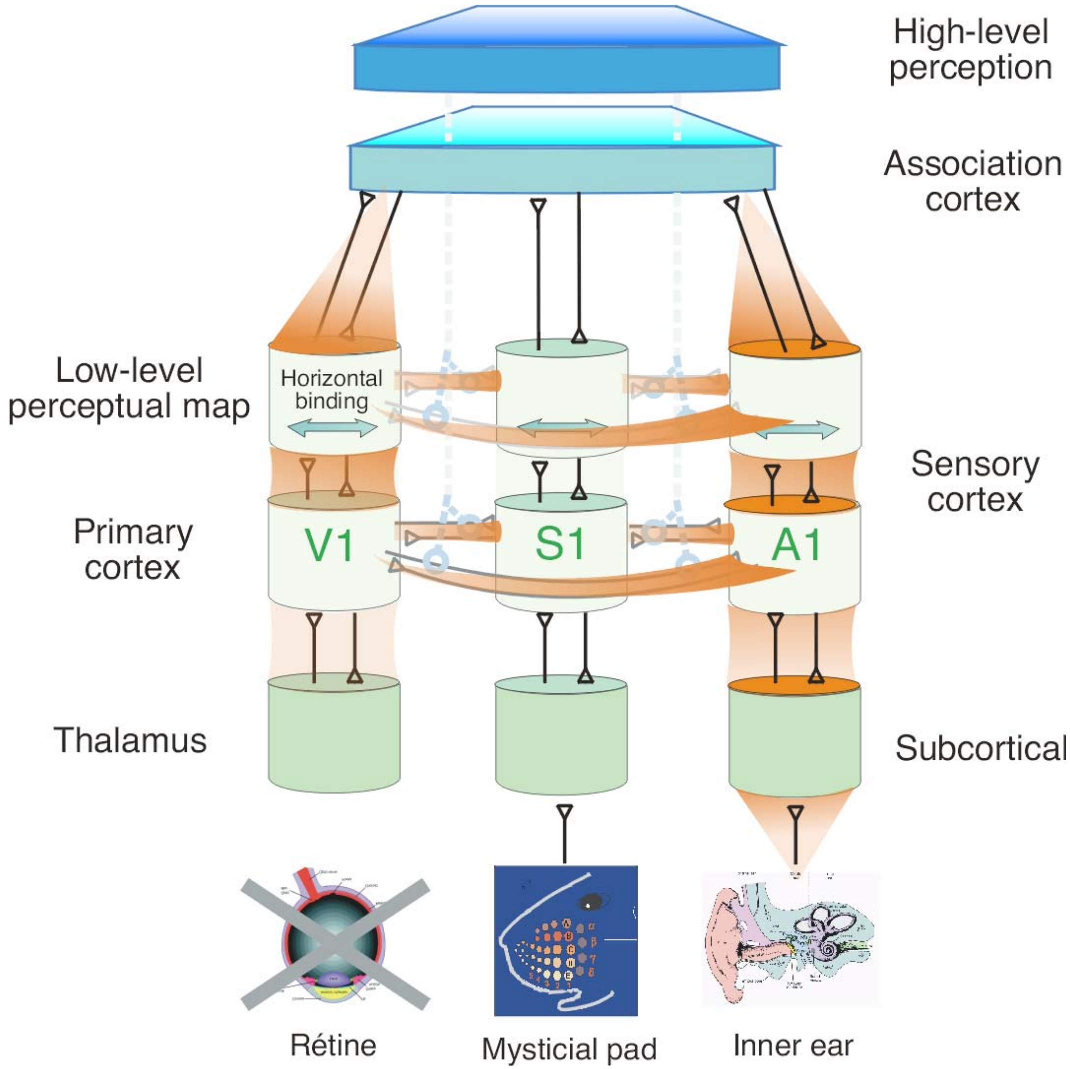
BIOLOGICAL PRINCIPLES: BRAIN-LIKE ARCHITECTURE



BIOLOGICAL PRINCIPLES: ADAPTIVE PROPERTIES



BIOLOGICAL PRINCIPLES: SENSORY SUBSTITUTION



BIOLOGICAL PRINCIPLES

- SYSTEM LEVEL : BRAIN-LIKE ARCHITECTURE
- **NEURAL PROCESSING LEVEL**
- « NOISE » AS A COMPUTATIONAL PRINCIPLE
- CONSTRAINTS ON ARTIFICIAL NETWORK ARCHITECTURE
- LOW-LEVEL IMPLEMENTATION: THE TWO-RING PROBLEM

BIOLOGICAL PRINCIPLES : *Rate* vs.*Time Coding*

SPARSE NOISE

QuickTime™ et un décompresseur
BMP sont requis pour visualiser
cette image.

DENSE NOISE

QuickTime™ et un décompresseur
BMP sont requis pour visualiser
cette image.

NATURAL IMAGE

QuickTime™ et un décompresseur
BMP sont requis pour visualiser
cette image.

CONTRAST EDGE

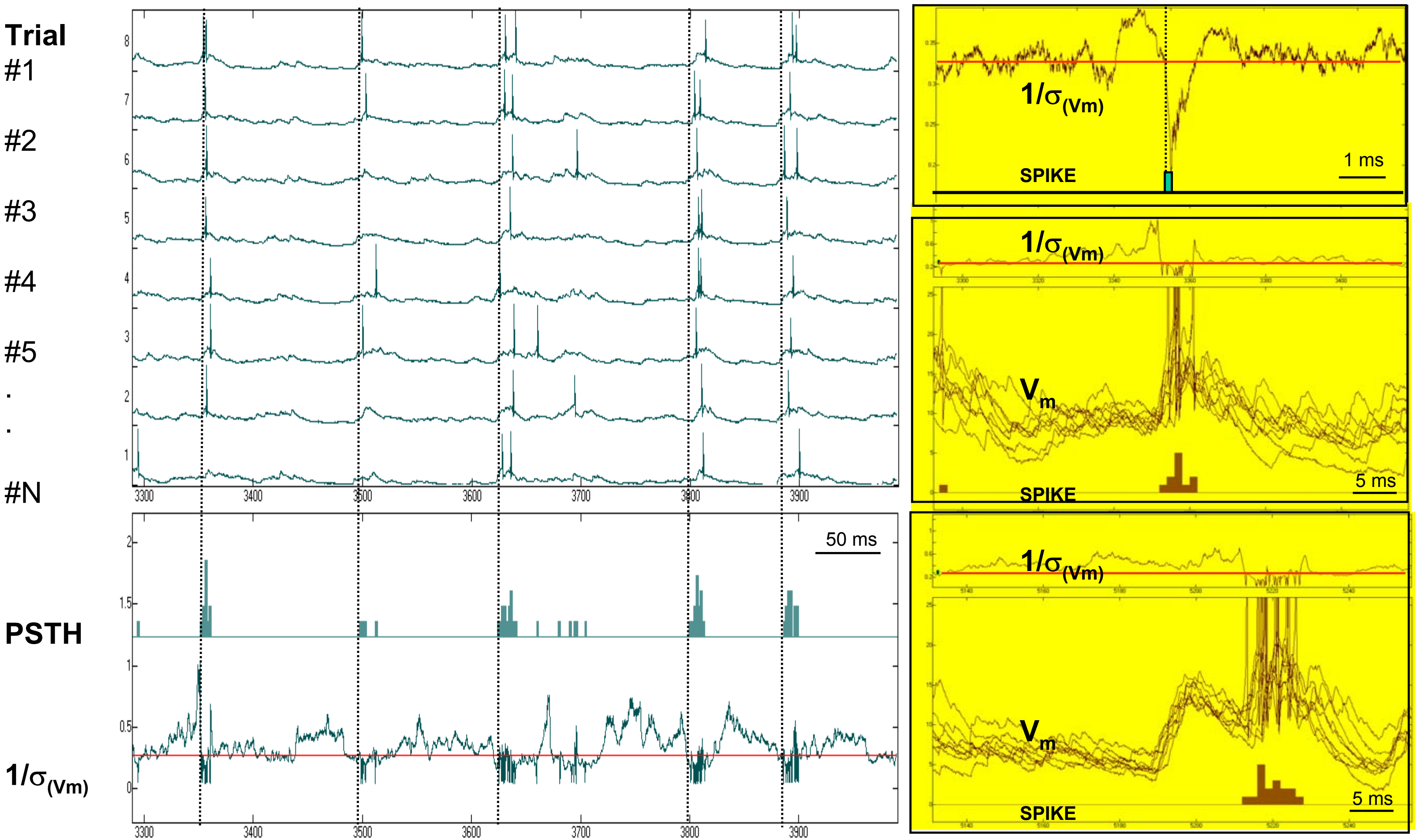
QuickTime™ et un décompresseur
BMP sont requis pour visualiser
cette image.

GRATING

QuickTime™ et un décompresseur
BMP sont requis pour visualiser
cette image.

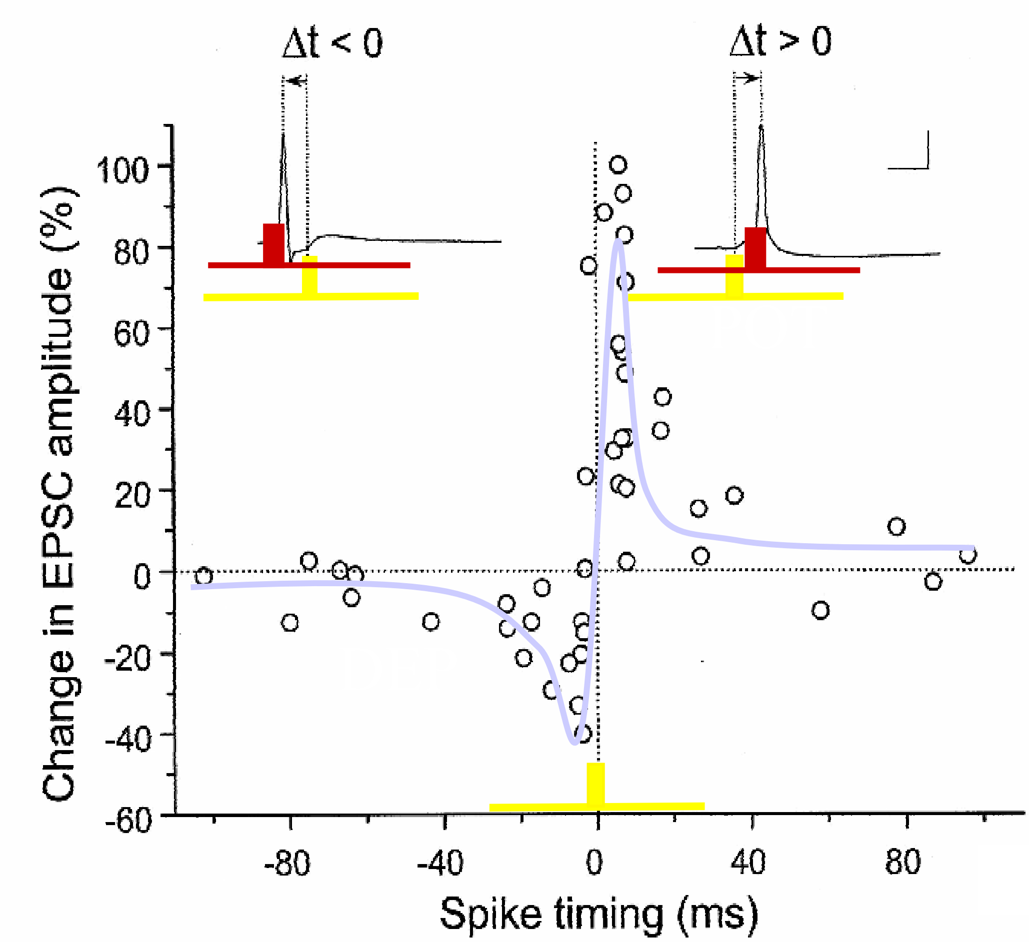
INCREASED LEVELS OF COMPUTATIONAL LOAD

BIOLOGICAL PRINCIPLES : *Time Coding*



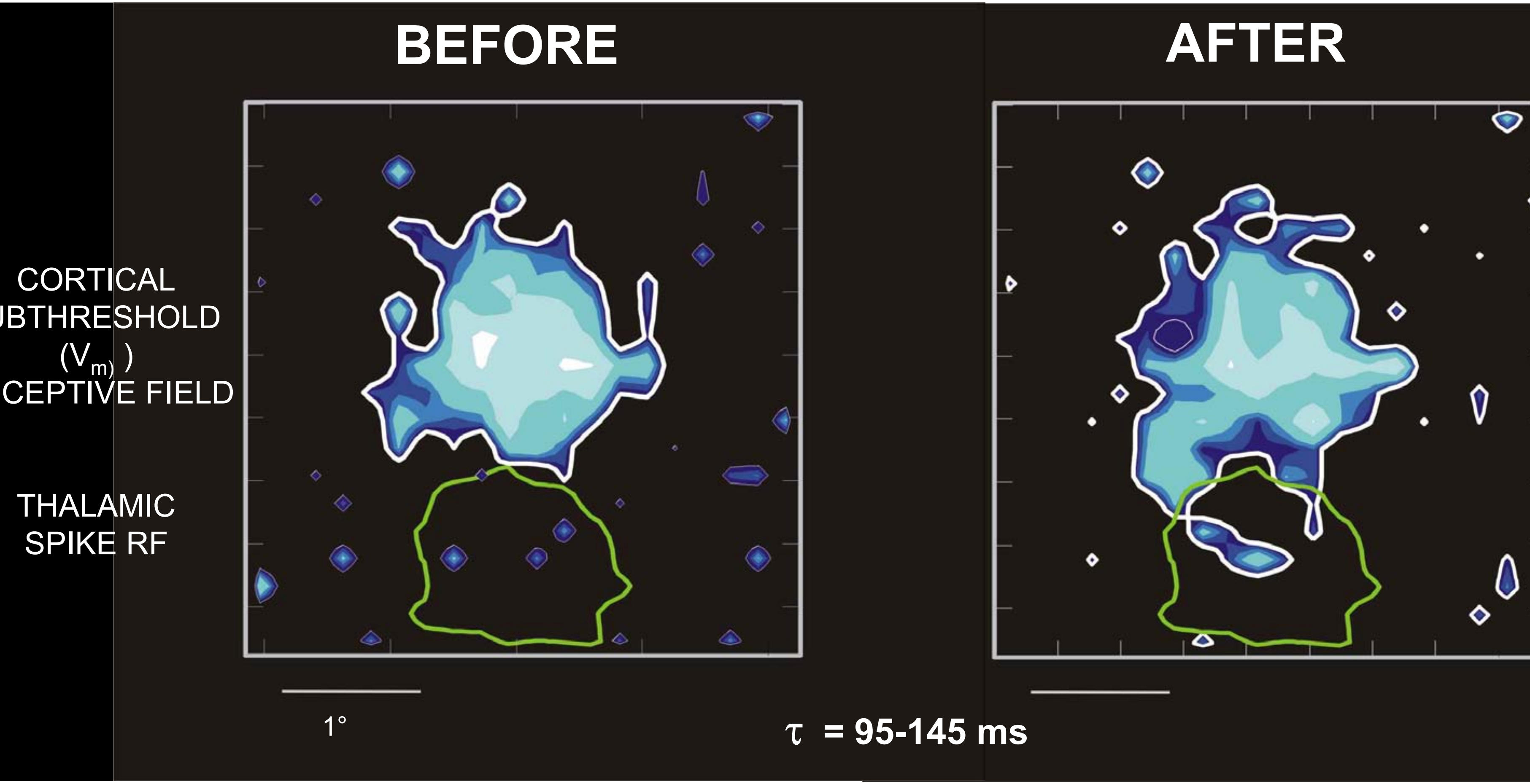
INVARIANCE OF THE FIRING PATTERN WHEN ITERATING THE SAME DENSE NOISE SEED

BIOLOGICAL PRINCIPLES: SPIKE-TIMING DEPENDENT PLASTICITY (S.T.D.P.)



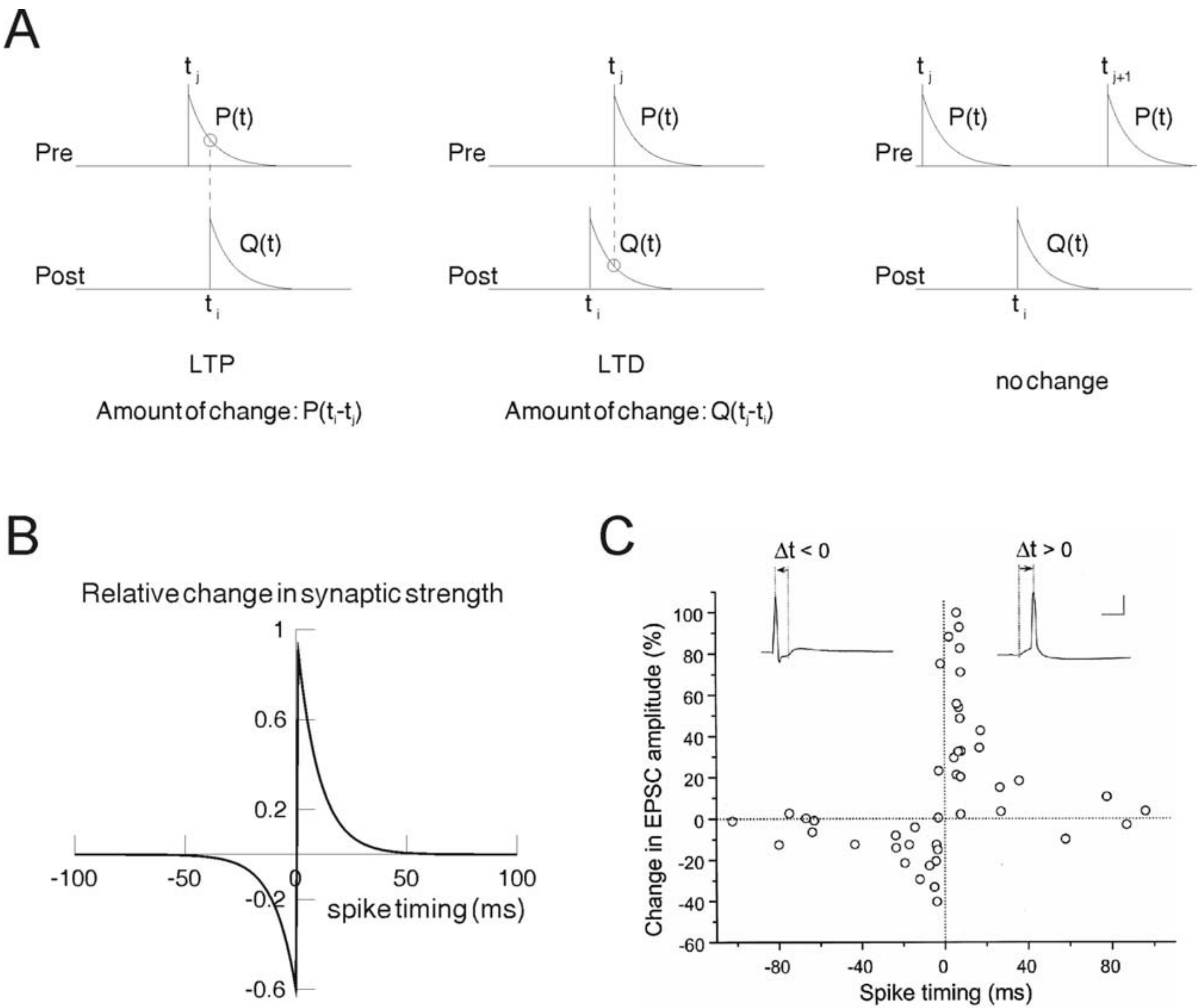
(Bi and Poo, Ann. Rev. Neurosci., 2001)

BIOLOGICAL PRINCIPLES : *IN VIVO* SPIKE-TIMING DEPENDENT POTENTIATION



FUNCTIONAL CORRELATE OF LTP : INDUCTION OF AN ECTOPIC RECEPTIVE FIELD

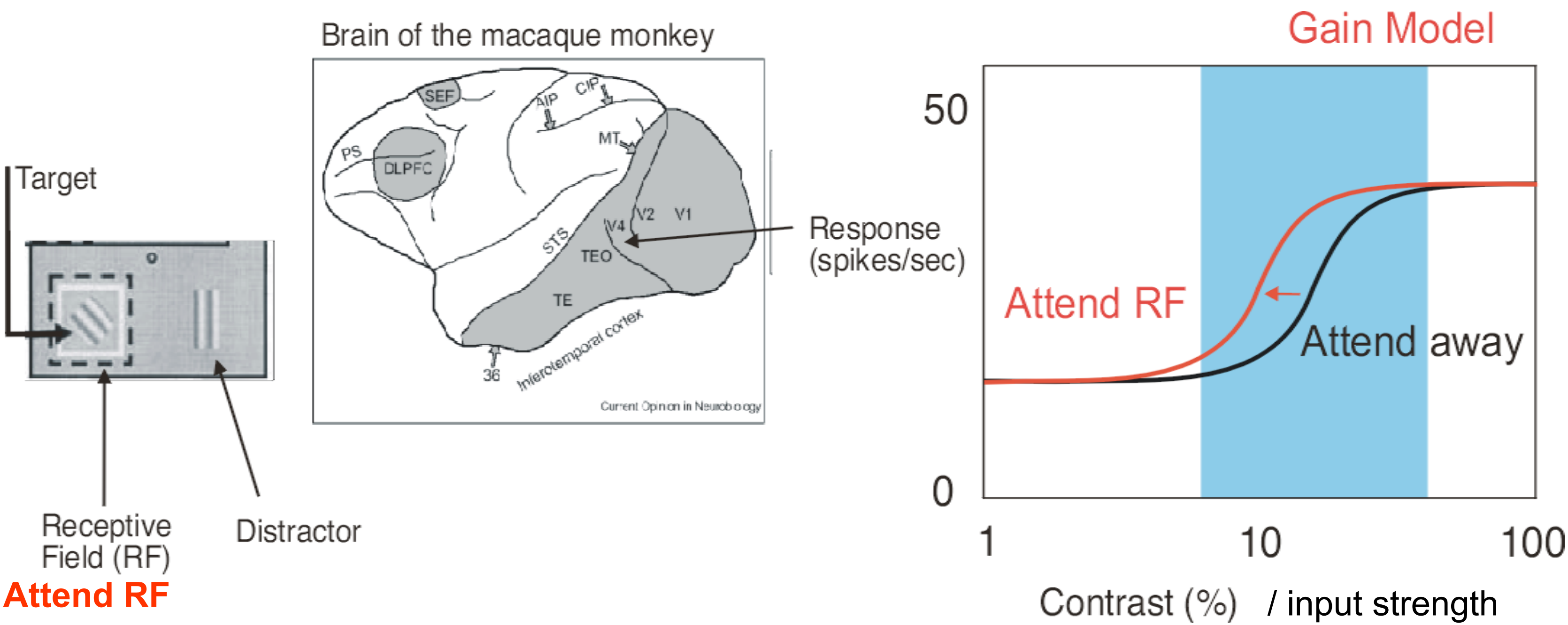
BIOLOGICAL PRINCIPLES : STDP ALGORITHM



BIOLOGICAL PRINCIPLES

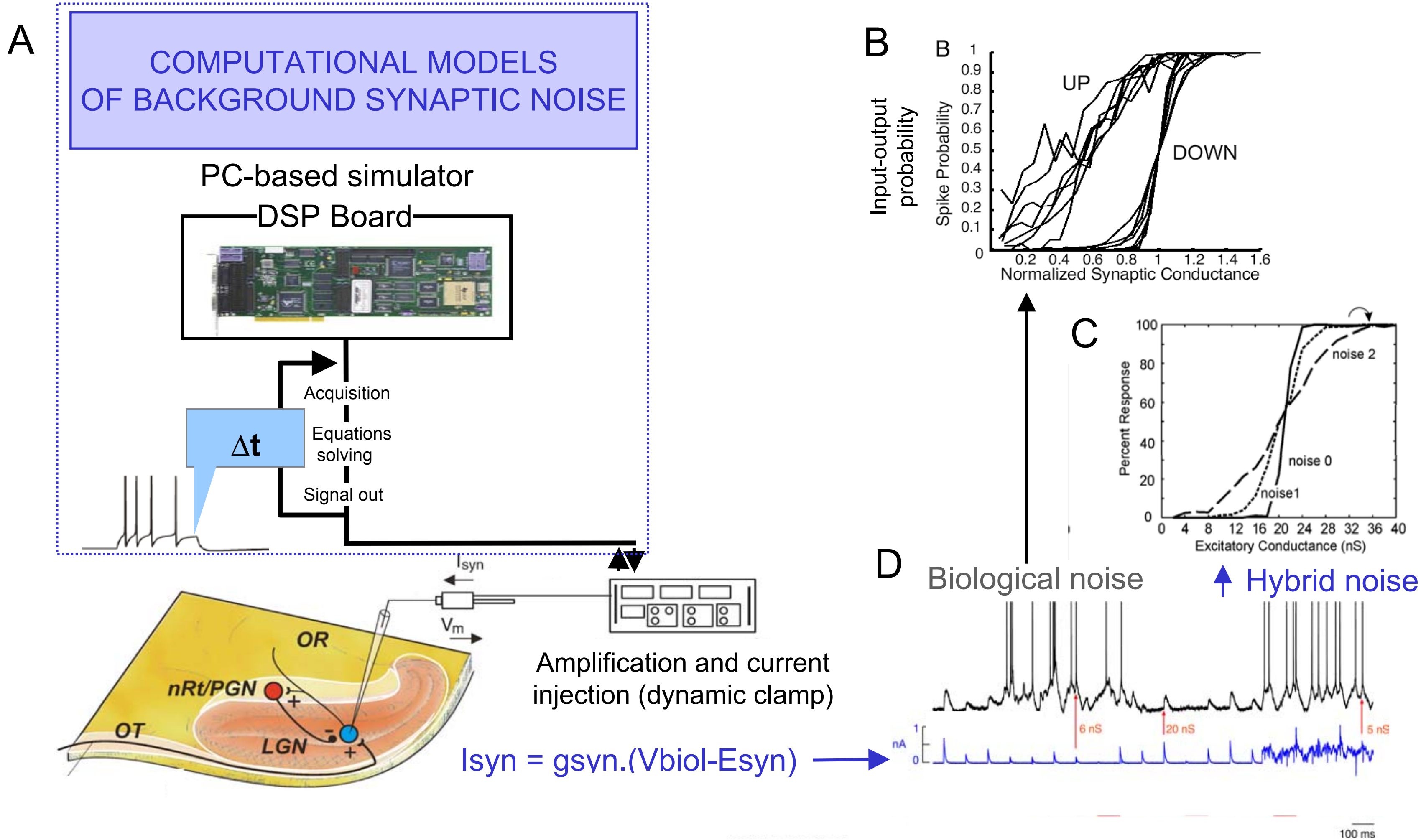
- SYSTEM LEVEL : BRAIN-LIKE ARCHITECTURE
- NEURAL PROCESSING LEVEL
- **« NOISE » AS A COMPUTATIONAL PRINCIPLE**
- CONSTRAINTS ON ARTIFICIAL NETWORK ARCHITECTURE
- LOW-LEVEL IMPLEMENTATION: THE TWO-RING PROBLEM

BIOLOGICAL PRINCIPLES : NOISE CONTROL OF NEURONAL RESPONSE PROBABILITY FUNCTIO



Synaptic noise, a new principle for attentional facilitation?

BIOLOGICAL PRINCIPLES : NOISE CONTROL OF NEURONAL RESPONSE PROBABILITY FUNCTIO

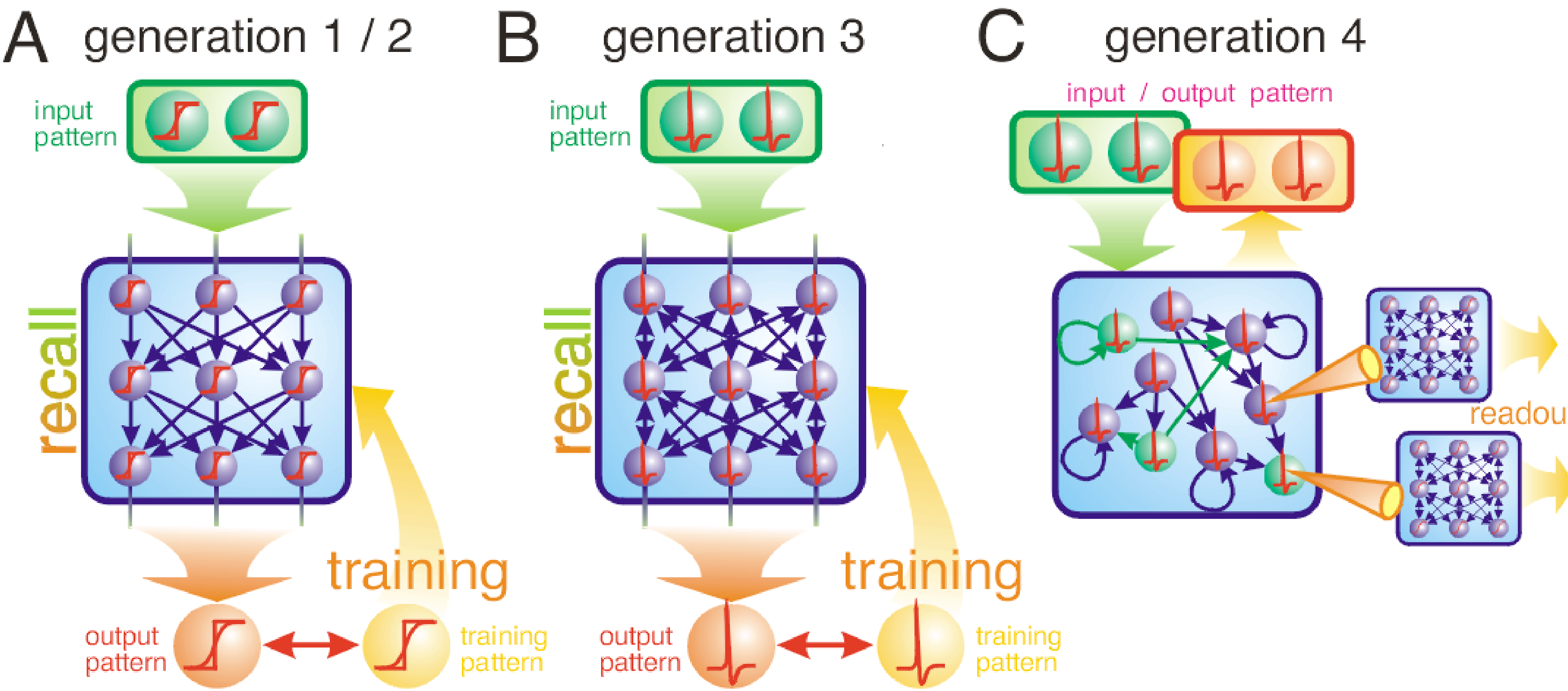


Hybrid artificial-biological neural networks *in vitro*

BIOLOGICAL PRINCIPLES

- SYSTEM LEVEL : BRAIN-LIKE ARCHITECTURE
- NEURAL PROCESSING LEVEL
- « NOISE » AS A COMPUTATIONAL PRINCIPLE
- **CONSTRAINTS ON ARTIFICIAL NETWORK ARCHITECTURE**
- LOW-LEVEL IMPLEMENTATION: THE TWO-RING PROBLEM

BIOLOGICAL PRINCIPLES : CONSTRAINTS ON ARTIFICIAL NEURAL NETWORK ARCHITECTURES



PERCEPTRONS
HOPFIELD / BOLTZMANN
MCCULLOCH-PITTS NEURONS
ACTIVATION FUNCTION

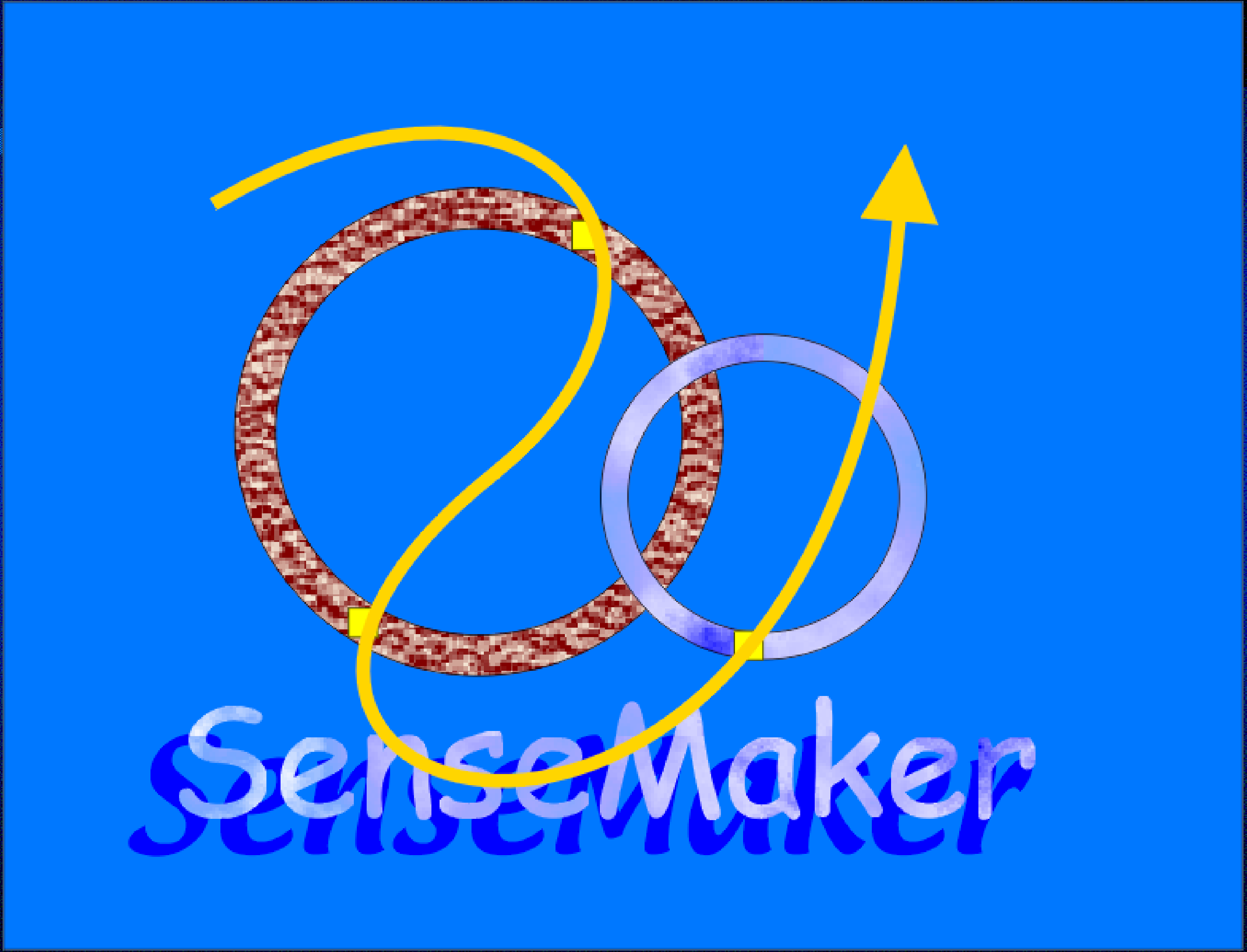
SPIKING RECURRENT
NEURAL NETWORKS
INTEGRATE-AND-FIRE NEURONS

ANY-TIME COMPUTING
LIQUID COMPUTATION
TIME CODING AND STDP

BIOLOGICAL PRINCIPLES

- SYSTEM LEVEL : BRAIN-LIKE ARCHITECTURE
- NEURAL PROCESSING LEVEL
- « NOISE » AS A COMPUTATIONAL PRINCIPLE
- CONSTRAINTS ON ARTIFICIAL NETWORK ARCHITECTURE
- **LOW-LEVEL IMPLEMENTATION: THE TWO-RING PROBLEM**

IMPLEMENTATION OF LOW-LEVEL ADAPTIVE PRINCIPLES

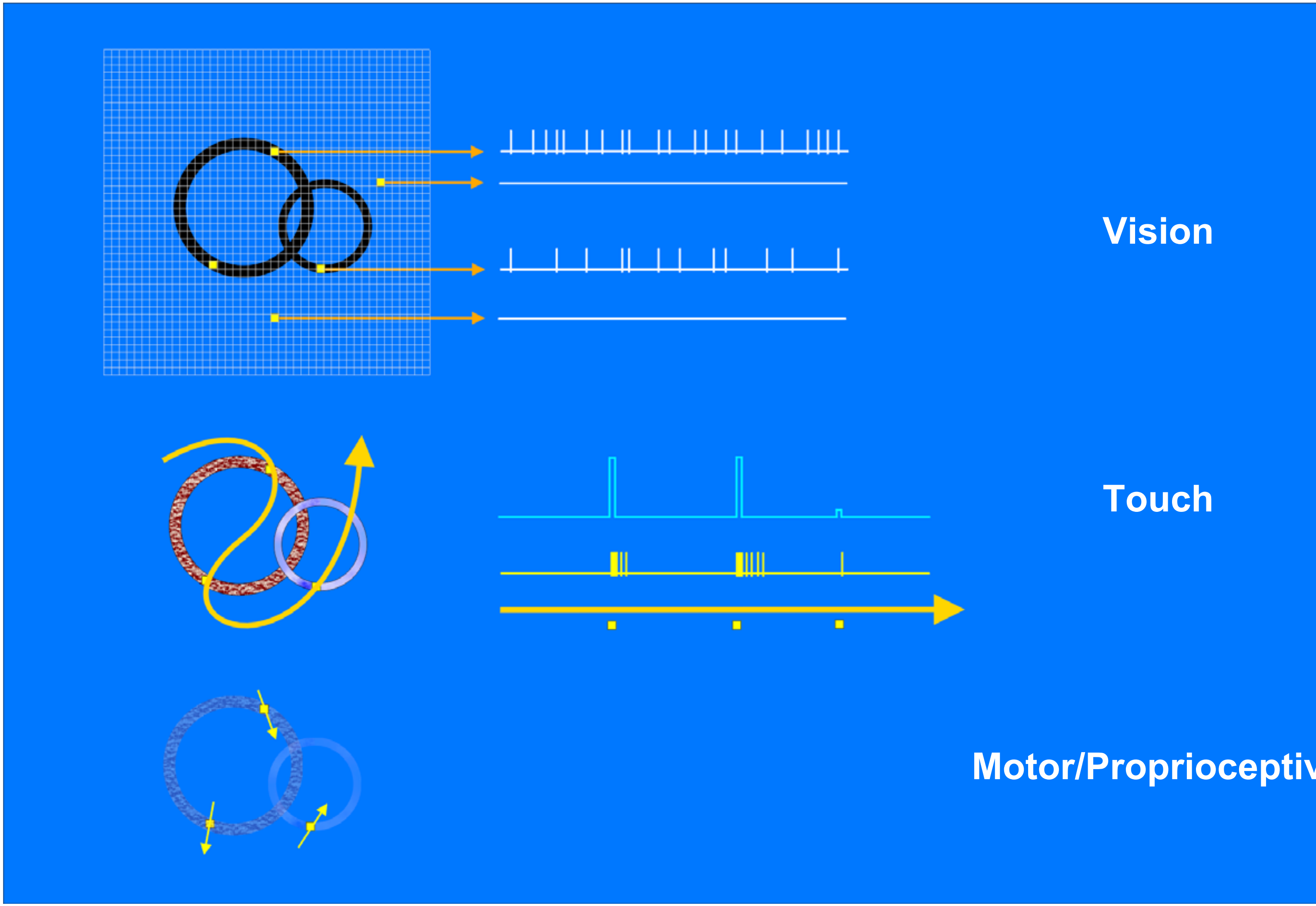


IMPLEMENTATION OF LOW-LEVEL ADAPTIVE PRINCIPLES : THE TWO-RING PROBLEM

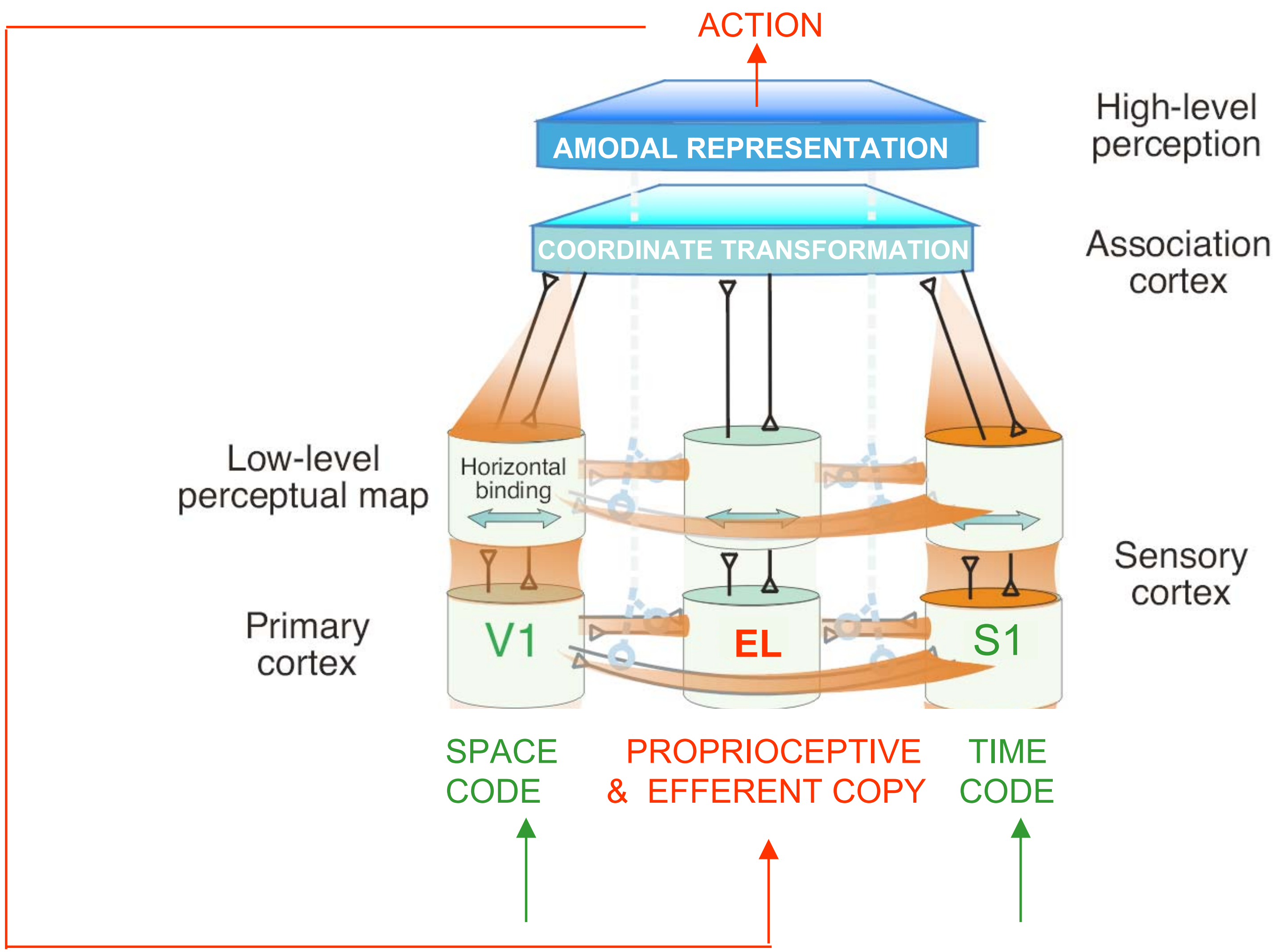
Space code

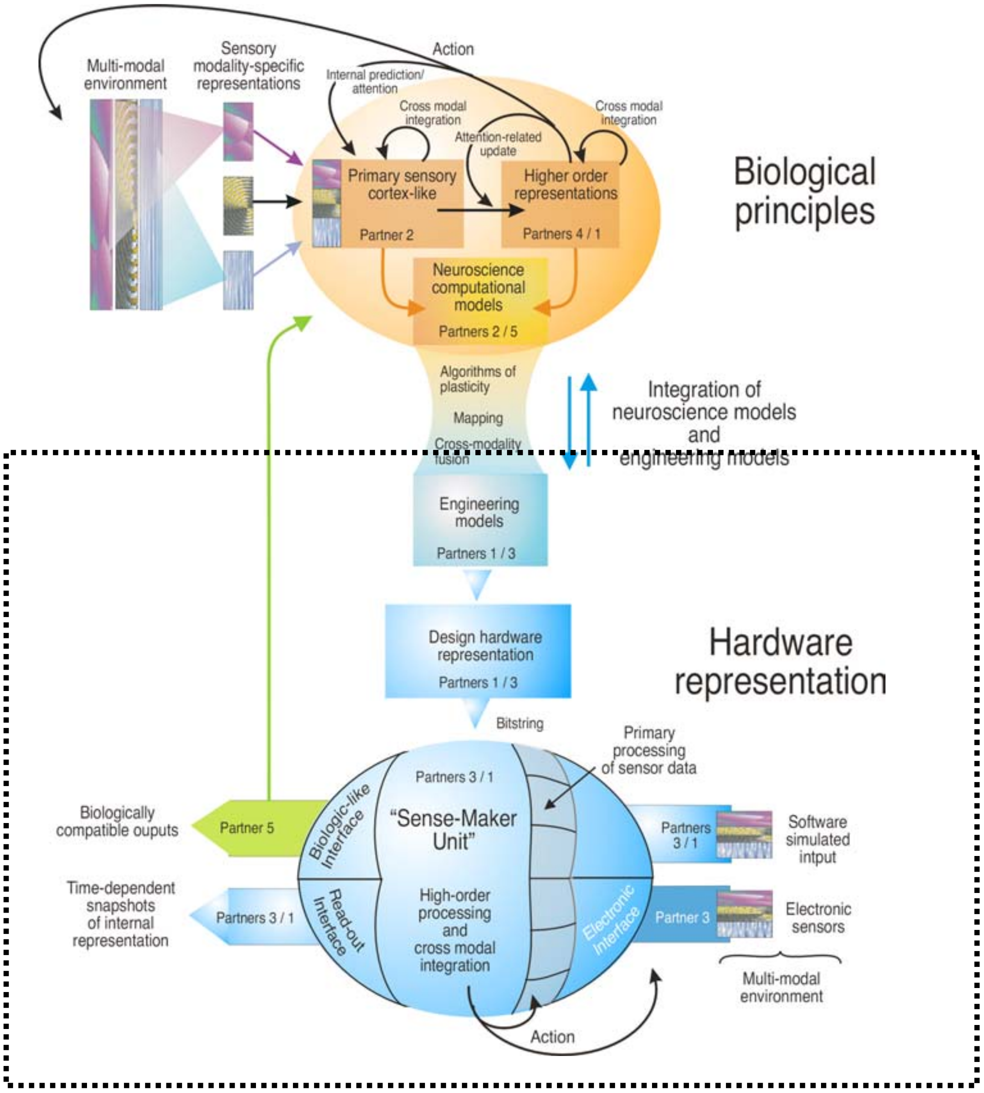
Time code

Action code



IMPLEMENTATION OF LOW-LEVEL ADAPTIVE PRINCIPLES : THE TWO-RING PROBLEM




Action
Multi-modal environment
Sensory modality-specific representations
Internal prediction/ attention
Cross modal integration
Attention-related update
Cross modal integration
Primary sensory cortex-like
Partner 2
Higher order representations
Partners 4 / 1
Neuroscience computational models
Partners 2 / 5
Biological principles
Algorithms of plasticity
Mapping
Cross-modality fusion
Integration of neuroscience models and engineering models
Engineering models
Partners 1 / 3
Design hardware representation
Partners 1 / 3
Hardware representation
Bitstring
Primary processing of sensor data
Partners 3 / 1
"Sense-Maker Unit"
High-order processing and cross modal integration
Biologically compatible ouputs
Partner 5
Biologic-like Interface
Time-dependent snapshots of internal representation
Partners 3 / 1
Read-out Interface
Electronic Interface
Partners 3 / 1
Software simulated intput
Partner 3
Electronic sensors
Multi-modal environment
Action

VLSI Development for the SenseMaker Project

Task

Design, construction and operation of
VLSI based substrates for biologically inspired neural computation with sensory input and (possibly) bio-compatible output

Concept

Follow 2 parallel strategies and transfer results :

close biological model - low complexity

simplified biological model - high complexity

Ruprecht-Karls-Universität Heidelberg
Kirchhoff-Institut für Physik

Université Bordeaux
ENSEIRB-CNRS

SenseMaker Roadmap from Biology to VLSI

ixed signal
erceptron

Design and Construction | Experiments

istributed
ystem

Design and Construction | Experiments

ixed signal
NN

VLSI Implementation of biological principles | Design and Construction | Experi-ments

high complexity

mixed signal

high speed

simplified biological models

scalability

ngle neuron
nalog SNN

Design and Construction | Experiments

ulti neuron
nalog SNN

Design and Construction | Experiments

low complexity

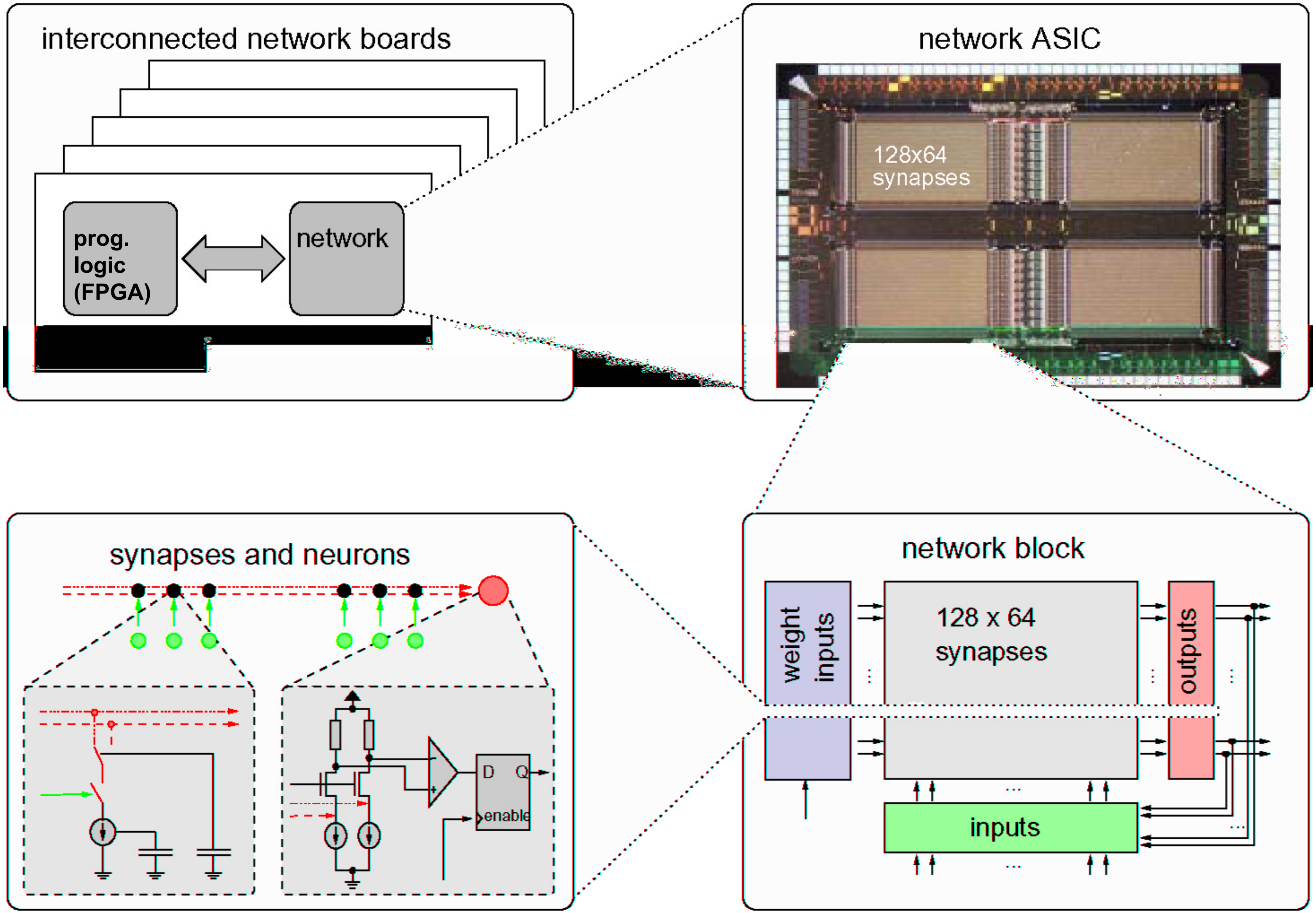
fully analog

biological speed

complex biological models

configurability

A mixed-signal Recurrent Perceptron FPNN System



Mixed Signal Perceptron : Design Considerations

Hardware

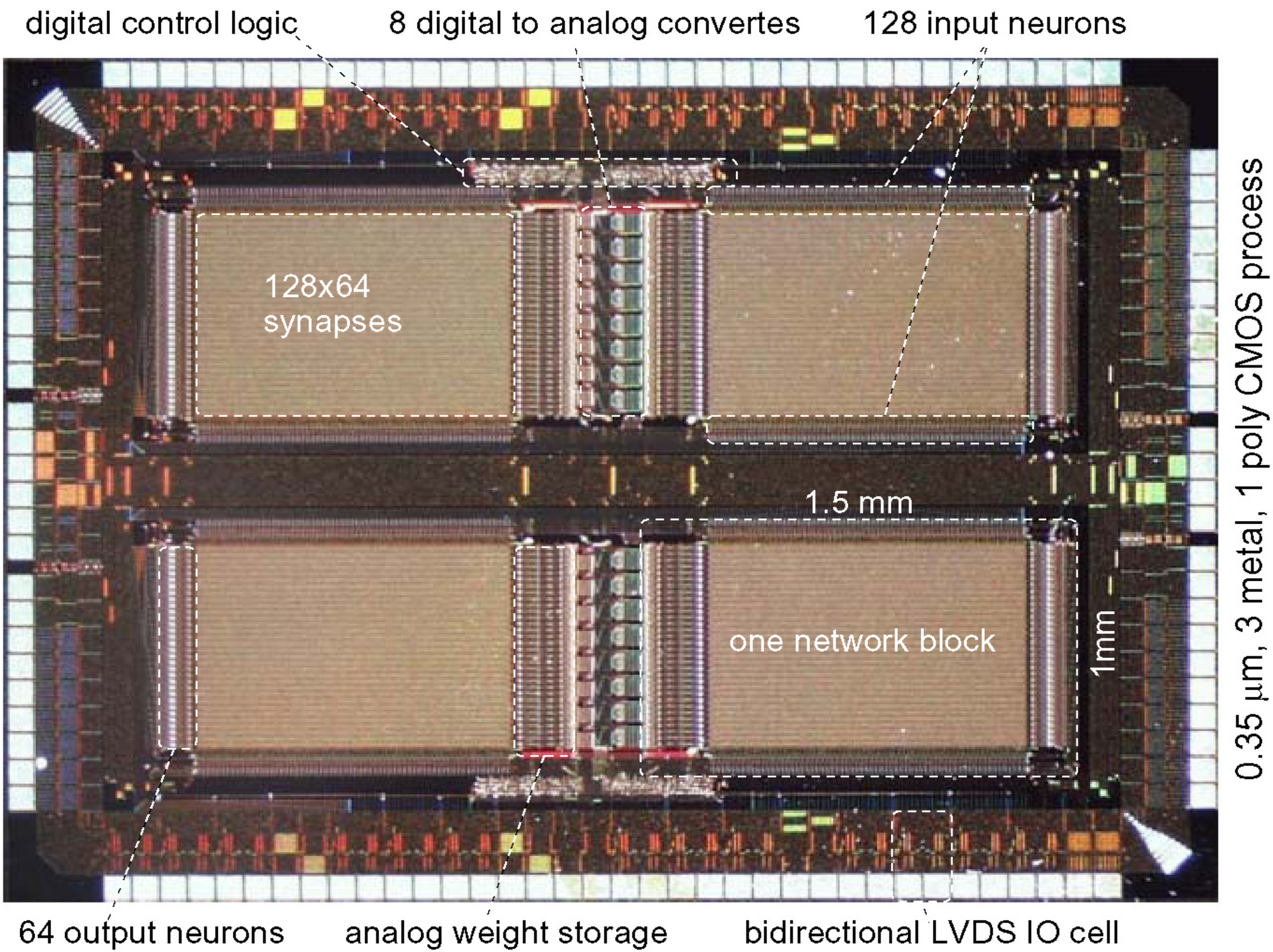
- FPNN chip with **mixed signal** neural network function
- **Analog** weights, **digital** communication
- 0.35 µm CMOS technology
- 128 input neurons and 64 output neurons in 4 blocks per chip
- 32768 synapses with size **6µm x 12 µm**
- Programmable hardware logic (FPGA) which accesses chip purely digitally and accelerate the low-level functions of the training algorithms
- Dedicated PCI card, so the hardware can be used with a common PC.

Software

- Low-level hardware abstraction layer (HAL)
- User interface

System is fully operational for experiments today

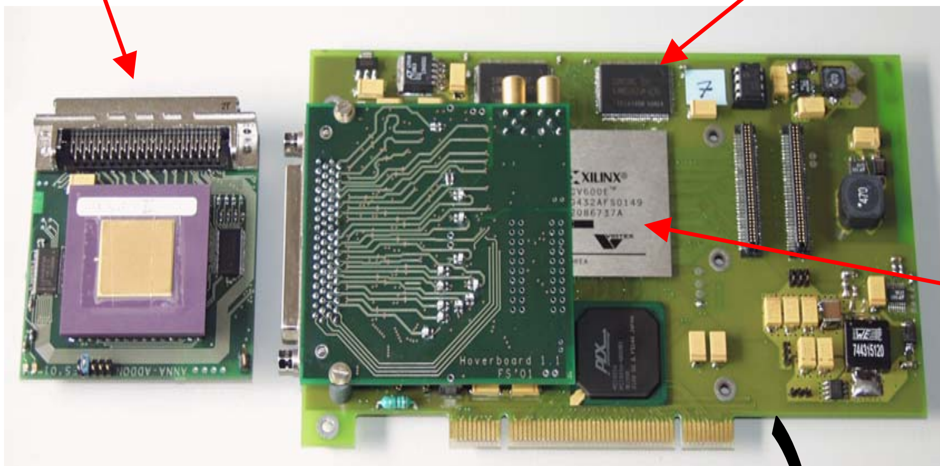
Microphotograph of the 0.35 µm VLSI Chip (HAGEN)



The Experimental Setup

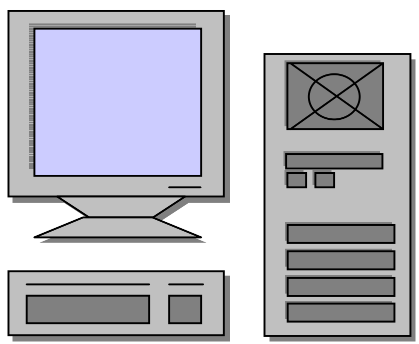
neural network chip FPNN on adapter board

local memory to store the neural weights and the training data



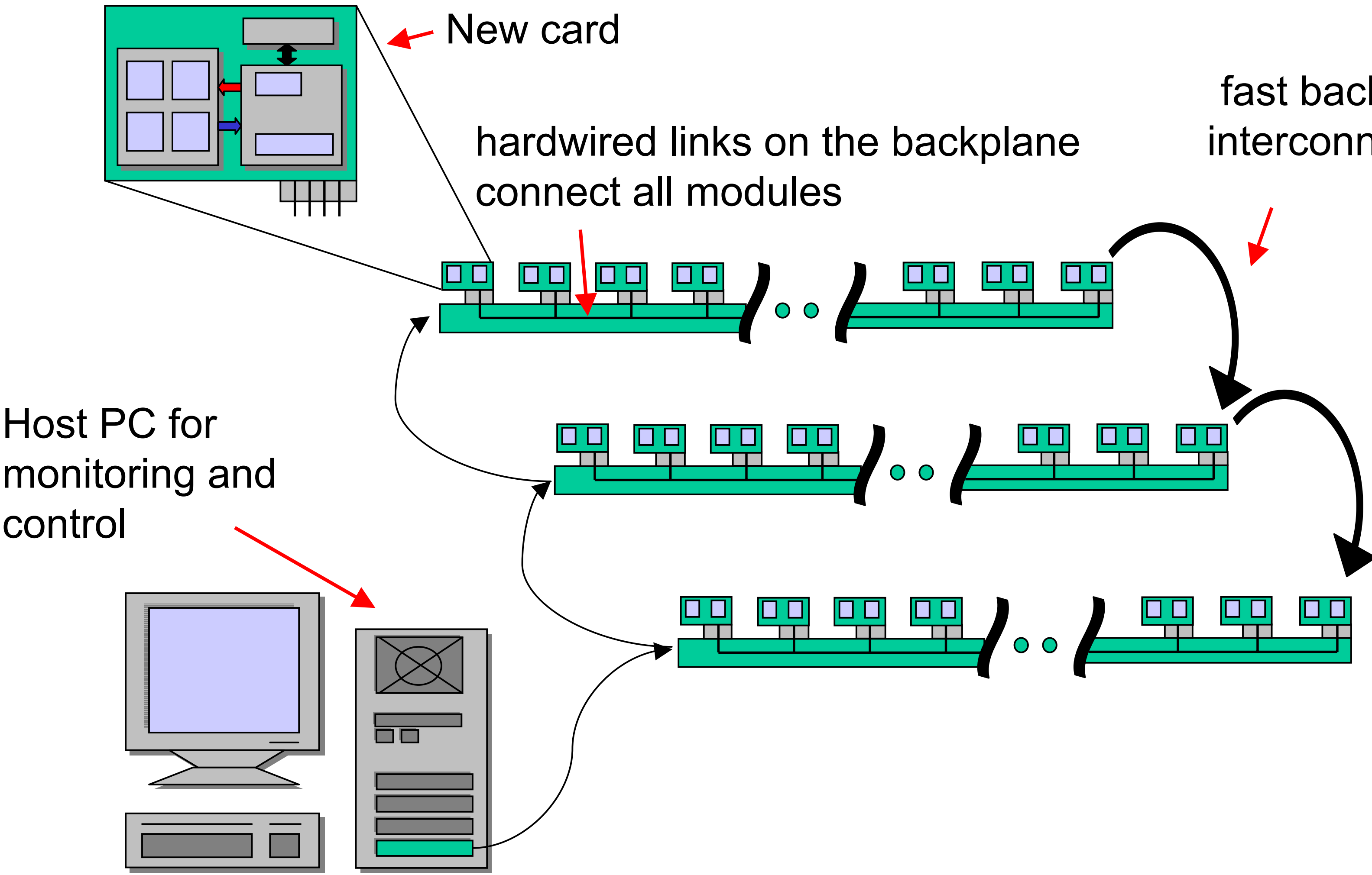
programmable hardware to execute low-level parts of the training algorithms

user interface and higher level parts of the training algorithms run on the PC

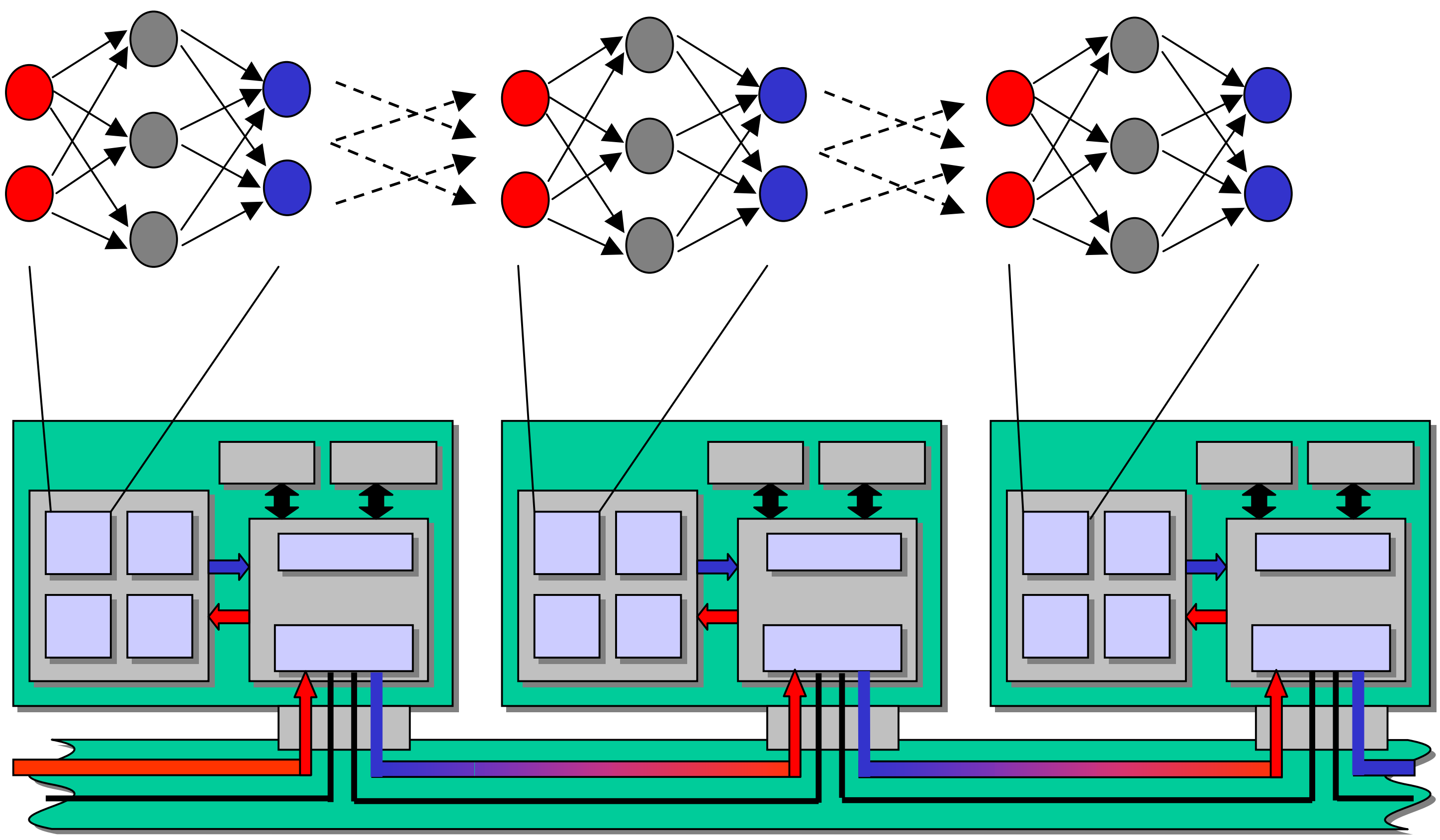


PCI bus allows using the hardware with a standard PC

Distributed Neural Processing : Schematic Functional Overview

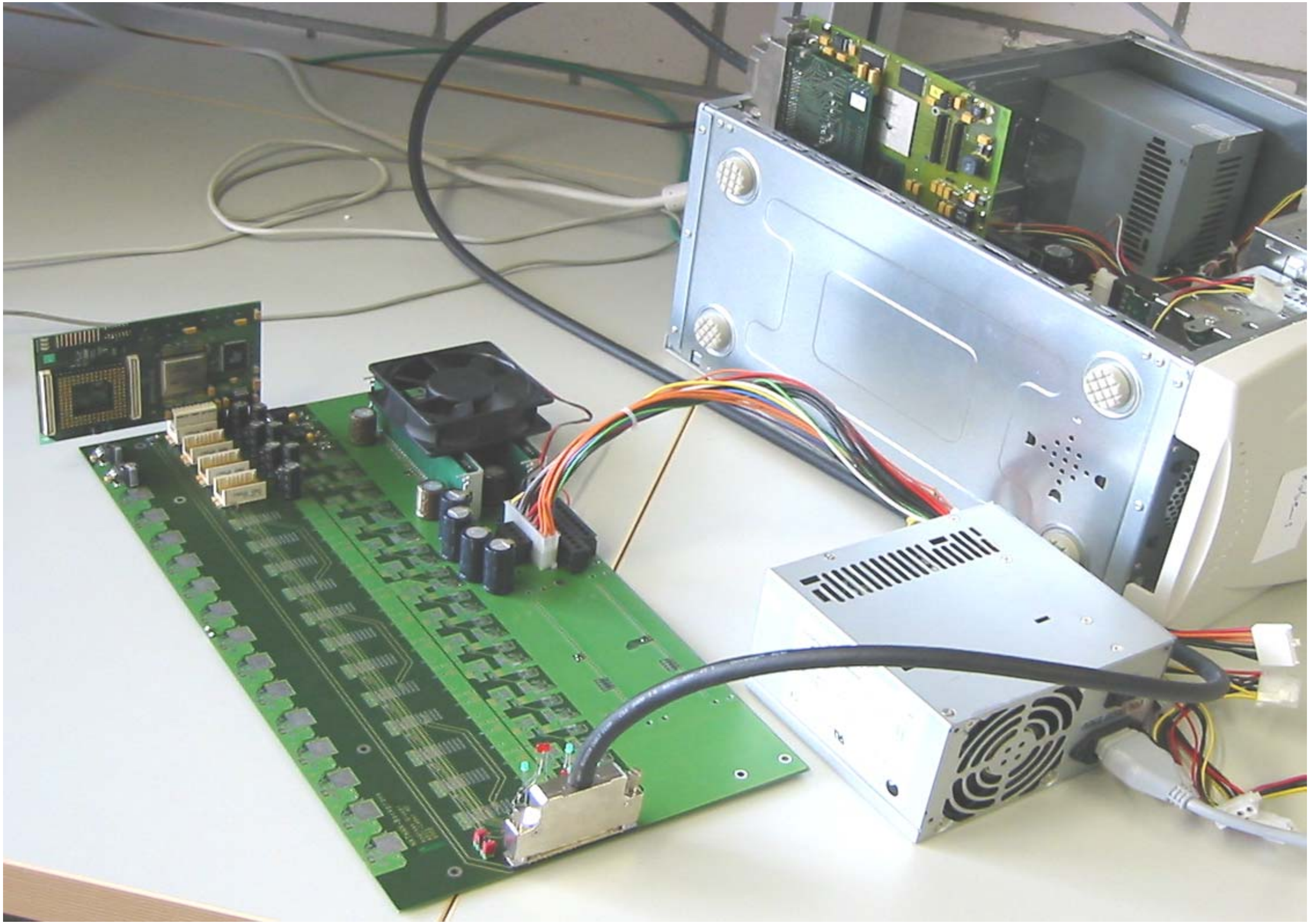


Building Large Neural Networks

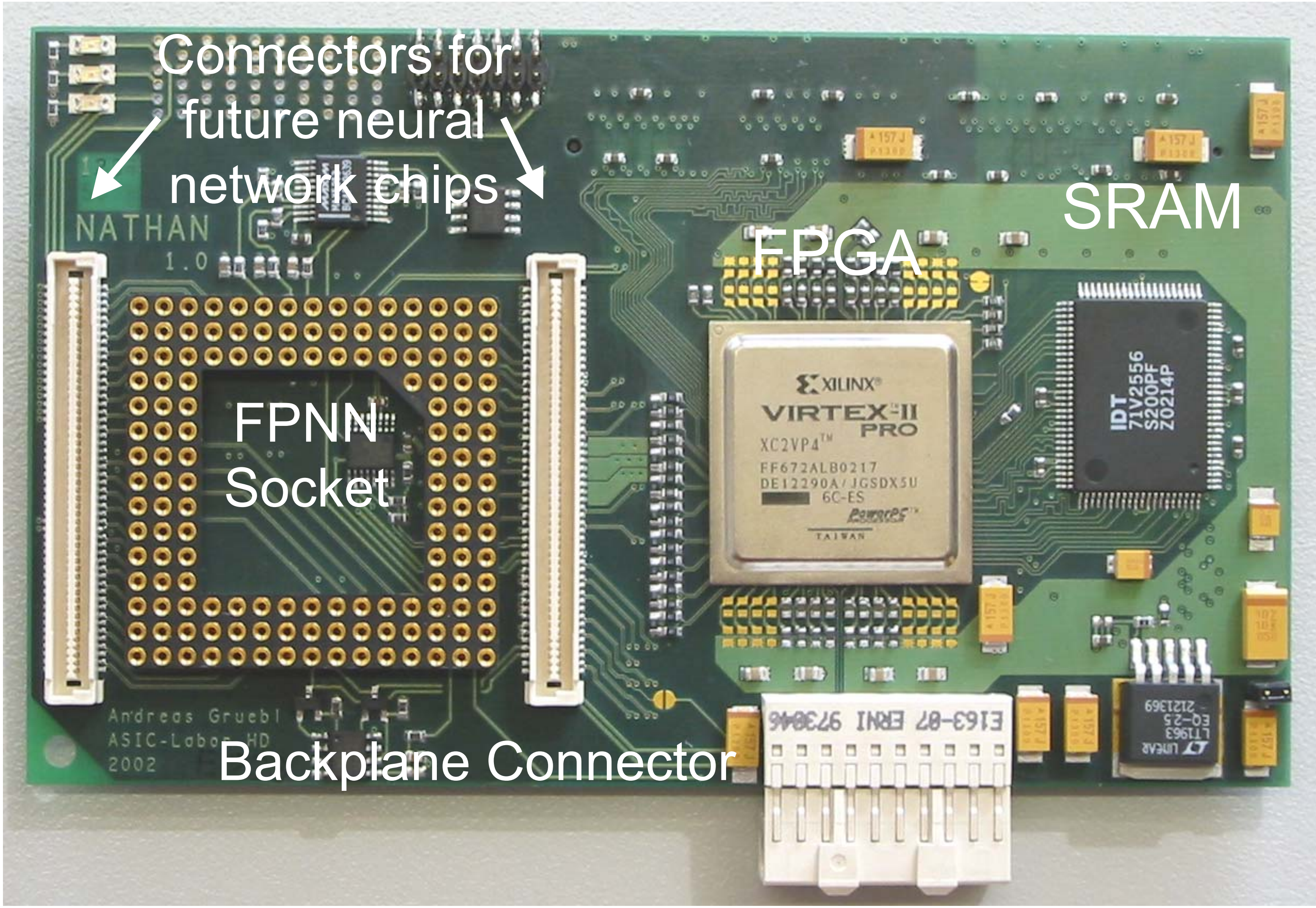


Example of how the FPGAs on the cards can be programmed to use the hardwired Backplane connections to build a large neural network

Picture of Setup under Construction



Distributed Neural Processing : Picture of a network card



Final step towards high complexity FPNN : Implementing Low-Level Biology directly into mixed signal VLSI

Actual Integrate and Fire Model as currently planned

$$c_m \frac{dV}{dt} = g_m(V - E_l) + \sum_k p_k g_k (V - E_x) + \sum_l p_l g_l (V - E_i)$$

sum over excitatory synapses *k*

sum over inhibitory synapses *l*

Synapse Model Simplifications

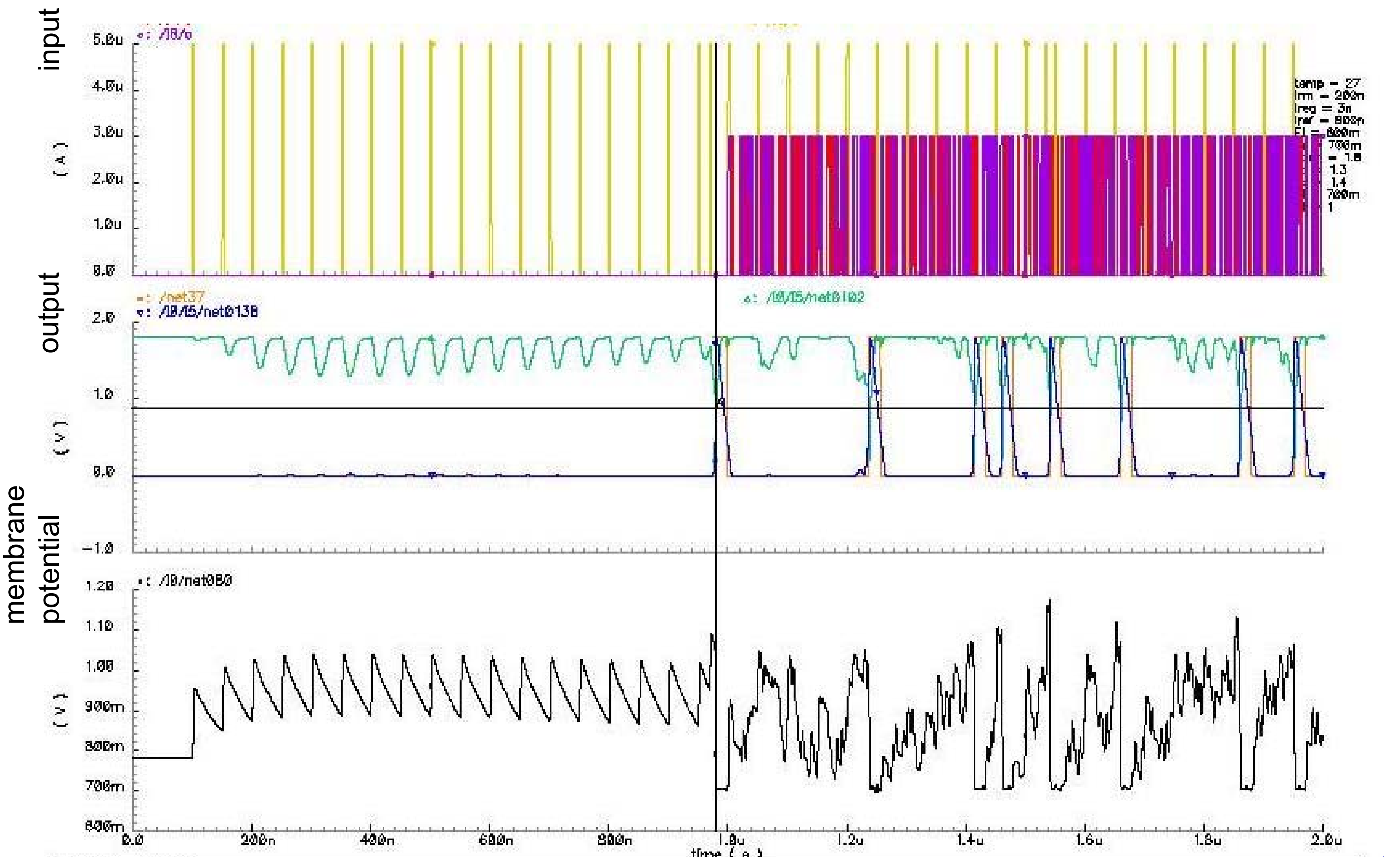
p(t) only zero or one
spike width modulated by *short-term-depression* circuit of presynaptic neuron

g 0 to g_{max} with digital (most likely 4 bit) resolution

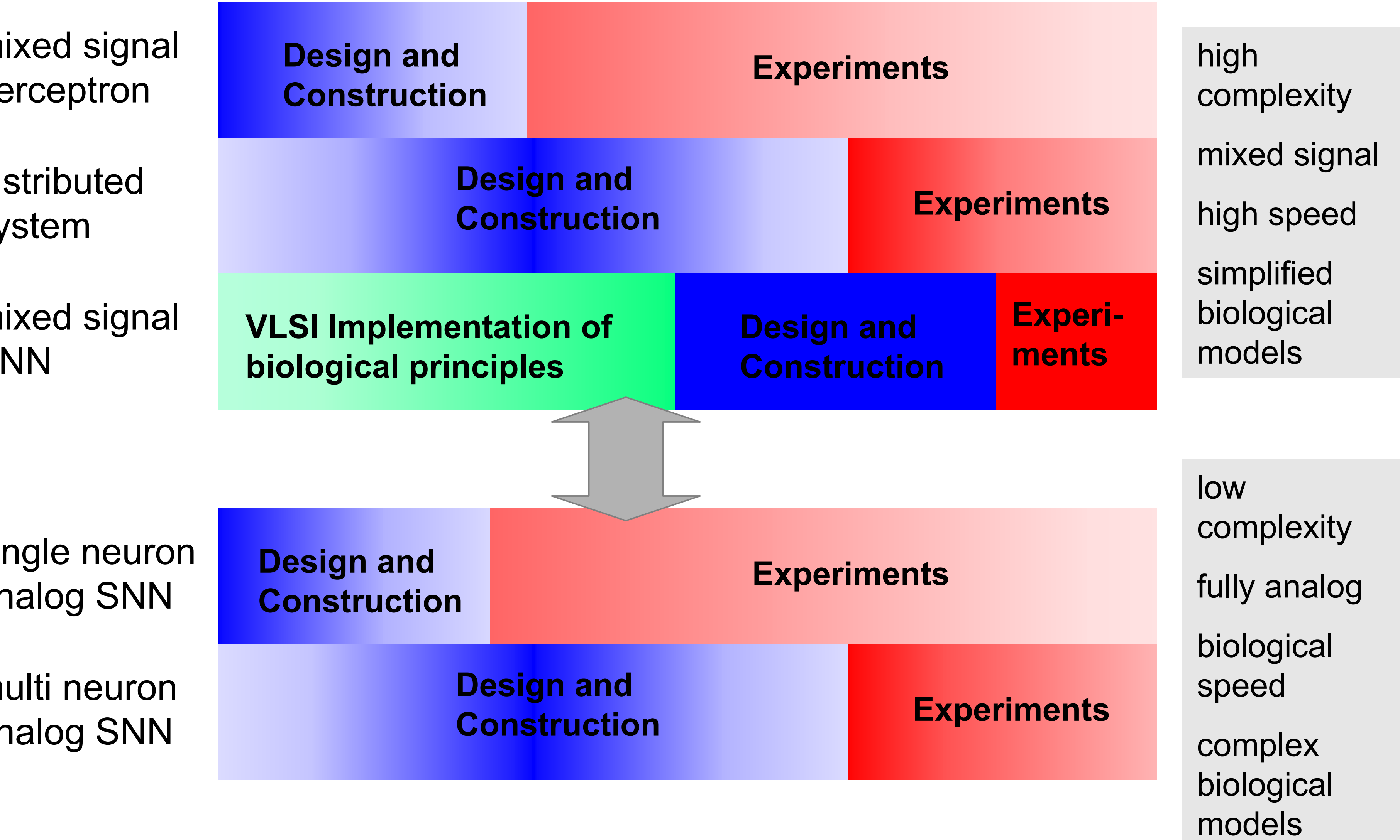
Overview of the SNN Chip - Current Planning

- technology: UMC 0.18μm, 6 metal, 1 poly
- chipsize: 5 x 5 mm^2 (Europractice constraints)
- 4096 neurons, 524288 synapses
- programmable synapse/neuron ratio
- fully analog network core
- continuous time network operation
- scale factor 10^{-6} to 10^{-5} : 1 ns chiptime equals 0.1 to 1 ms realtime
- short-term depression: analog on-chip
- spike-time-dependent-plasticity: digital off-chip
- event (i.e. spike) based external interface
- synaptic weights stored on-chip in SRAM located in the synapses

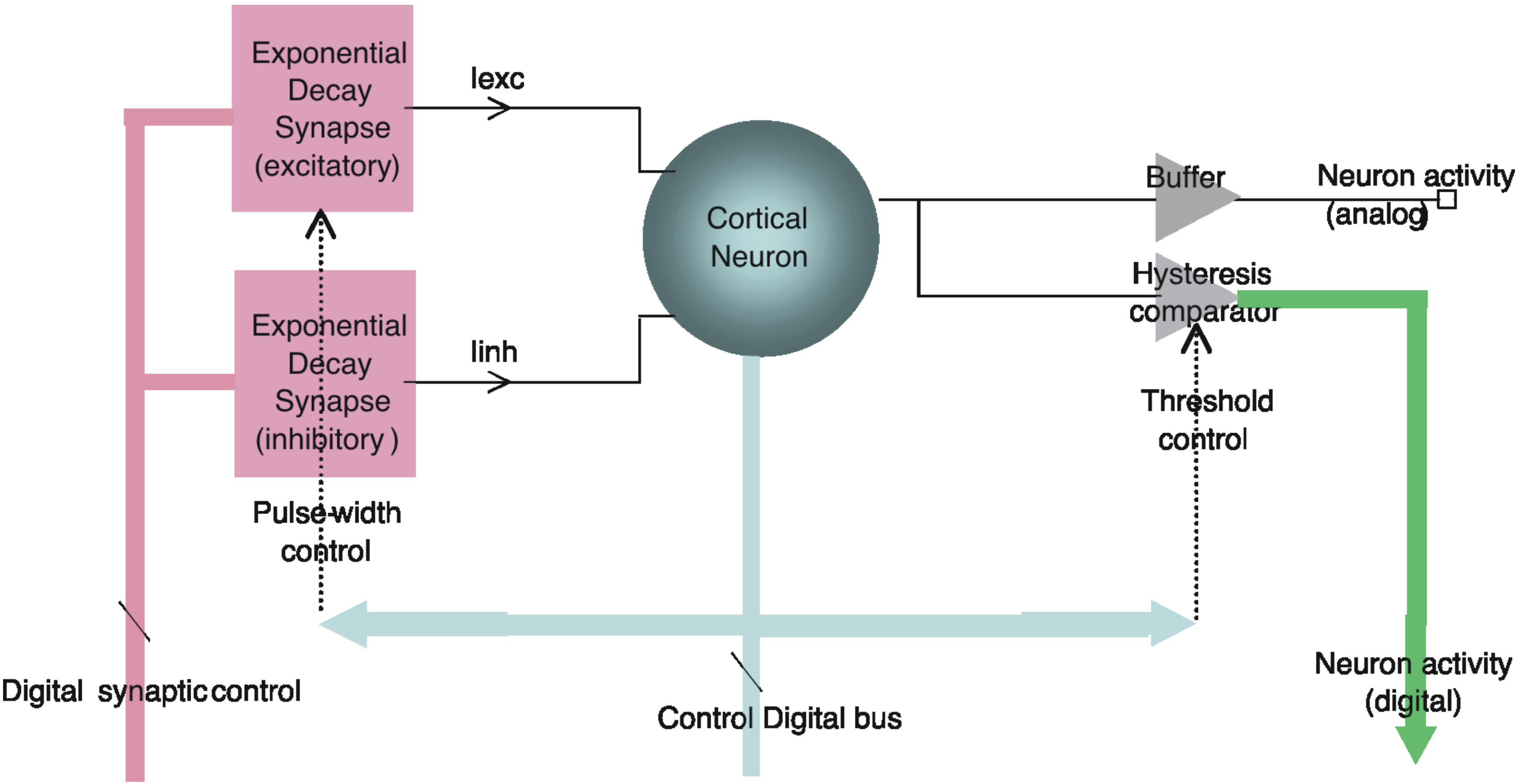
Neuron Response with High Random Background (full circuit simulation)



SenseMaker Roadmap from Biology to VLSI



Analog System : Integrated neural element



Conductance based spiking neuron model (Hodgkin-Huxley formalism)

➡ $C_{mem} \cdot \frac{dV_{mem}}{dt} + i_{Na} + i_K + i_M + \Sigma\, i_{syn} + i_{leak} = 0$

➡ $i_{Na} = g_{Na} \cdot m^3 \cdot h^1 \cdot (V_{mem} - V_{eqNa})$

➡ $i_K = g_K \cdot n^4 \cdot (V_{mem} - V_{eqK})$

➡ $i_M = g_M \cdot mp \quad (V_{mem} - V_{eqM})$

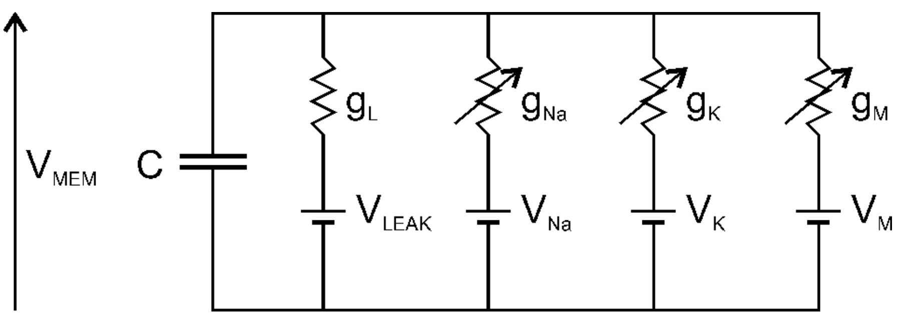
➡ $i_{leak} = g_{leak} \cdot (V_{mem} - V_{leak})$

➡ $i_{syn} = g_{syn} \cdot r \cdot (V_{mem} - V_{syn})$

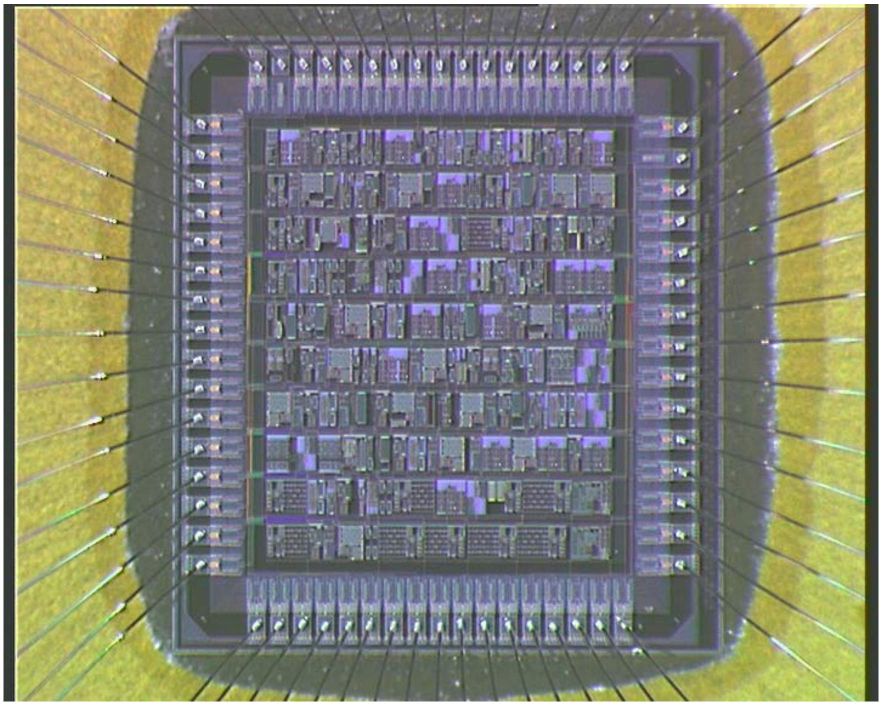
➡ $\tau_m \cdot \frac{dm}{dt} = m_\infty - m$

➡ $m_\infty = \dfrac{1}{1 + \exp\left(\dfrac{offset_m - V_{mem}}{slope_m}\right)}$

Identical for all state variables (m, h, n, mp)



« Trieste » ASIC

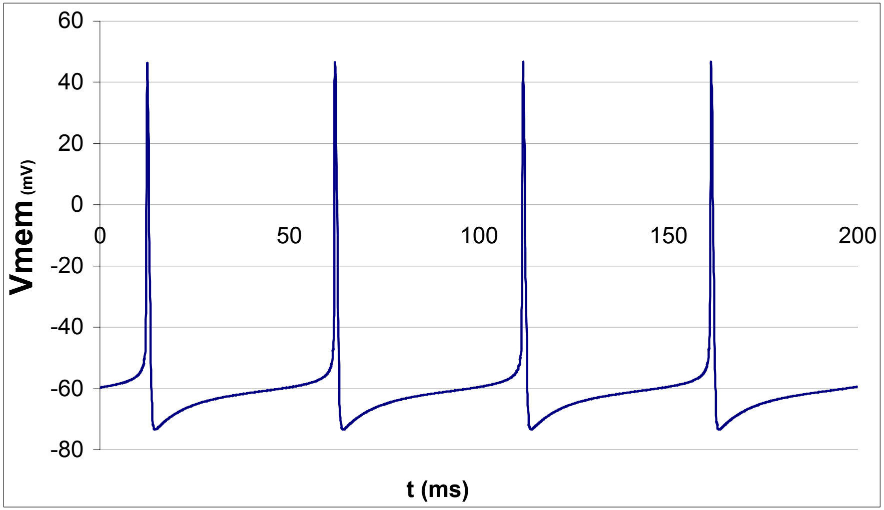


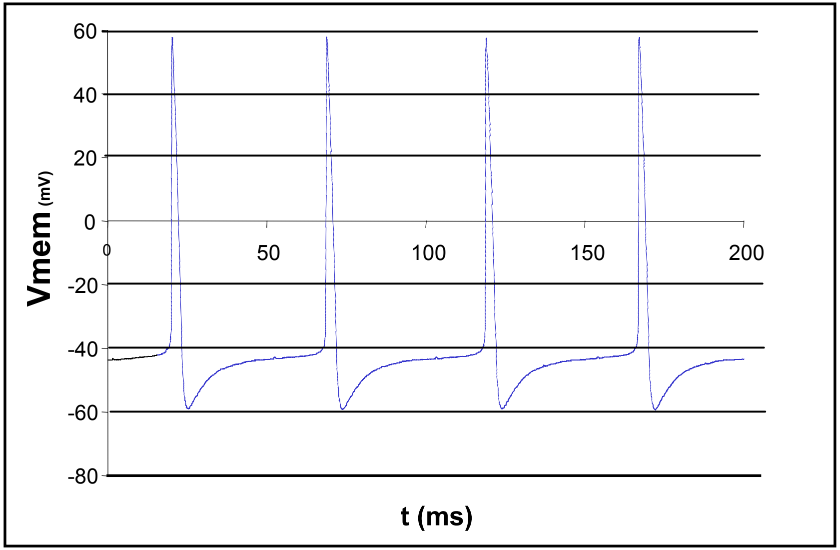
- AMS BiCMOS 0.8mm
- ~ 10000 transistors
- 11 mm^2
- 1 neuron
- 2 synapses (inhib., exci.)
- 68 pads

Hardware/ software simulations

Excitatory neuron

ASIC « Neuron » software simulation



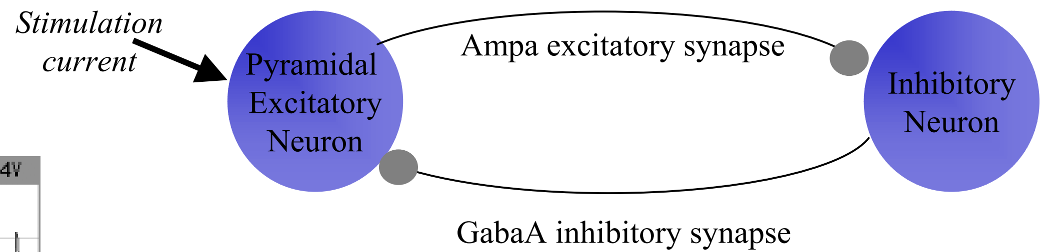


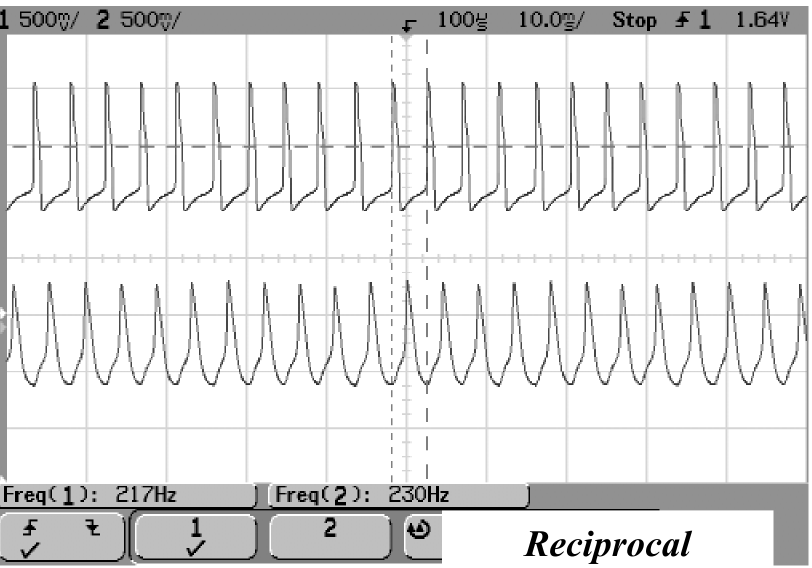
ASIC measurement

time dependence OK

« Trieste » neural network application

2 cortical neurons





Reciprocal synapses (average weights)

Reciprocal synapses (strong weights)

Sensor Development : High dynamic range CMOS sensor

Technical data :

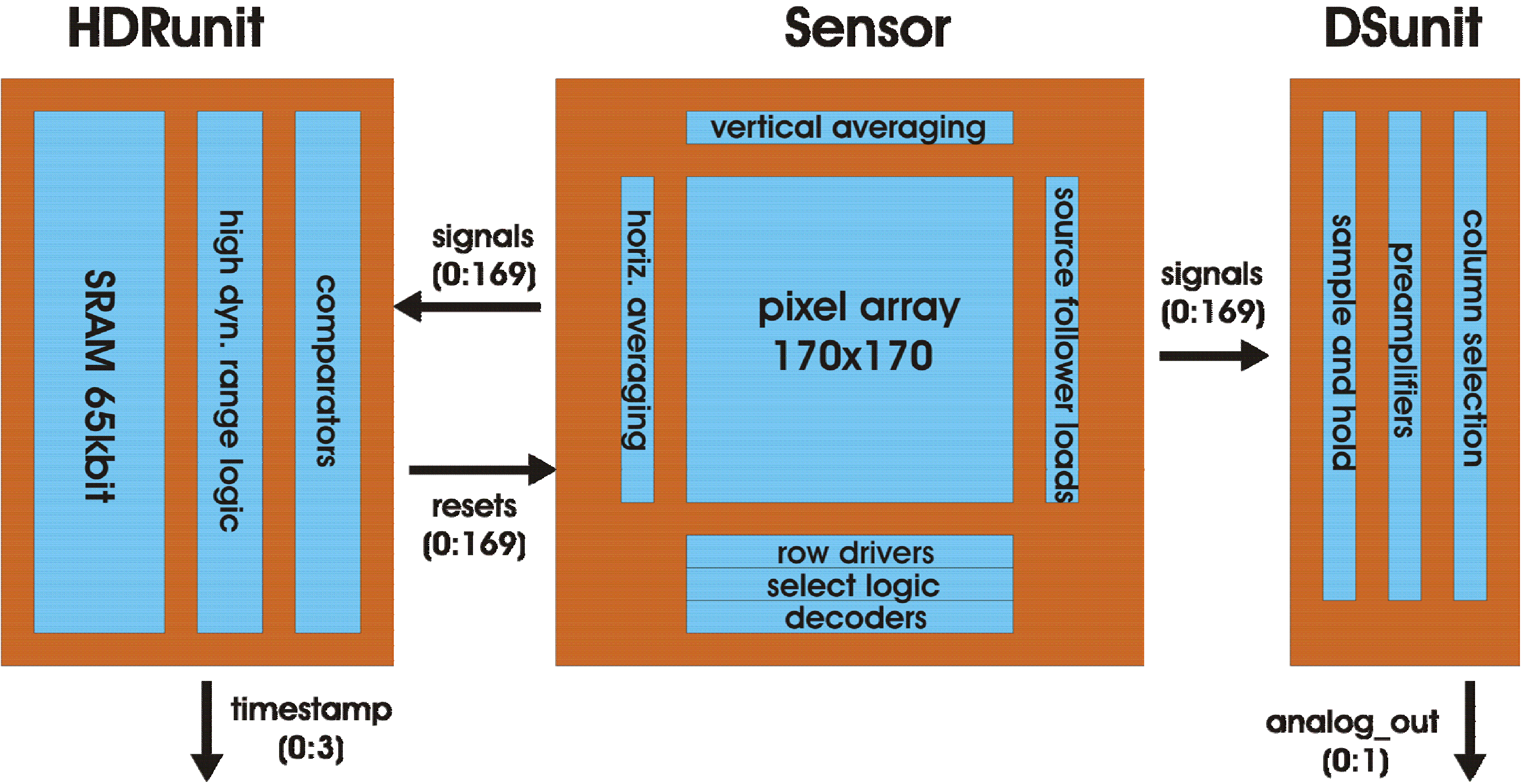
- sensor : n$^+$ - substrate diode
- technology : 0.25mm CMOS (IBM)
- pixel size : 7.5mm x 7.5mm
- die size : 2mm x 4mm
- resolution : 170 x 170 pixels

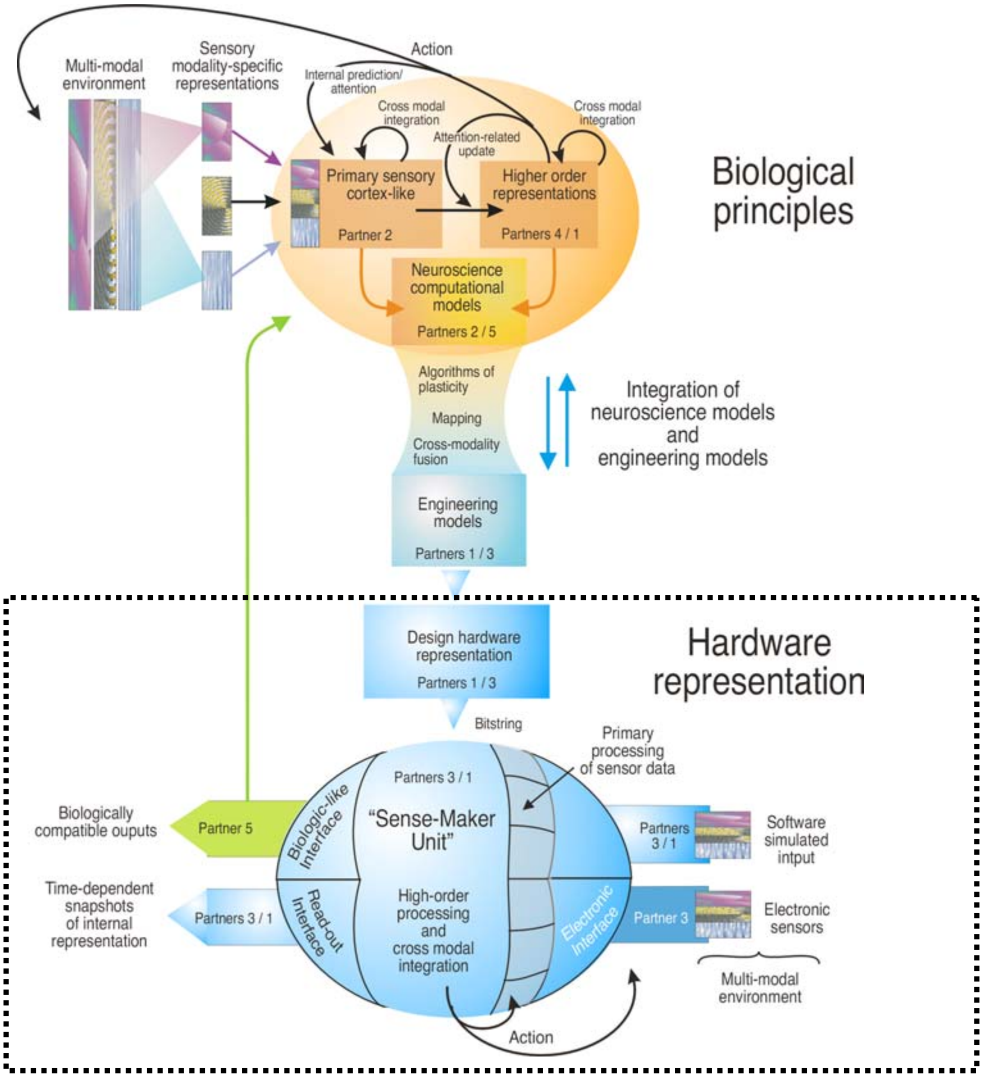
Special features :

- adjustable dynamic range
- averaging of neighbouring pixels (2x2, 4x4, 8x8) (resolution matching)
- movable foveal region of high dynamic range (87x87)

submitted for production in June 03

Sensor Development : Schematics




Action
Multi-modal environment
Sensory modality-specific representations
Internal prediction/ attention
Cross modal integration
Attention-related update
Cross modal integration
Primary sensory cortex-like
Partner 2
Higher order representations
Partners 4 / 1
Neuroscience computational models
Partners 2 / 5
Biological principles
Algorithms of plasticity
Mapping
Cross-modality fusion
Integration of neuroscience models and engineering models
Engineering models
Partners 1 / 3
Design hardware representation
Partners 1 / 3
Hardware representation
Bitstring
Primary processing of sensor data
Partners 3 / 1
"Sense-Maker Unit"
High-order processing and cross modal integration
Biologically compatible ouputs
Partner 5
Biologic-like Interface
Time-dependent snapshots of internal representation
Partners 3 / 1
Read-out Interface
Electronic Interface
Partners 3 / 1
Software simulated intput
Partner 3
Electronic sensors
Multi-modal environment
Action

Design Environment

- High level design environment required with common approach
  - SMU hardware
  - Commercial FPGA devices
- Matlab
  - Standard tool for high level language and graphical exploration of various signal processing and network topologies
  - Routes have now been developed and verified to target both hardware platforms

Matlab to SMU Hardware

Matlab

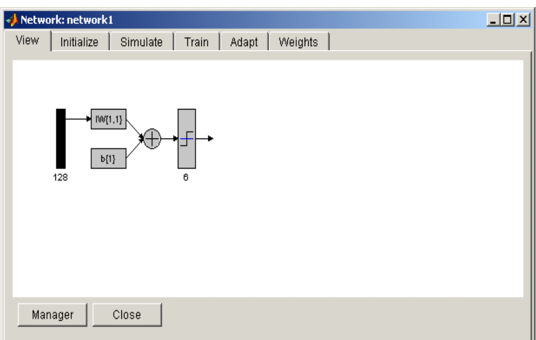
Network Training

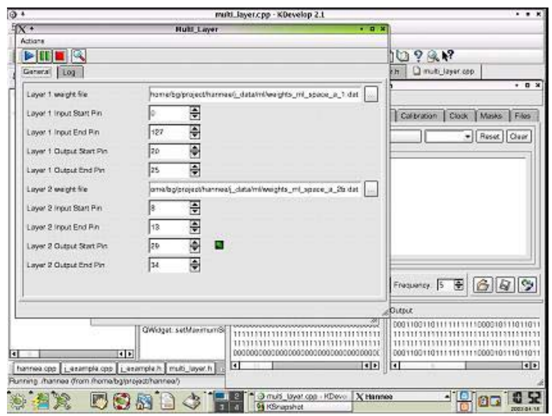
SMU Support Software

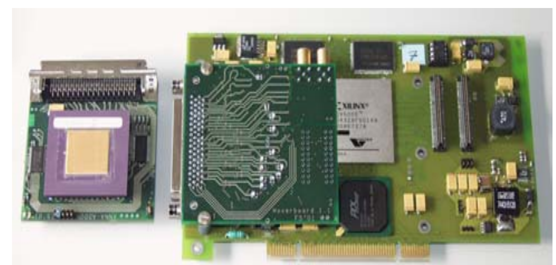
Software Interface

SMU

Hardware Implementation

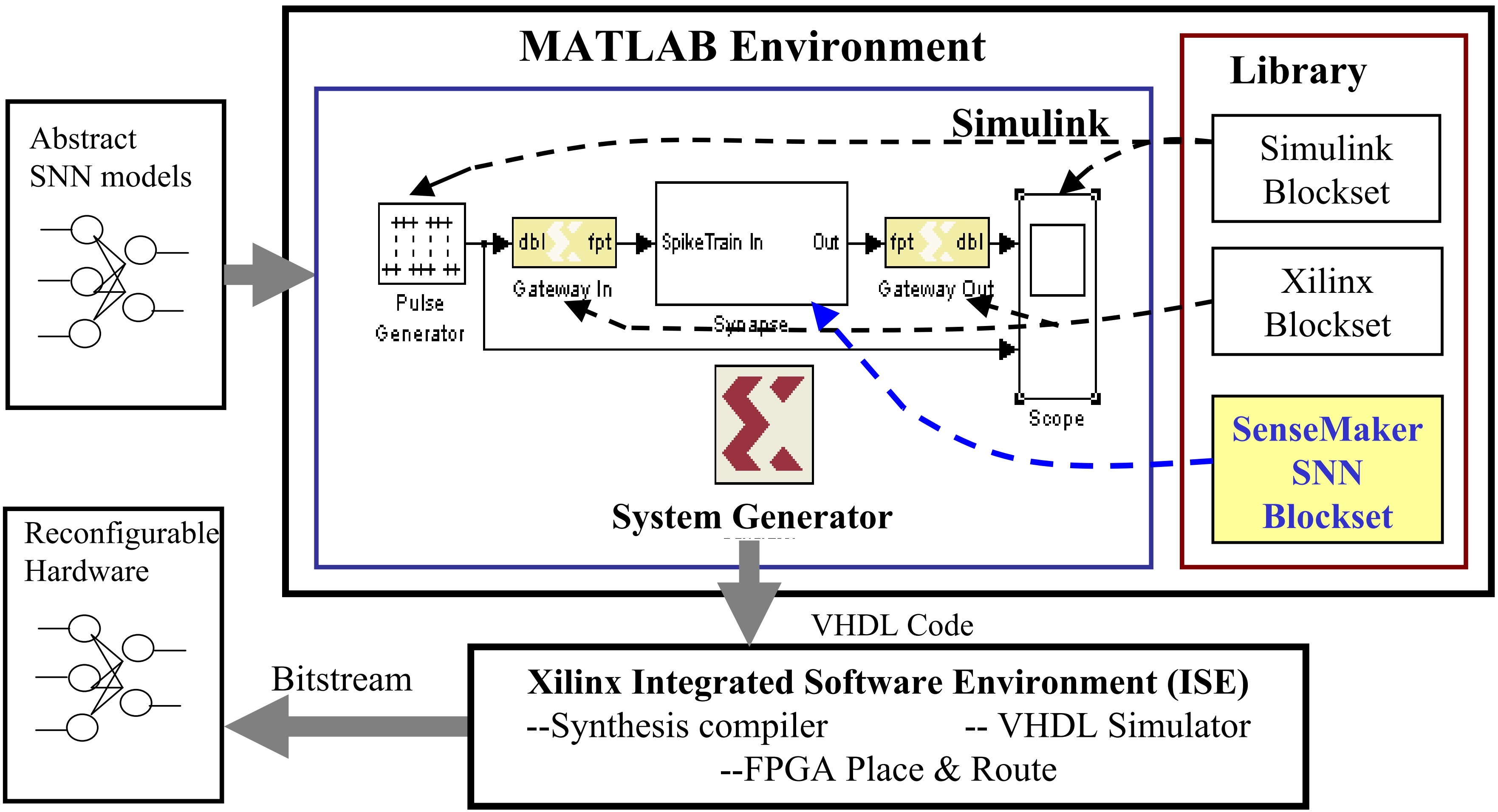






Programmable Hardware Implementation

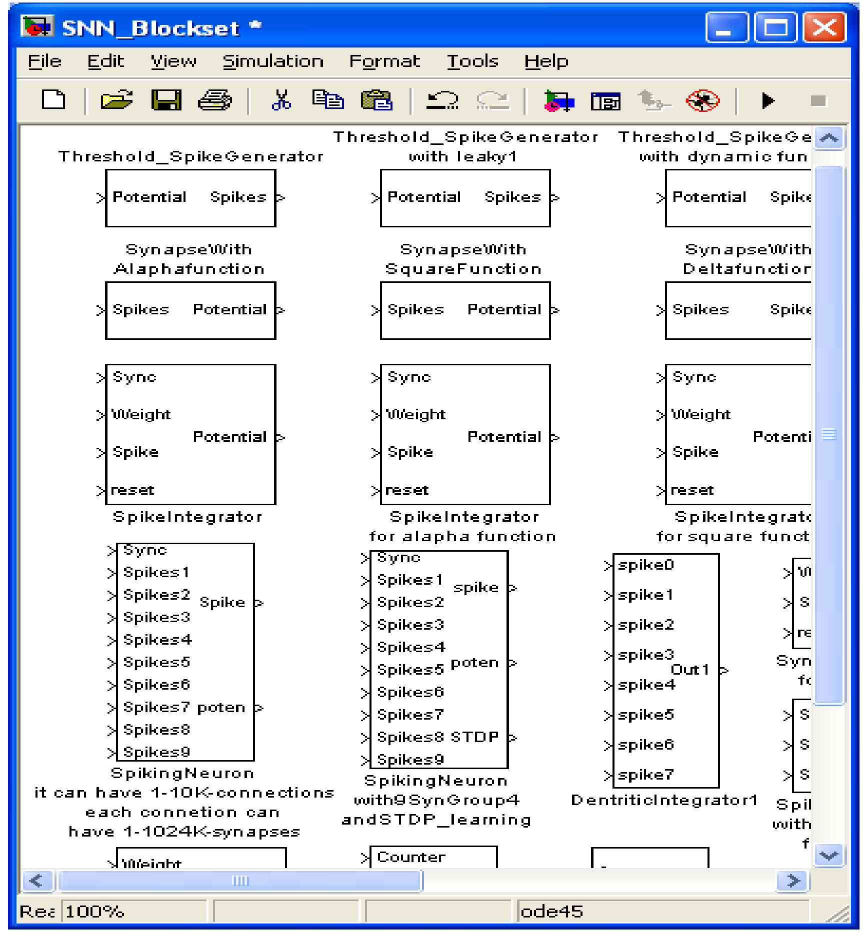
Matlab to Commercial FPGAs



Programmable Hardware Implementation

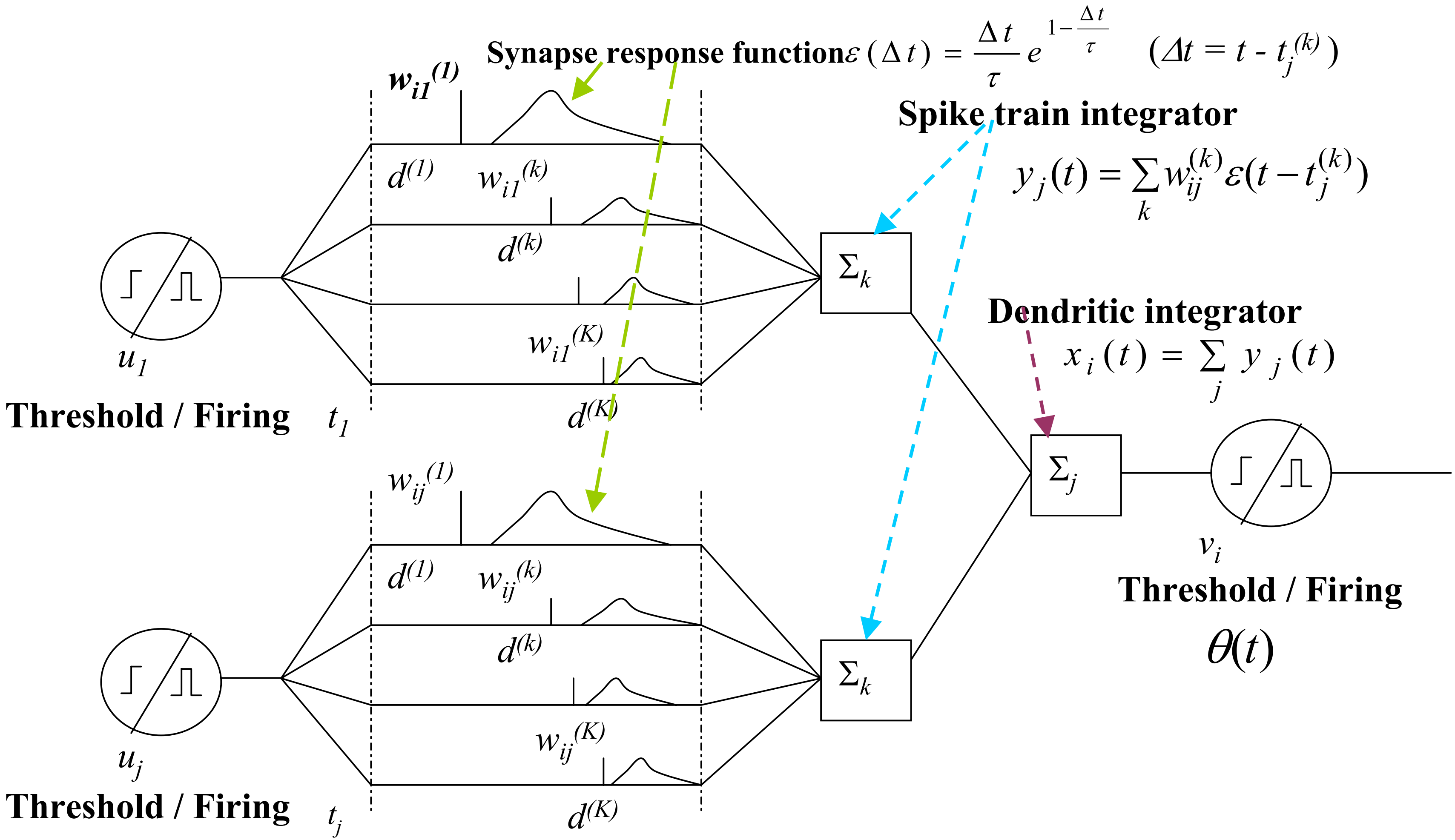
SenseMaker SNN Blockset

- Synapses with programmable weights and synapse response functions
- Dynamic Threshold Functions
- Spike Generators
- Spike Train Integrators
- Dendritic Integrators
- STDP learning circuits
- =>Enables building of Multi-layer SNN



Programmable Hardware Implementation

Generic Computational Representation of a Spiking Neuron



SpikeProp Learning Algorithm

Learning rule for output layer

$$\Delta w_{ij}^{k}(t_j^a) = -\eta \frac{y_i^k(t_j^a).(t_j^d - t_j^a)}{\sum_{i \in \Gamma j} \sum_l w_{ij}^k \frac{\partial y_j^l(t_j^a)}{\partial t_j^a}}$$

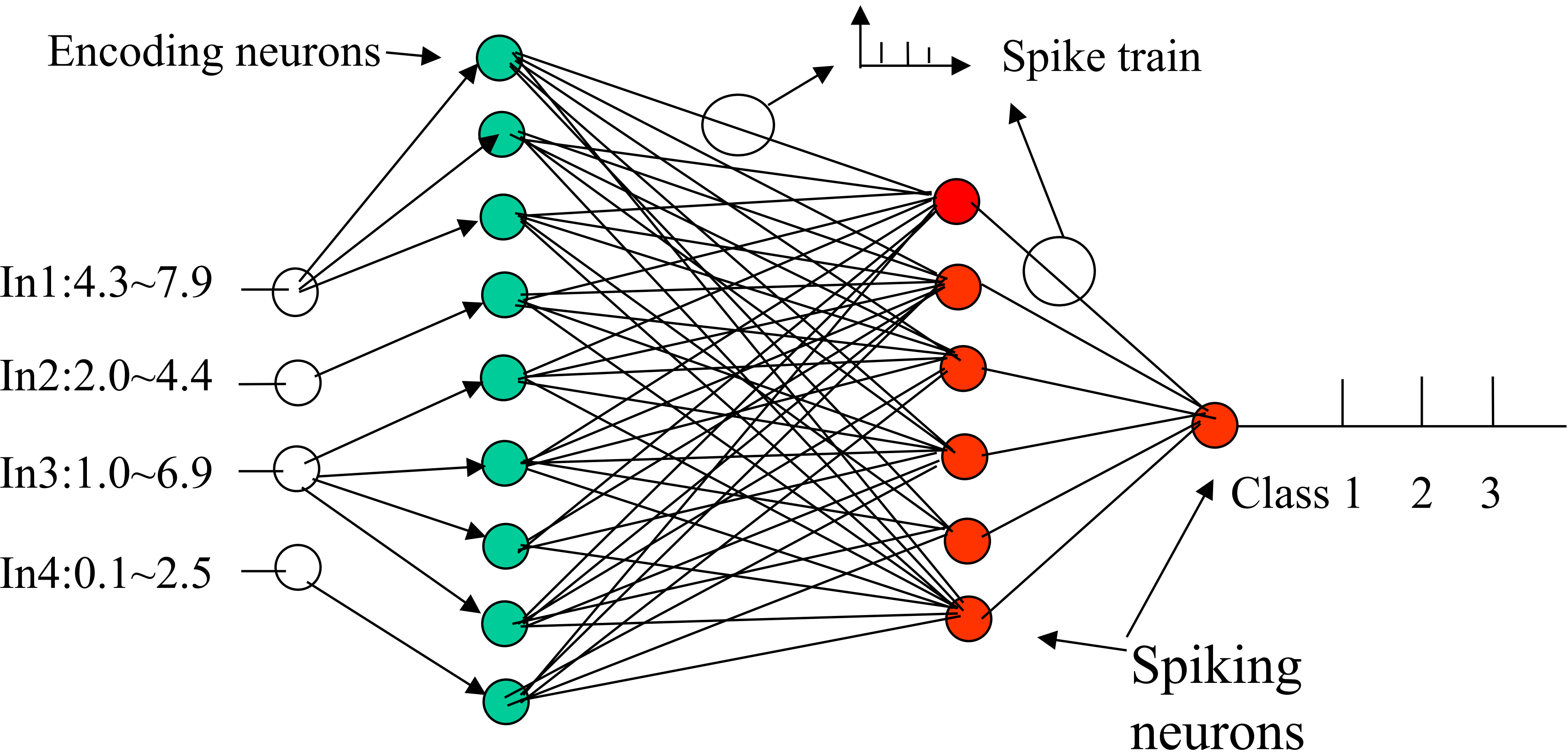
Learning rule for hidden layers

$$\Delta w_{hi}^{k} = -\eta y_h^k(t_i^a)\delta_i = -\eta \frac{y_h^a(t_i^a)\sum_j \{ \delta_j \sum_k w_{ij}^k \frac{\partial y_i^k(t_j^a)}{\partial t_i^a} \}}{\sum_{n \in \Gamma_i} \sum_l w_{ni}^l \frac{\partial y_n^l(t_i^a)}{\partial t_i^a}}$$

where $\delta_i = \frac{\sum_{j \in \Gamma^i} \delta_j \{ \sum_k w_{ij}^k \frac{\partial y_i^k(t_j^a)}{\partial t_i^a} \}}{\sum_{h \in \Gamma_i} \sum_l w_{hi}^l \frac{\partial y_h^l(t_i^a)}{\partial t_i^a}}$

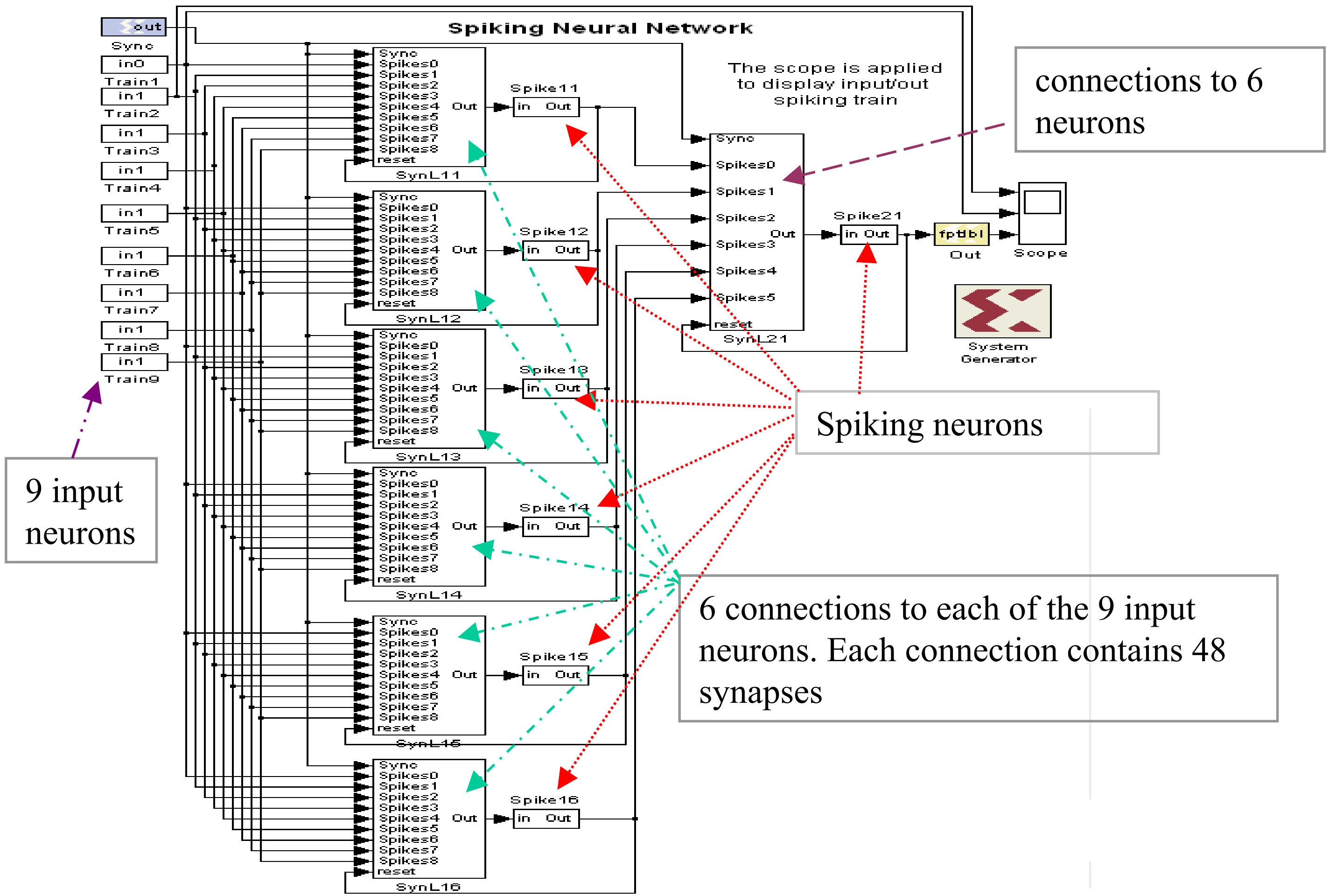
Programmable Hardware Implementation

Example 1: Modified Spikeprop Applied to Iris Data



Programmable Hardware Implementation

Matlab Implementation of SNN



Programmable Hardware Implementation

Synthesis Report

Device utilization summary(This report is generated by Xilinx ISE 5.1i)
Selected Device :The Virtex-II XCV6000

Number of Slices:	8454 out of 33792	25%
Number of Slice Flip Flops:	5267 out of 67584	7%
Number of 4 input LUTs:	10310 out of 67584	15%
Number of BRAMs:	60 out of 144	41%
Number of MULT18X18s:	7 out of 144	4%
Number of GCLKs:	1 out of 16	6%

Timing Summary:Speed Grade: -6
Minimum period: 23.092ns (Maximum Frequency: 43.305MHz)
Minimum input arrival time before clock: 2.588ns
Maximum output required time after clock: 14.229ns

Potential Network Size

The Virtex-II XCV6000 provides 33,792 slices. An approximate analysis indicates that the chip can thus support a network with dimensions of the order of 10^2 neurons and 10^4 synapses.

System Architecture

- Three phases of implementation SMS1, SMS2, SMS3
- SMS1
  - Based on SMU1 PCI card
  - McCulloch Pitts neurons
  - Proof of concept
- SMS2
  - Further development of McCulloch Pitts neurons on SMU2
  - Development of digital SNNs on commercial FPGA systems
  - Essential intermediate step towards large spiking neuron assembly
  - Exploration of high level controller architecture on commercial FPGAs
  - Implementation of two ring problem
  - Investigation of various training algorithms

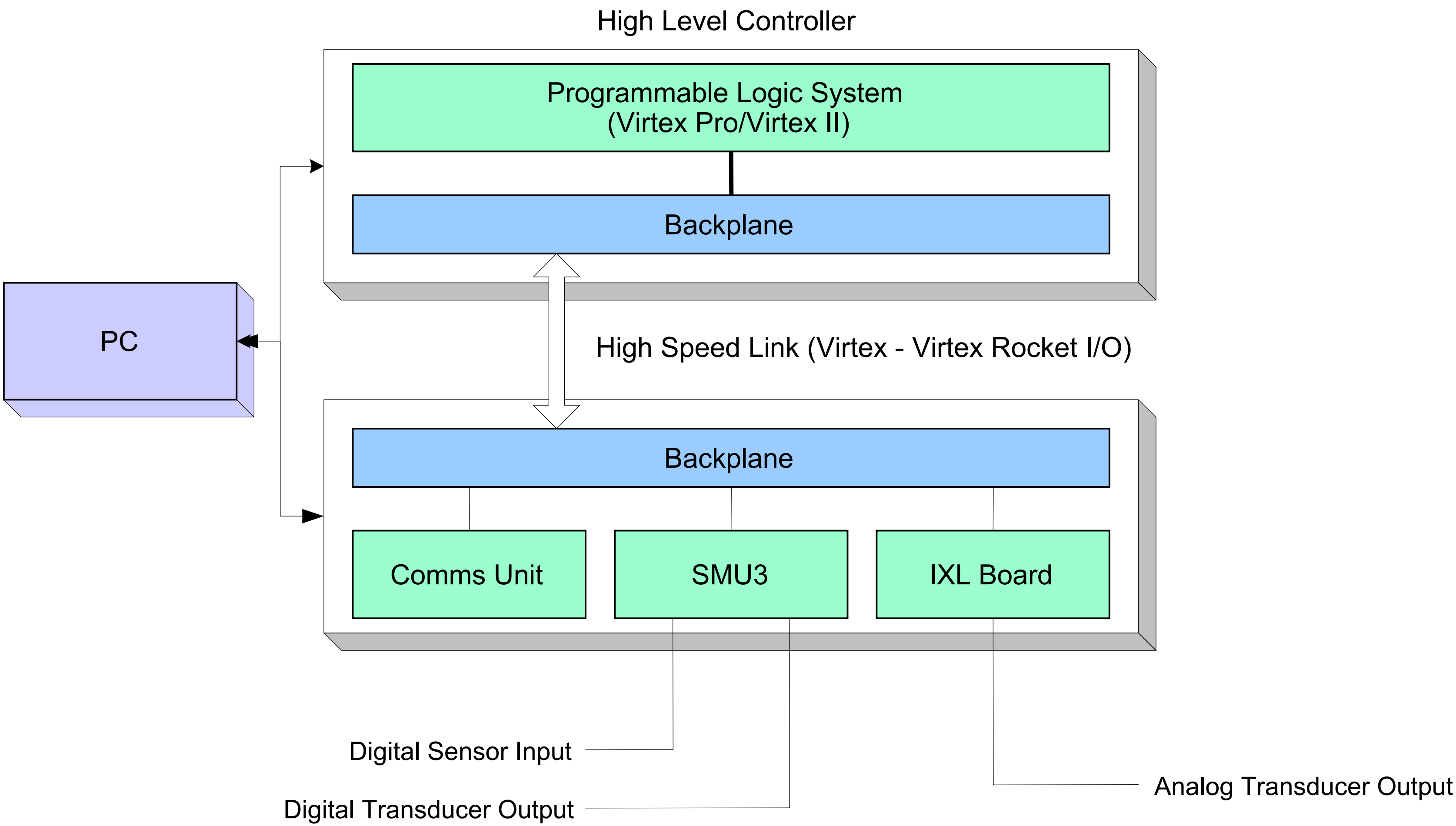
System Architecture

SMS3

- SMU3, commercial FPGA & IXL analog board
- Spiking neuron based
- Communication via high speed serial links
- IXL analog output board used to generate biologically compatible signals for interfacing to hybrid silicon –biological systems

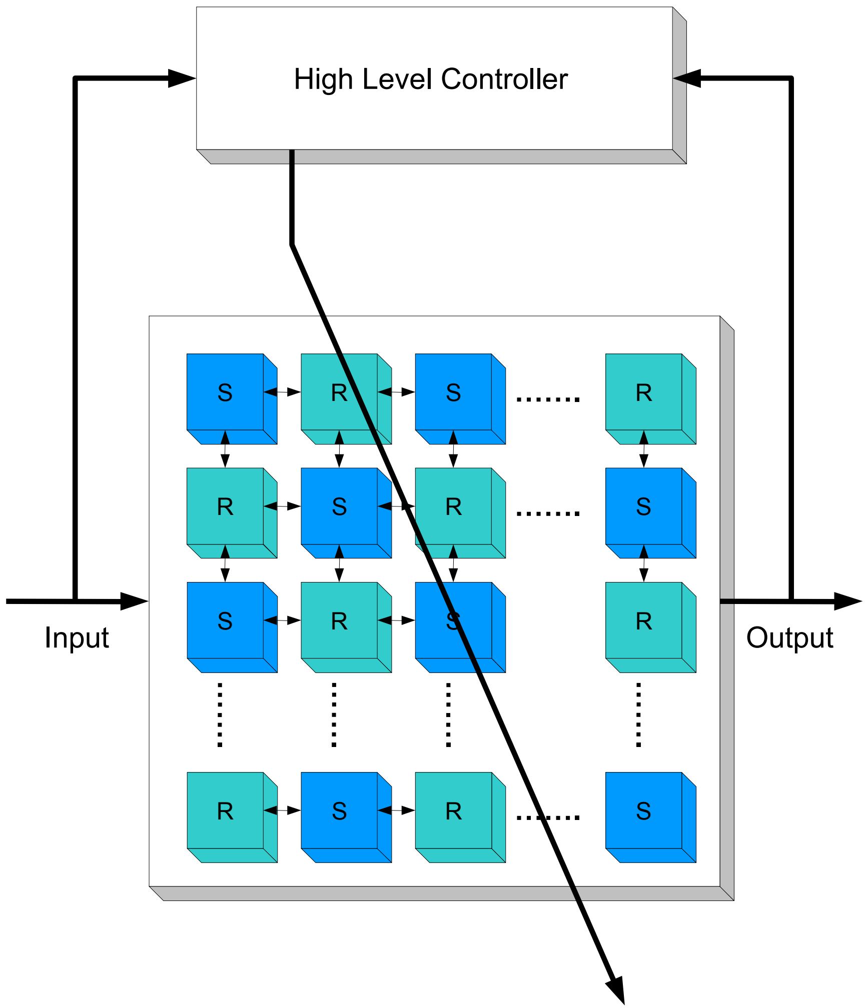
System Architecture

SMS3



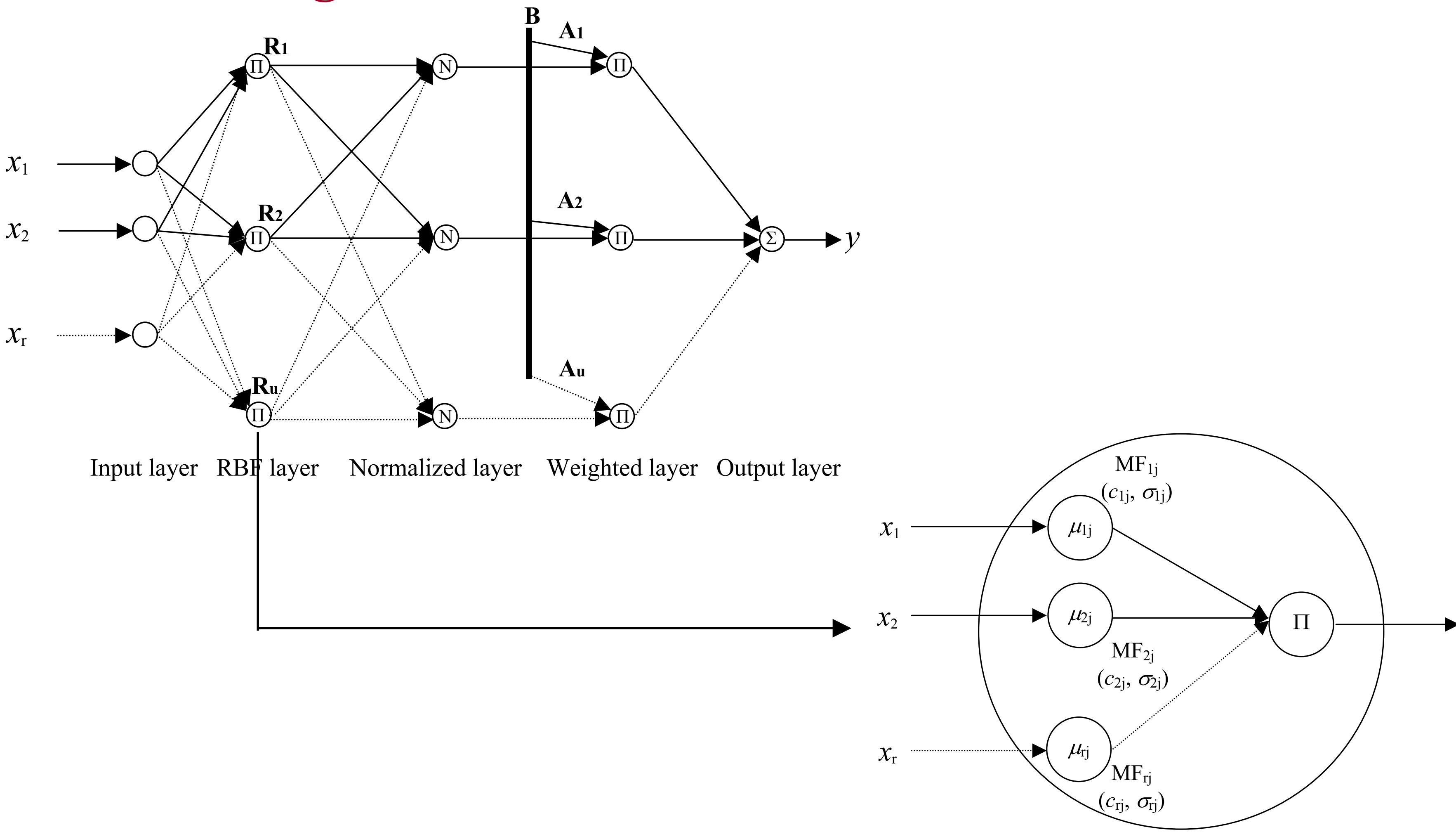
System Architecture

High Level Controller



System Architecture

SOFNN High Level Controller



Fuzzy Neural High Level Controller

- Adding a neuron- rule based approach
  - Two criteria to judge re adding a RBF neuron
  - error criterion considers the generalization performance of the overall network
  - if-part criterion evaluates whether existing fuzzy rules or RBF neurons can cover and cluster the input vector suitably.
- Pruning a neuron
  - Combines the Optimal Brain Surgeon approach with RLS algorithm
  - Deletes the least important neuron if the performance of the network is within the desired tolerance limit

Presentation Summary

- Presentation has covered:
  - Low-level biological principles
  - Design of a mixed analog/digital simulator for biologically-realistic neuron networks
  - VLSI based implementations of neural network systems and sensors
  - Programmable hardware implementations
  - System architecture