

From Cellular Nonlinear Networks to

***Virtual and Physical Cellular
Machines***

Tamás Roska

Hungarian Academy of Sciences and the
Pázmány University, Budapest

MIND Workshop, Notre Dame, August 18,
2009

The technology trend

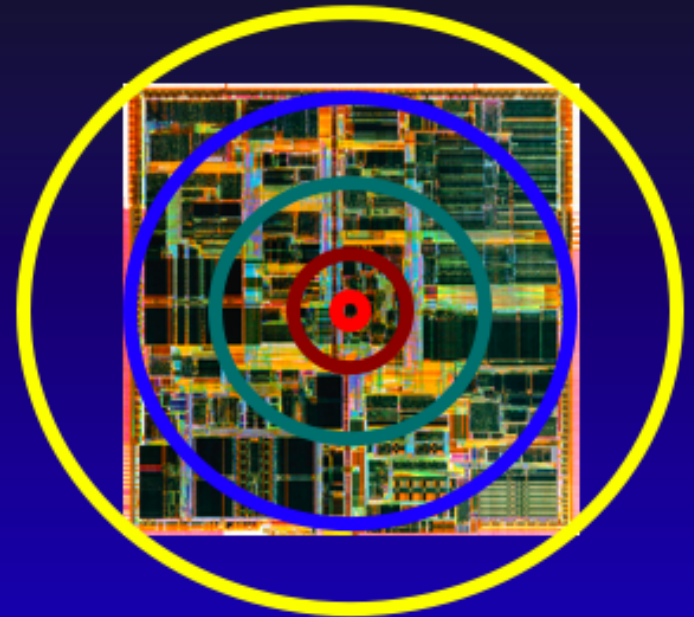
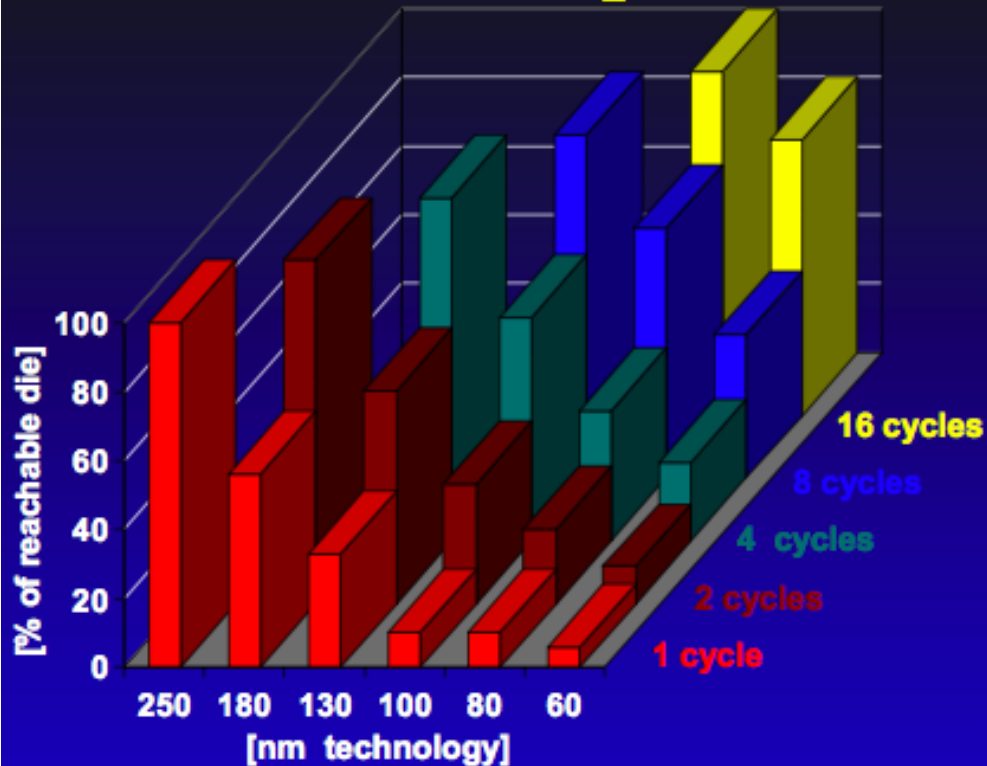
- Since ~2001 the number of processors/chip grows with limited clock speed and power dissipation
- At 65nm a few hundred arithmetic (DSP) and logic (look-up, etc.) processors per chip (max 2 Billion tr.)
- 25 k mixed mode processors and sensors on a cellular visual microprocessor at 180 nm
- 45 nm about 1 million 8 bit microprocessors could be placed (~5 Billion transistors) ...Why not?
- Wire delay is bigger than gate delay, hence communication speed is limited (synchron radius)

Hence

- The ***precedence of Locality*** ~ i.e. Cellular (mainly locally and sparsely globally connected architectures) is a must for high computing power

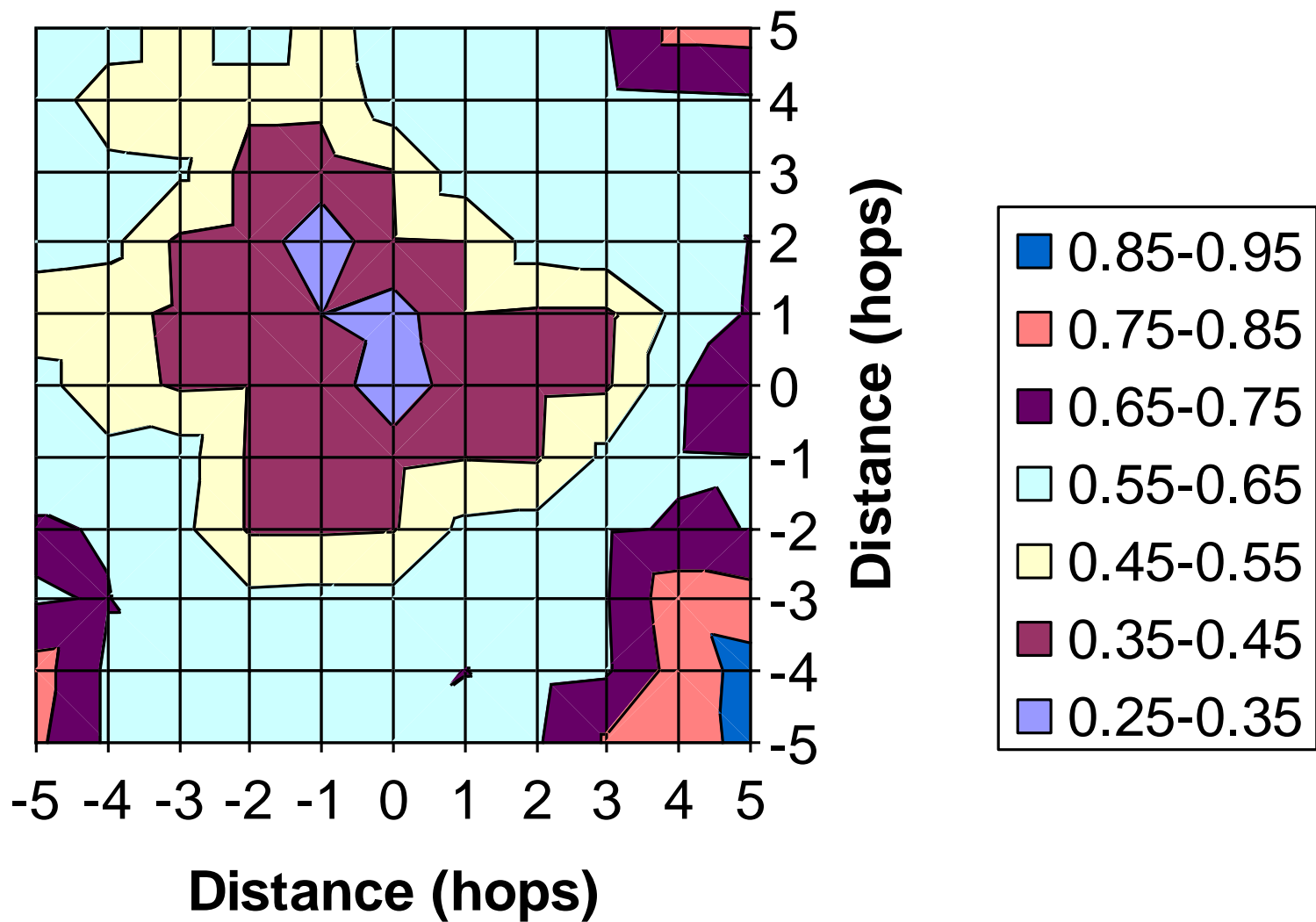
...but it also doesn't scale terrible

On-chip interconnect latency



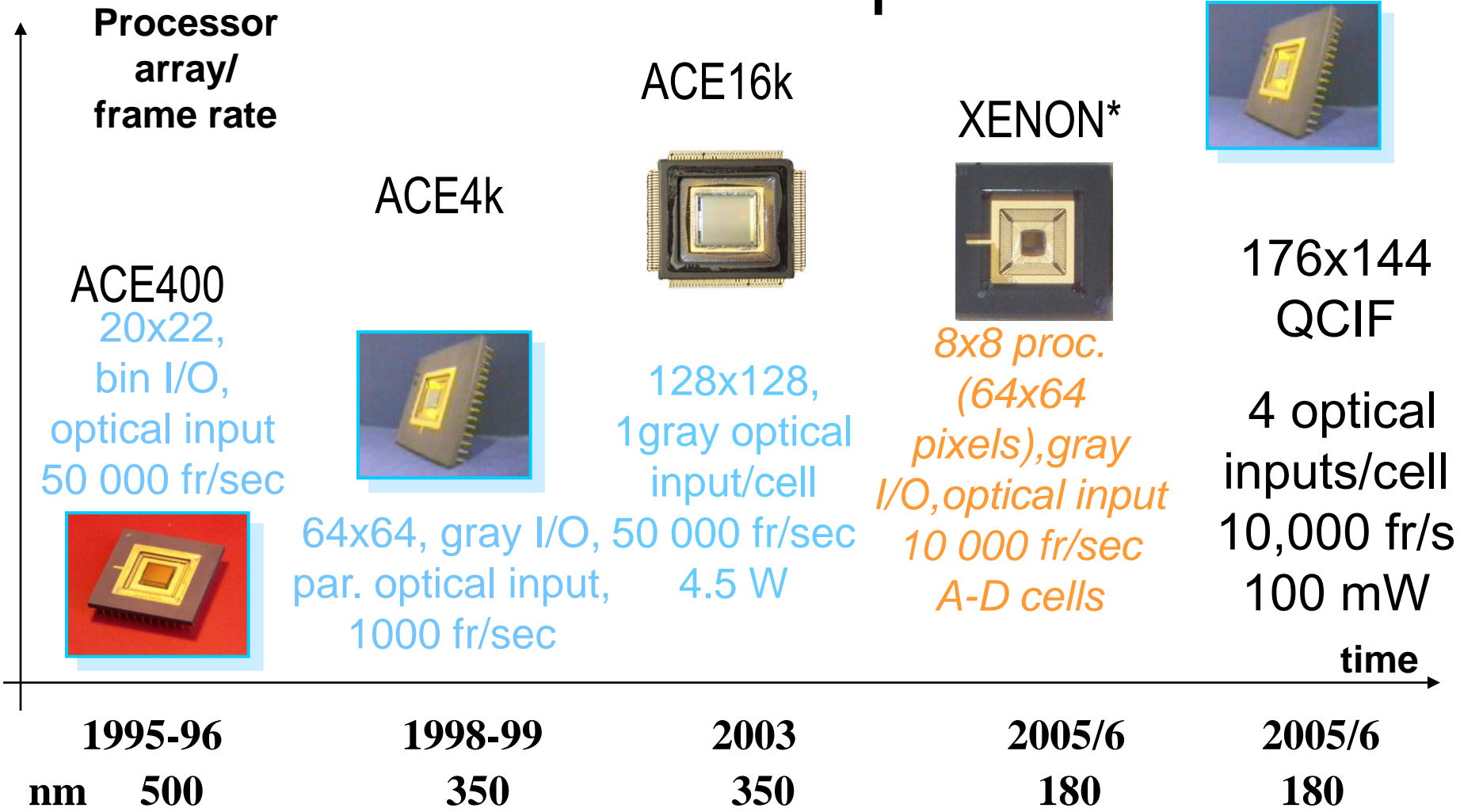
- *“For a 60-nanometer process a signal can reach only 5% of the die’s length in a clock cycle”* [D. Matzke (Texas Instruments), IEEE Computer Sept. 97]
- Shift from **function-centric** to **communication-centric** design

Wire delay distribution (ns)



Cellular Visual Microprocessor

Roadmap



The ACE family was designed at IMSE-CNM and the Q-Eye/Eye-Ris family at AnaFocus Ltd., Seville. The XENON chip was designed by MTA-SZTAKI, Budapest extended by Eutecus Inc, Berkeley, CA

Combining the best of the two computing modes

First prize and Product of the year at Vision 2003, Stuttgart

Bi-i: a standalone visual system

Version 1.1:

- **Standalone Compact**

Embedded 128x128 ACE16k* chip (1 or 2)

Above 20 000 Fps

Embedded 250MHz DSP/600 MHz

Embedded 1.3M CMOS imager with ROI

Ethernet 100MBit/s

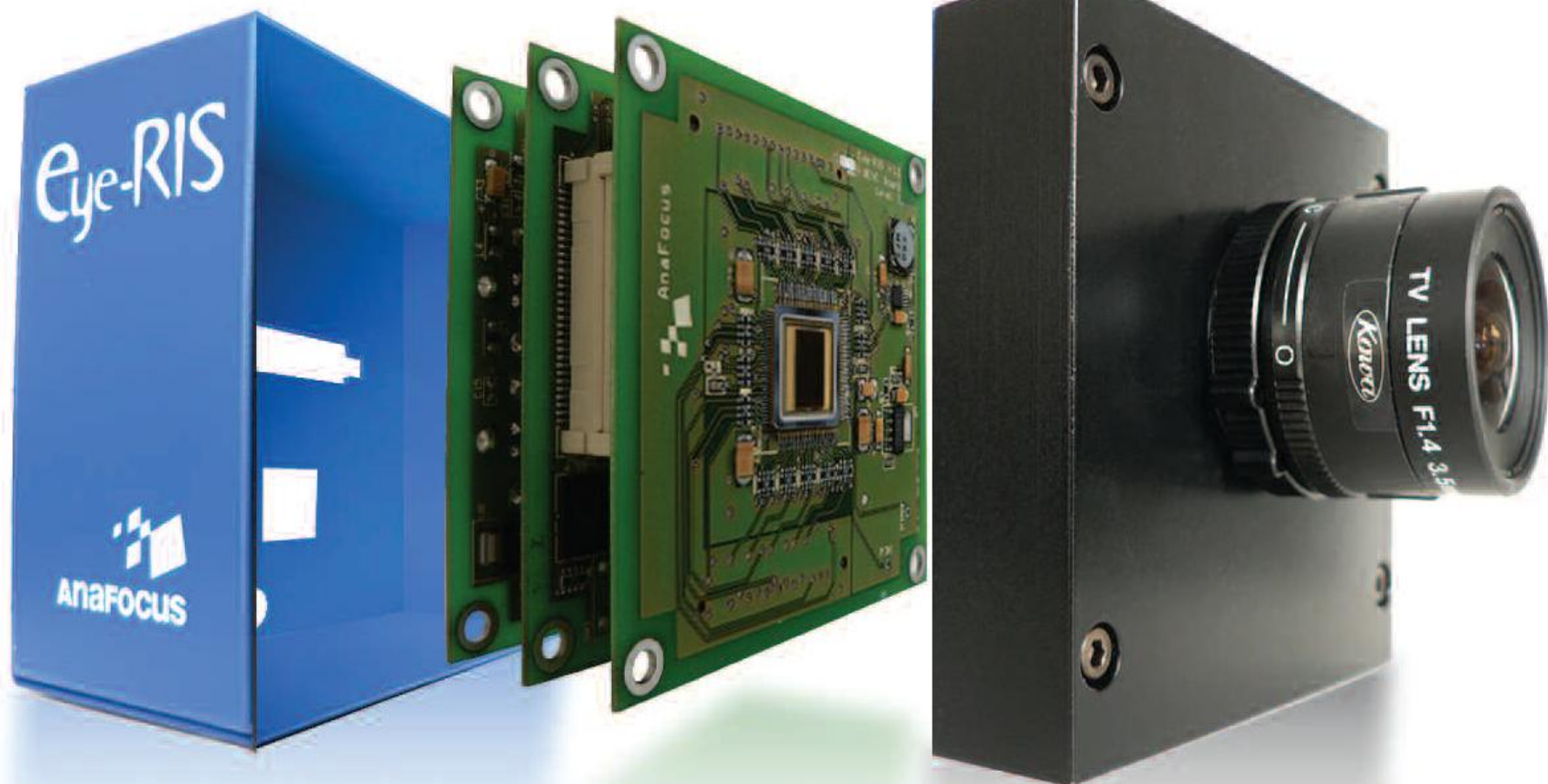
USB

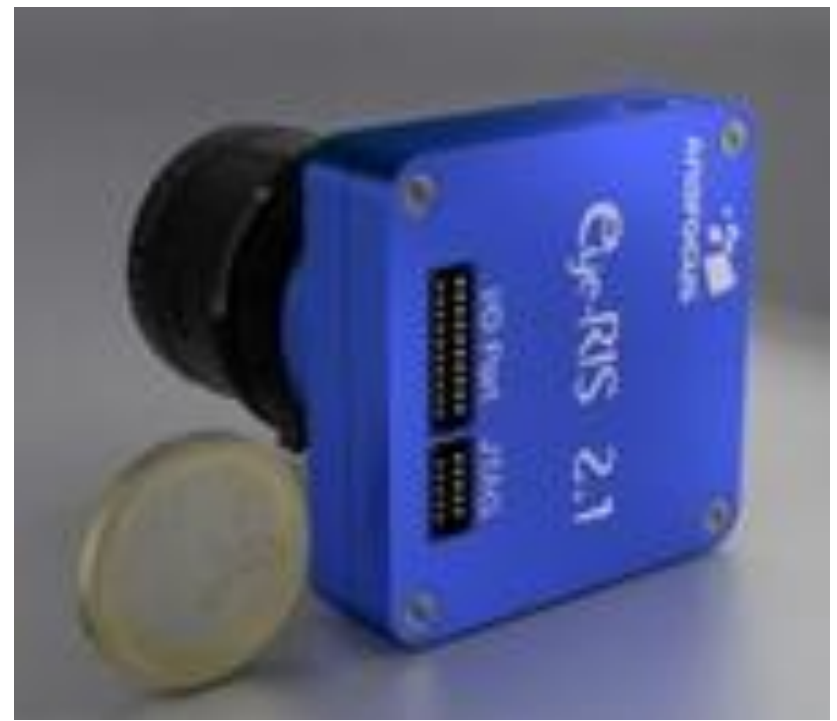


Á. Zarándy, Cs. Rekeczky, et. al., MTA SZTAKI, Budapest.

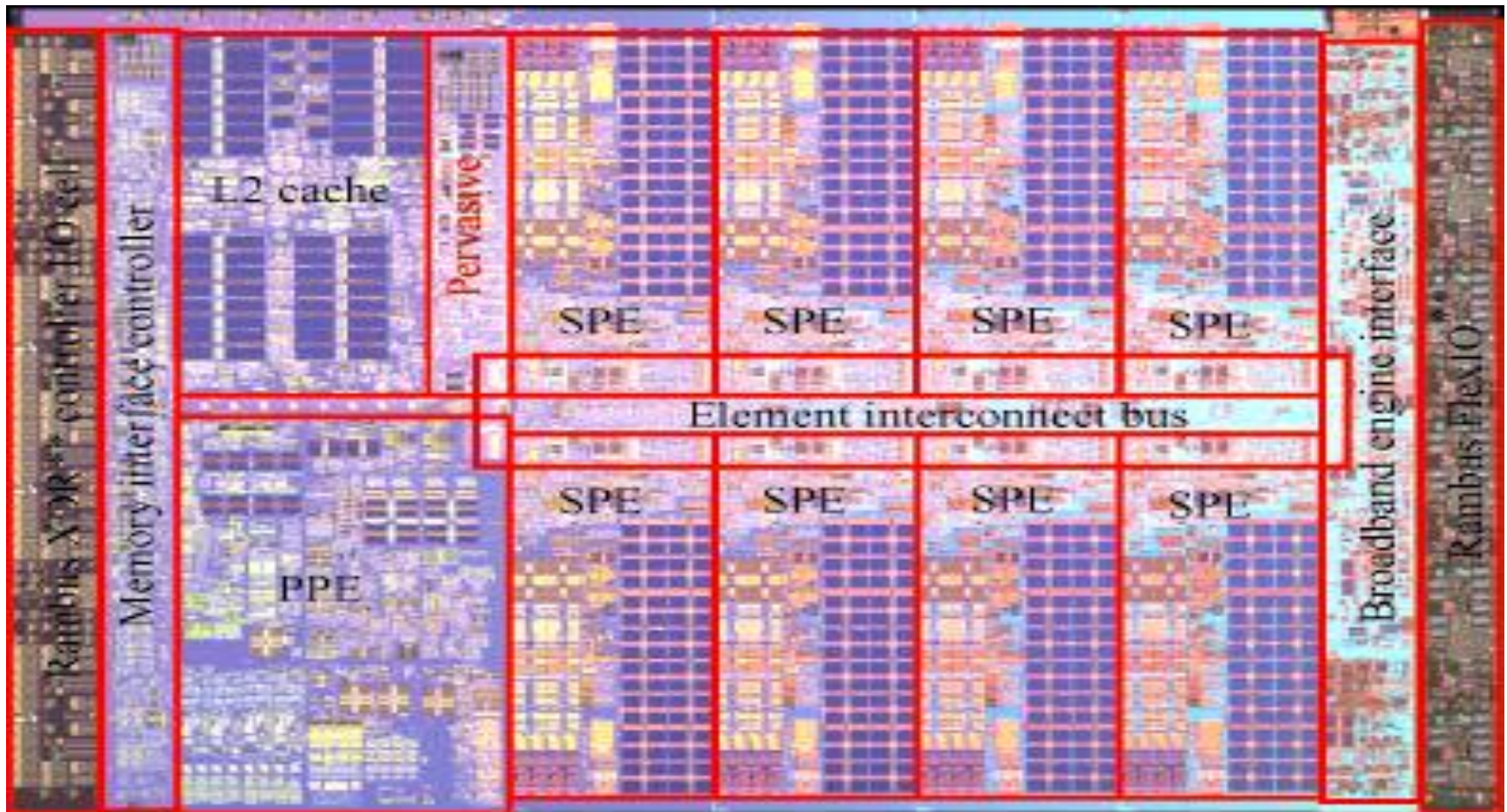
*ACE16k chip: IMSE-CNM/AnaFocus Ltd., Seville Spain

AnaFocus Eye-RIS v.1.2





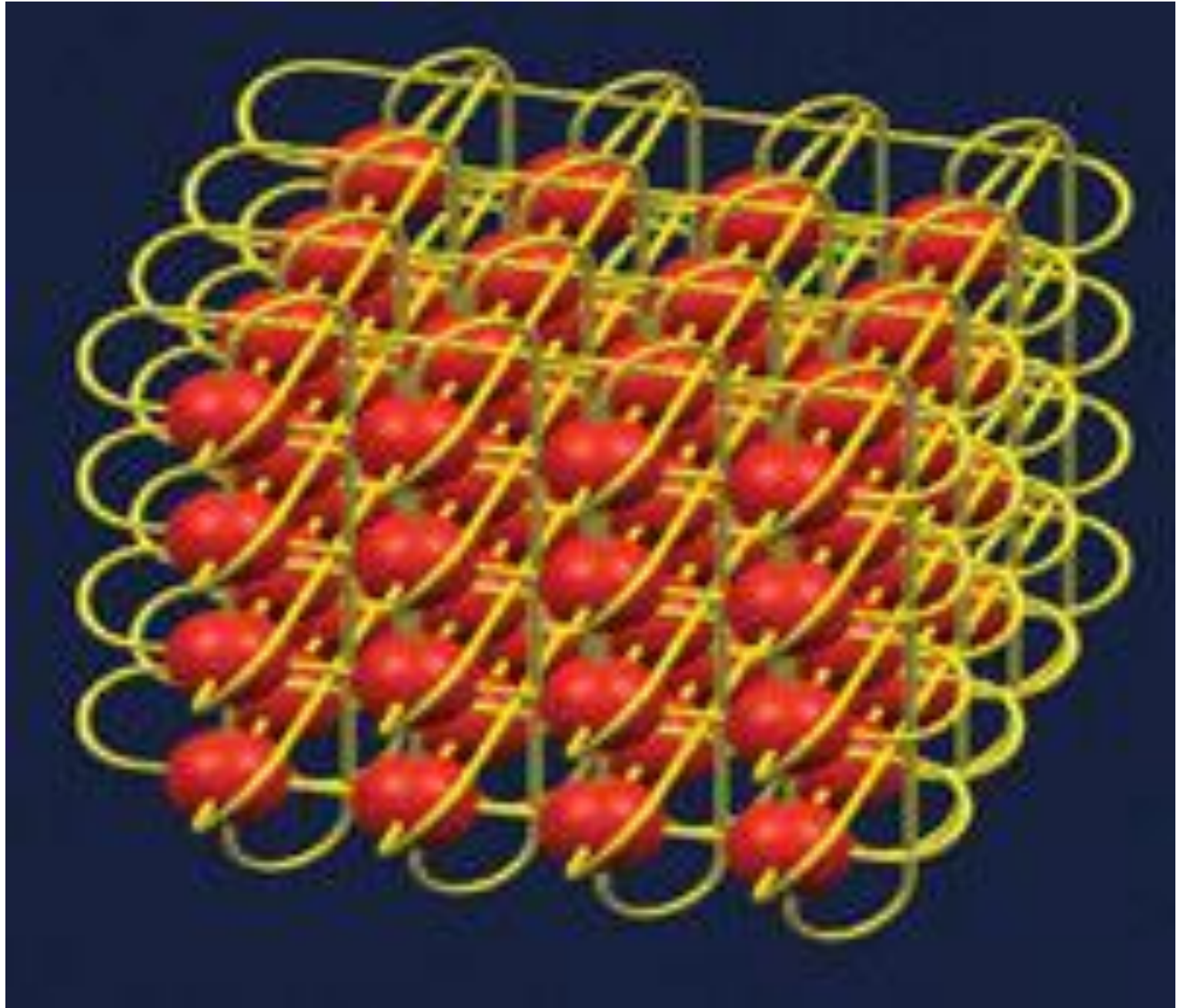
CELL Multiprocessor/65nm



The next generation CELL Multiprocessor

The 65-nm Cell Broadband Engine processor.

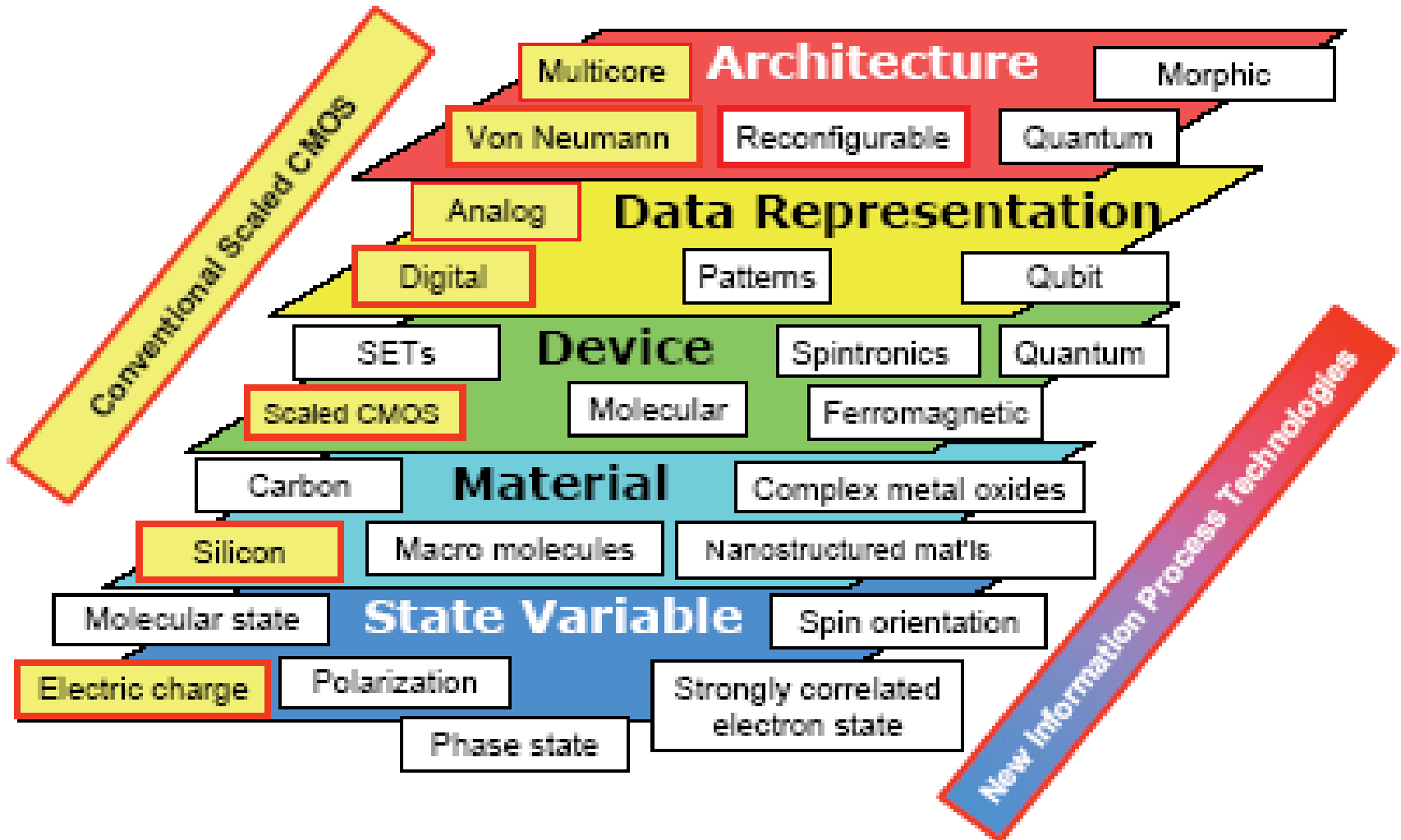
IBM Cyclops 64



A fundamental principle:

- Physics is forcing the cellular, mainly locally connected architectures with *processors and memories* having a *spatial address and bandwidth constraints* plus *spatially distributed dynamic input*

International Technology Roadmap for Semiconductors, ITRS/2007



ITRS/2007 Emerging Research Architectures

Architecture	Implementation	Computational Elements	Network	Application	Research Activity	
Homogeneous Many-Core	Symmetric cores	CMOS	Irregular/Fixed	Synthesis/GPP	158	
Heterogeneous	Asymmetric cores	CMOS	Irregular/Fixed	Synthesis/GPP		
	CMOL	CMOS+molecular switches	Irregular/Fixed	Synthesis/GPP		12
	Molecular cross-bar	Molecular switches	Regular/Flexible	Synthesis/GPP		23
	Check-point	CMOS+ferromagnetic logic	Irregular/Fixed	Synthesis/GPP		3
Morphic	CNN	CMOS+sensors	Regular/Flexible	Recognition/Vision	84	
	AMP	FG-FET, SET	Irregular/Fixed	Recognition/Vision	11	
	Bio-inspired	MFTD, Spin-gain transistor	Mixed	Recognition Mining Synthesis	35	

The architectural challenge

- Combining homogeneous, heterogeneous and morphic
- processor and memory arrays
as architectural components

New principles and results

- ***Many kilo cores and locality* → Virtual and Physical Cellular Machines**
- A new kind of compiler: mapping an algorithm from the Virtual to the Physical Cellular Machine (size, bandwidth, latency, power constraints)
- Equivalent transformations in space and time
- Combining algorithms and physics (noise and delay)
- New kind of decompositions (not the AND OR basis and disjunctive normal form)
- New measures for algorithmic and physical computational complexity – search, algebraic, and dynamics classes

The many-core Cellular Wave Computer Components

The *architectural element* is defined as

- A processor array placed on a 2D or 3D grid
- The processors communicate with a speed inversely proportional with the distance. There are typically two speeds, a local ***f_o*** within a synchron radius ***r***, and a global ***F_o*** via a Manhattan type bus system. In case of continuous time dynamics the ***f*** is $1/2T$, ***T*** being the dominant time constant
- The local and global clock speeds are adjusted to keep the power dissipation within prescribed limit ***P_d***

The architecture is shown, for a 2D case, in Figure 1

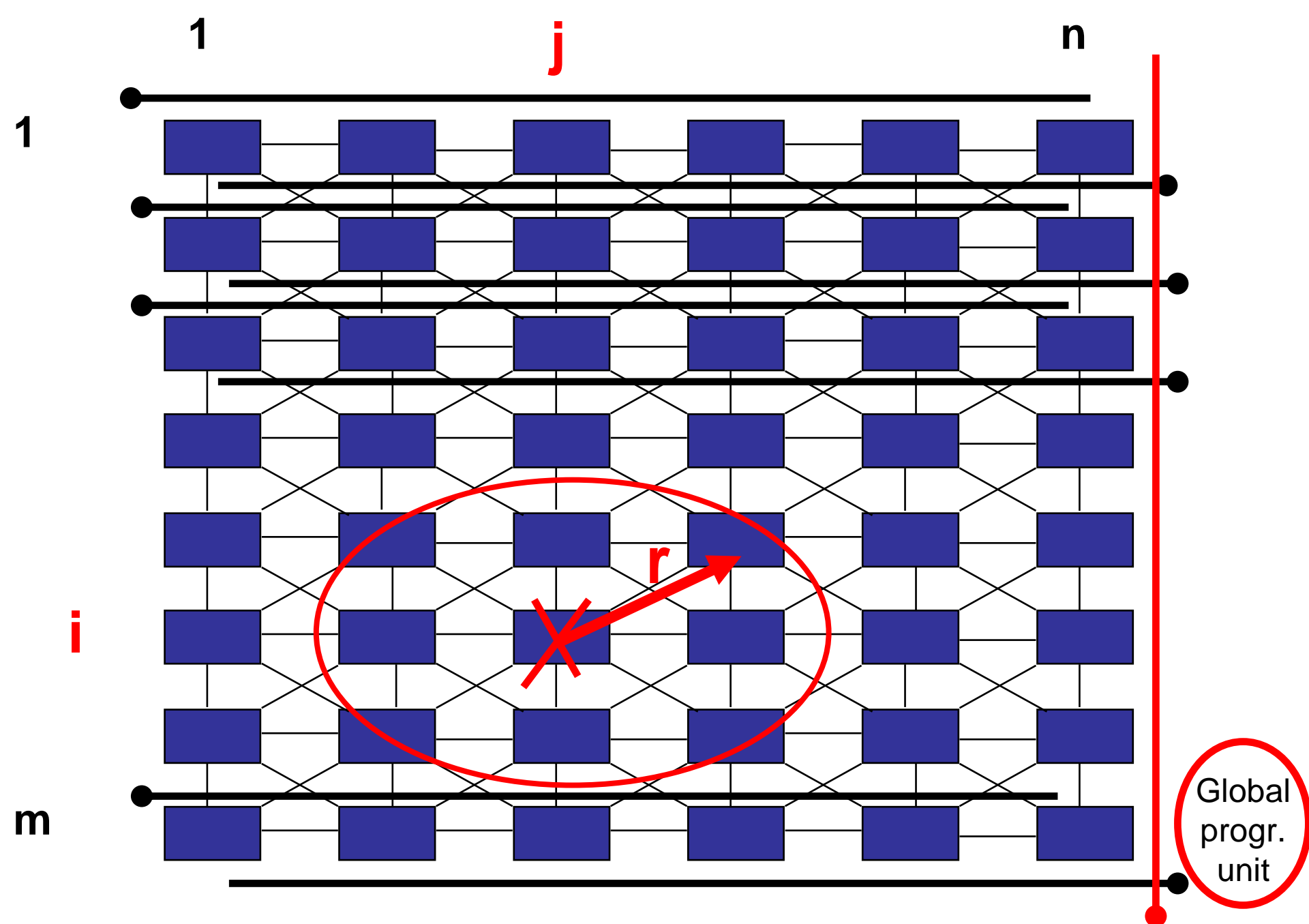


Figure 1: the Cellular many-core 2D Wave Computer architecture

The physical constraints of communication speed and power dissipation

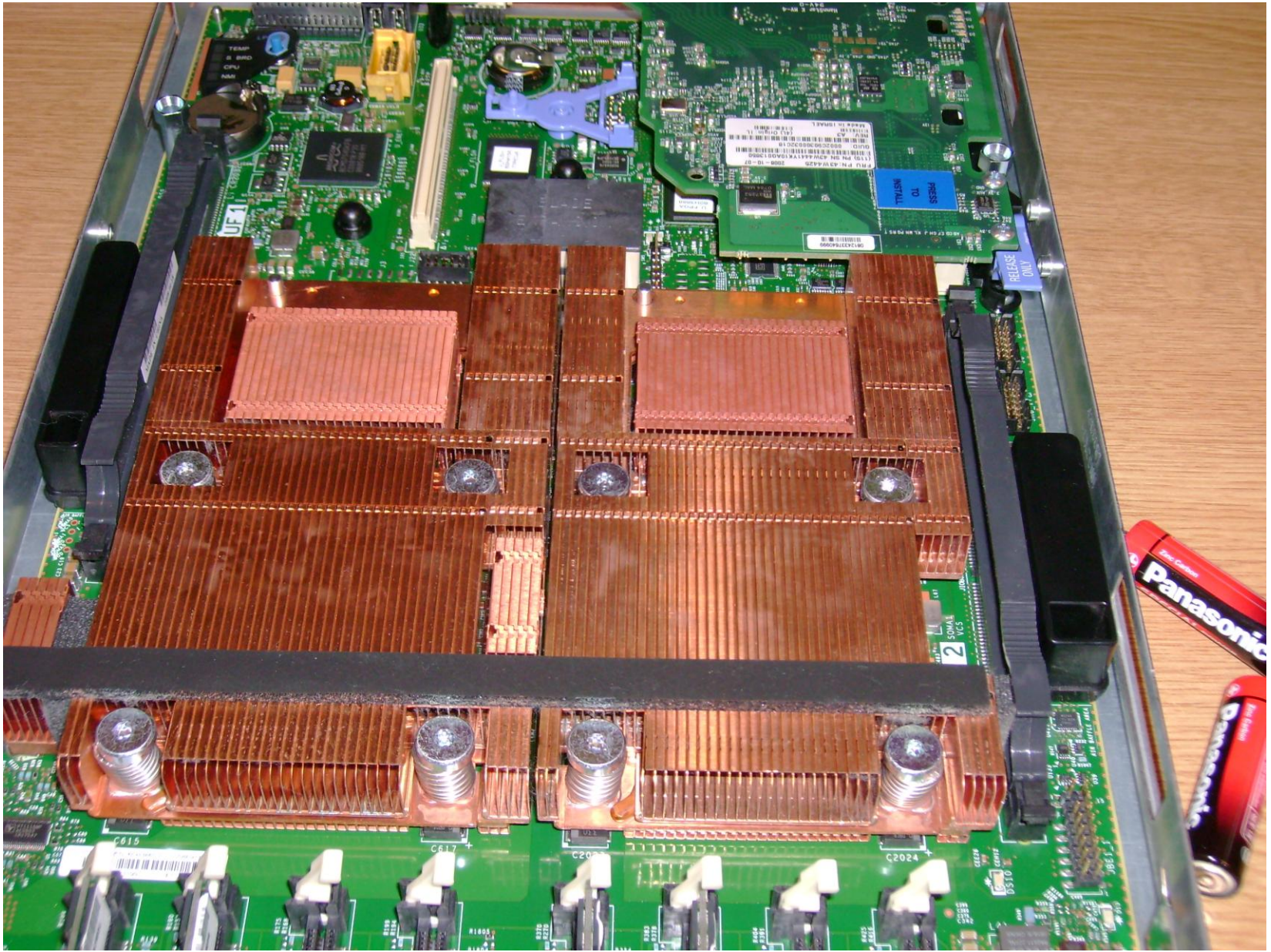
The basic constraints are: wire delay and power dissipation. A side constraint is the number of communication pins (contacts).

The execution of a **logic, arithmetic or symbolic elementary array instruction** is defined via r input ($u(t)$), m output ($y(t)$) and n state ($x(t)$) variables (t is the time instant).

This unit is characterized by its

- **s**, surface/area,
- **e**, energy,
- **f**, operating frequency,
- **w = e f** local power dissipation, and
- the signals are traveling on a wire with length **L** , width **q**, and with speed **v_q** introducing a delay of **D = L v_q**

- **Ω cores** can be placed on a **single Chip** , typically in a rectangle grid, with R input and Q output physical connectors typically at the corners of the Chip, altogether there are **K** input/output connectors.
- The maximal value of dissipation of the Chip is **W** .
- The physics is represented by the maximal values of **Ω , K , and W** (as well as the operating frequency).
The operating frequency might be global for the whole Chip **F_0** , or could be local within the Chip, **f_0** , **f_i** (some parts might be switched off, **$f_i = 0$**)



Virtual Cellular Machines

- Computational building blocks are mainly arrays, composed of operator and memory elements, acting in space and time
- Heterogeneous, homogeneous and morphic computational blocks and memory blocks
- Building blocks have no physical constraints
- size, bandwidth, latency, speed, power dissipation

- A typical **logic elementary array instruction** (LA) is a binary logic function on n variables, or a memory look-up table
- A typical **arithmetic/analog elementary array** (AA) instruction is a multiply and accumulate (add) term (MAC core) array,
- A **symbolic elementary array instruction** (SA) might be a string manipulation core array (e.g. a P system)
- a **complex cell based array instruction** (XA), hosting cells with all the three above types of data and instructions

An 8, 16, or 32 bit microprocessor could be considered as well as an elementary array instruction with iterative or multi-thread implementation.

A Virtual Cellular Machine

is composed of five types of building blocks:

- (i) *cellular processor arrays/layers*, **CP**, with simple (L, or A, or S type) or complex (X) *cells and their local memories*, these are the protagonist building blocks,
- (ii) *classical* digital stored program *computers*, **P** (microprocessors),
- (iii) multimodal topographic (**T**) or non-topographic *inputs and outputs*, **I/O** (e.g. scalar, vector, or matrix signals),
- (iv) *memories* of different data types **M**, organized in qualitatively different sizes and access times (e.g. in clock cycles), and
- (v) *interconnection pathways* **B** (busses).

- **tasks**, the algorithms to be implemented, are defined on the Data/Memory Architecture of the Virtual Cellular Machines

Two main types of machines:

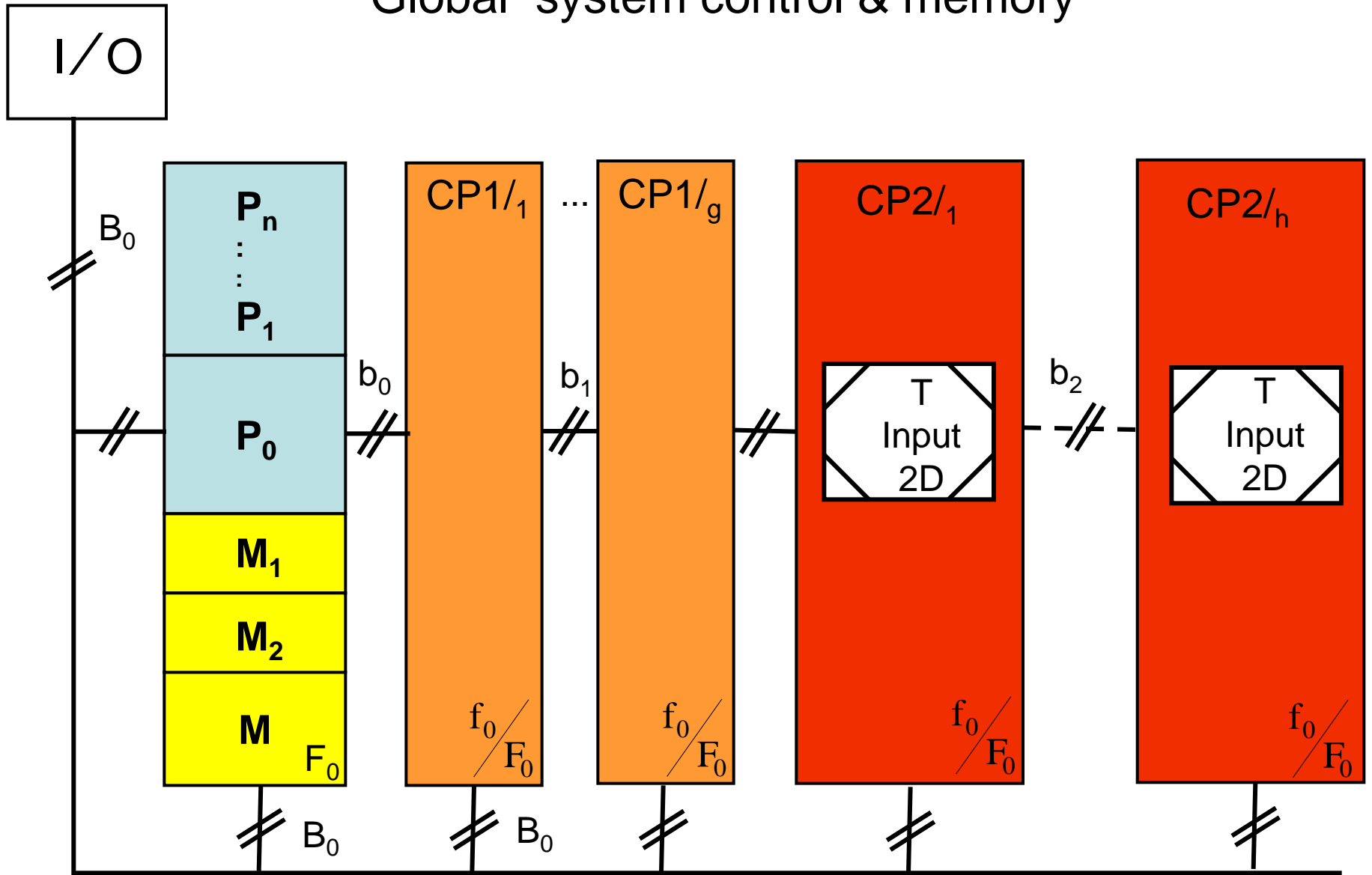
- Single- or multi- layer machine with identical cells, called **homogeneous** cellular machine, and
- Single- or multi-layer machine with different cells, called **heterogeneous** cellular machine.

Two heterogeneous cellular machines

Notations for the five building blocks:

- CP: - one dimensional , CP1, and two dimensional , CP2, cellular processor arrays, their sizes are large enough to handle the given task,
- P: - classical digital computer with memory & I/O, for example a classical microprocessor
- T: - topographic fully parallel 2D (or 1D) input
- M: - memory with high speed I/O, (L1, L2, L3 parts as cache and/or local memories with different access time ranges)
- B: - data bus with different bandwidth ranges (B1, B2, ...) and latencies (D1, D2, ...)

Global system control & memory



The physical implementation of the kilo-core components and mega-core system architectures

We have three **elementary programmable CELL processor** (cell core) types in **array** implementations

- A) **An algorithm** with input, state and output vectors having real/arithmetic, binary/digital logic, and symbolic variables (*typically implemented via digital circuits*).
- B) **A real valued** state and output **dynamic system** with analog/continuous or arithmetic variables (*typically implemented via mixed mode/analog-and-logic* circuits and digital control processors)
- C) **A physical dynamic entity** with well defined **geometric layout** and I/O ports (function in layout) – (typical implementations are CMOS and/or beyond CMOS *nanoscale designs*, or optical architectures with programmable control)

Physical Cellular Machines

Via the same building blocks, however, with given physical parameters as the

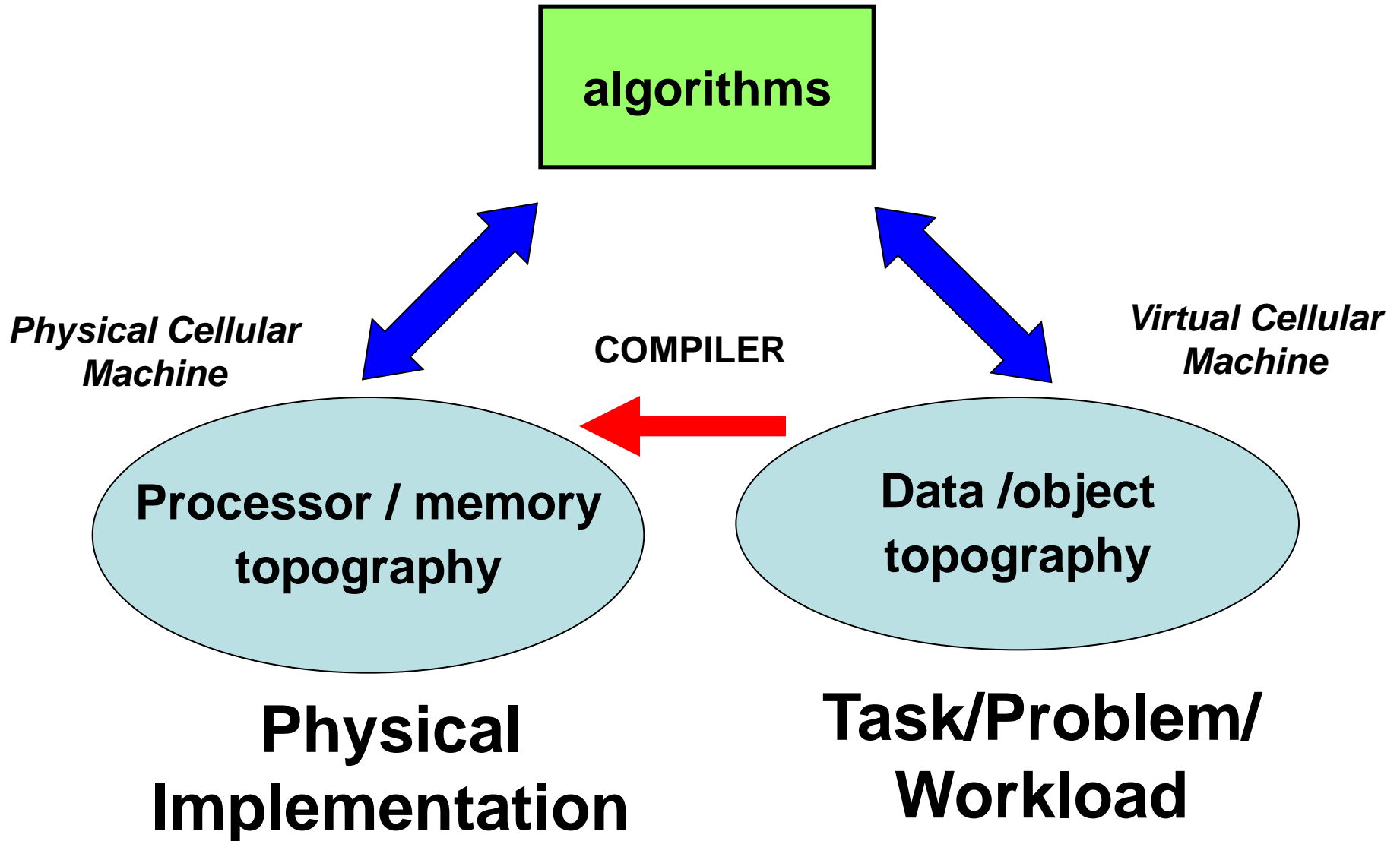
- Size
- Bandwidth/latency
- Operation speed
- Power dissipation

It might have the same or a different architecture
- compared to the Virtual Cellular Machine

- The **geometry of the architectures** are reflecting the physical layout of the chips or systems. A building block could be implemented as a separate chip or a part of a chip. This architectural geometry defines also the communication (interacting) speed ranges. Hence physical closeness means higher speed ranges.
- The **spatial location** or **topographic address** of each elementary cell processor or cell core, as well as that of each building block within the architecture, and the ***communication bandwidth***, plays a crucial role.

In addition to the number of processors, these are the most dramatic differences compared to classical computer science.

The Design Scenario



The representation of a topographic algorithm

If the algorithm is operating on a topographic problem, that is, if the one-to-one correspondence between a processing core cell and a physical or data object is straightforward (e.g. a pixel, taxel, voxel, etc.), the problem is

to find the ***equivalent transformations***

- using additional space (array) to save running time (***time to space***), and
- to manage the bandwidth/speed constraints, that is using parallel mainly identical operators (arrays) to keep the channel speed lower (***channel speed/ bandwidth to space***) and
- ***power-dissipation***/energy constraints.

For homogeneous architectures : four solutions by Á. Zarándy

Array Signals, variables, memory

■ logic / symbolic array

□ logic / symbolic value

● arithmetic/analog array

○ arithmetic/analog value

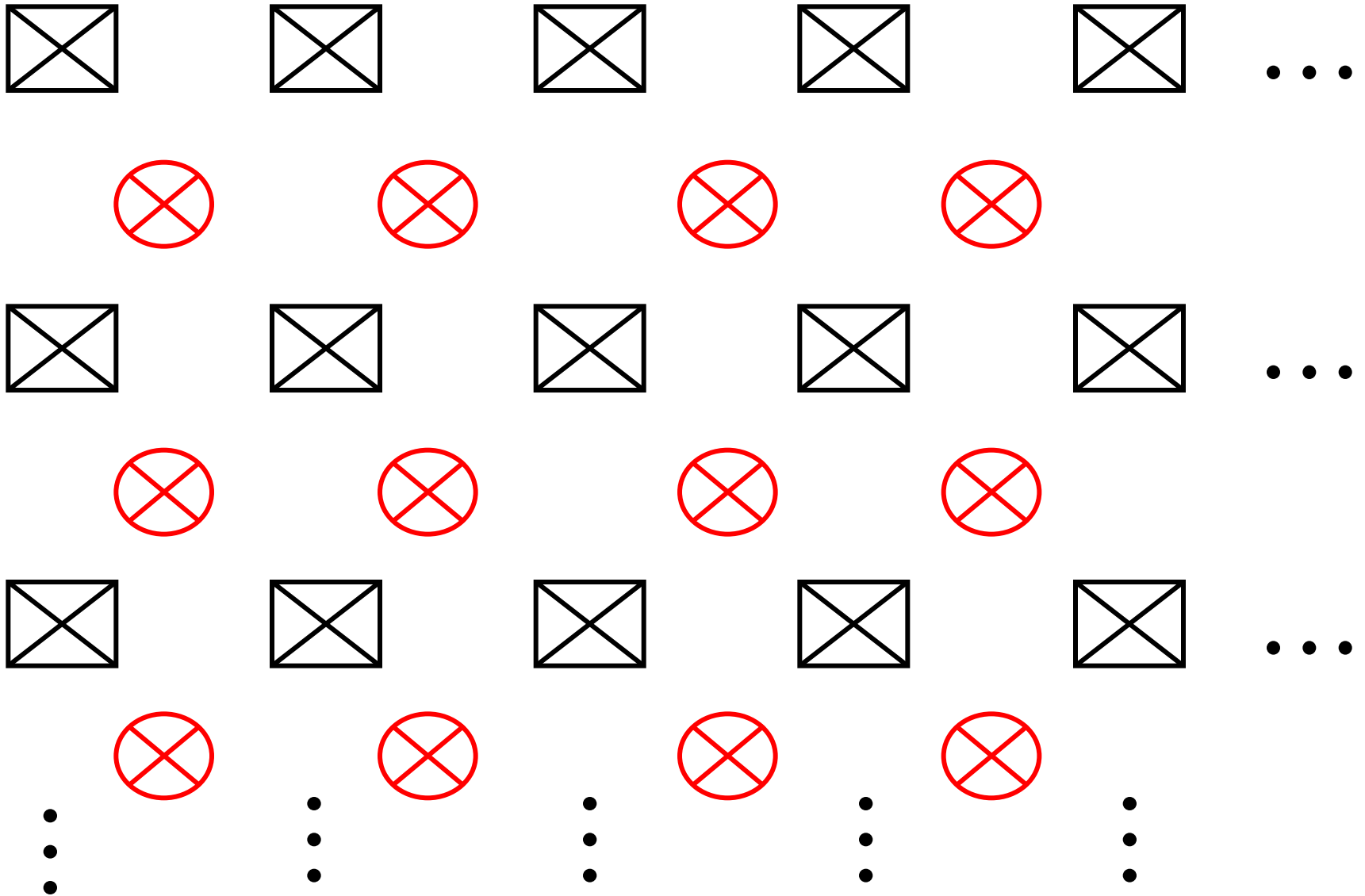
Processors

⊠ logic/ symbolic processor

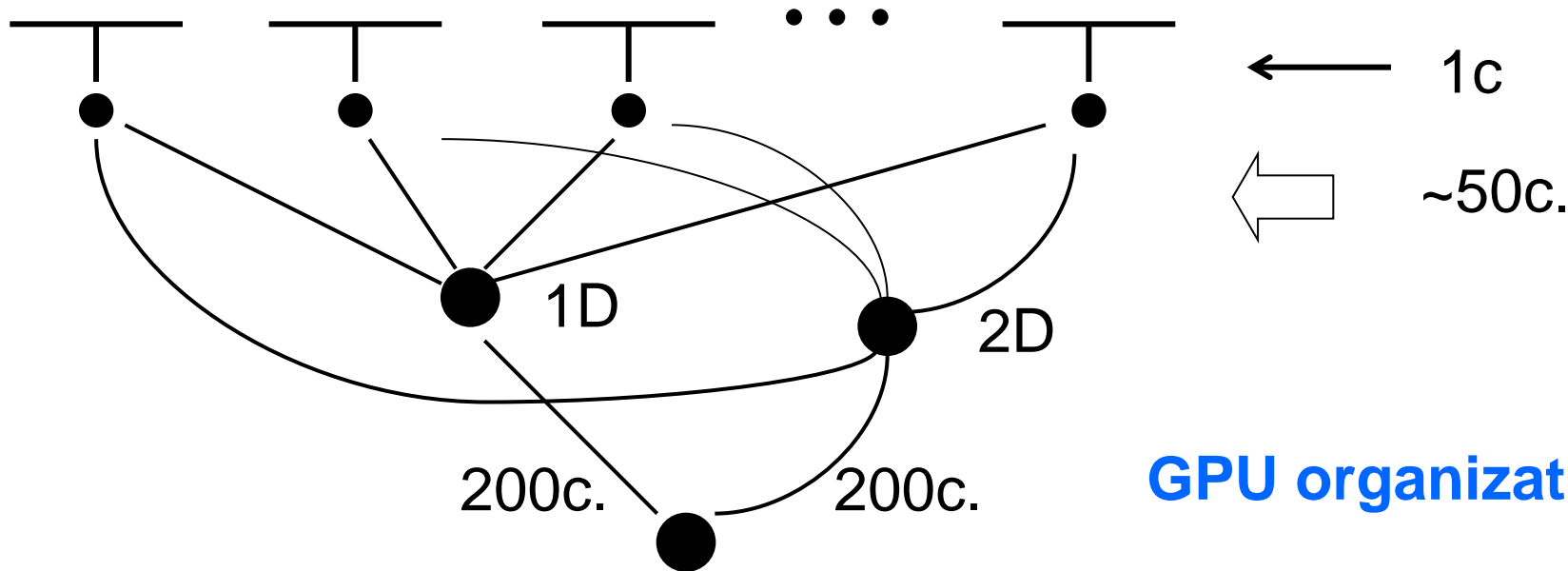
⊗ arithmetic/analog processor

— arithmetic/analog processor array

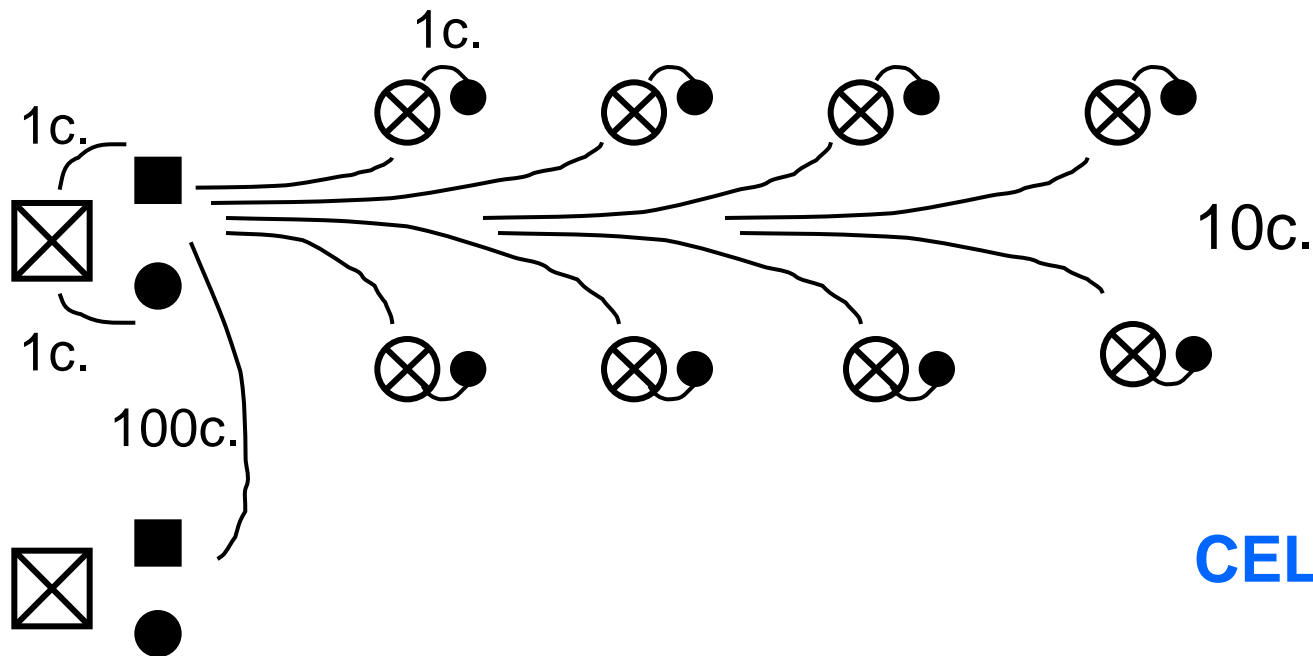
- - - - logic/symbolic processor array



Tiled-in cellular processor arrays

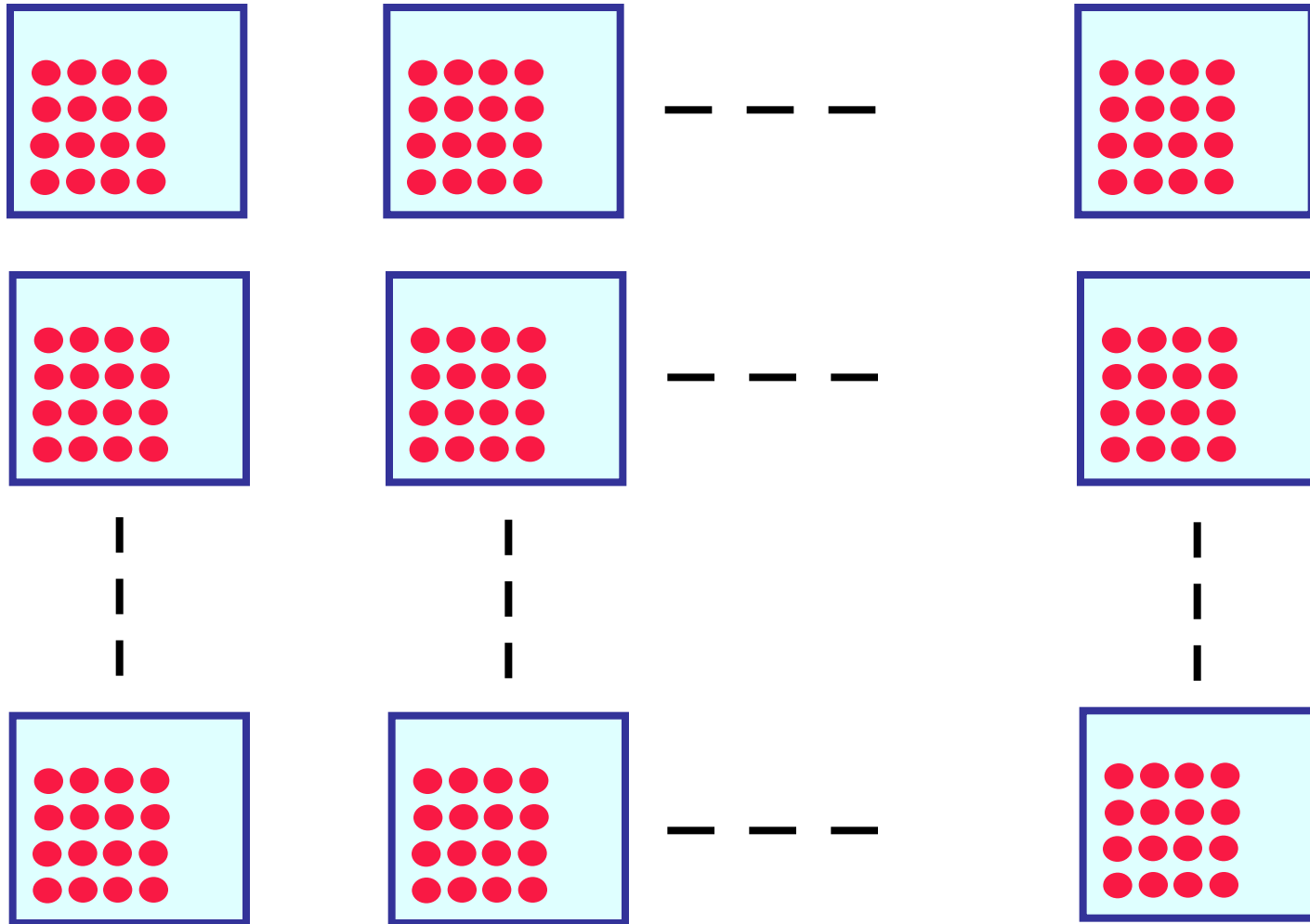


GPU organization



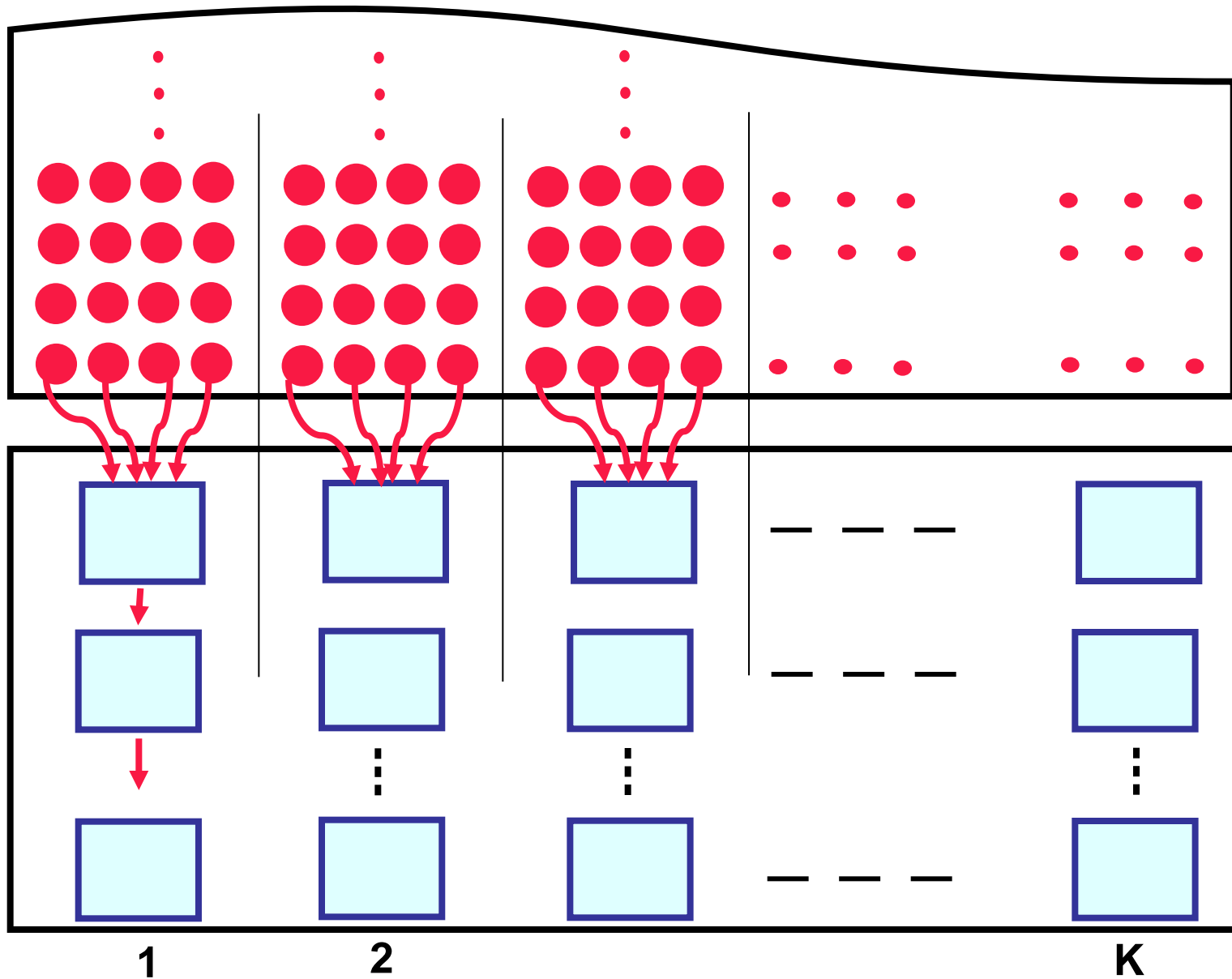
CELL organization

Many sensors/data/memory units on one Cellular Processor

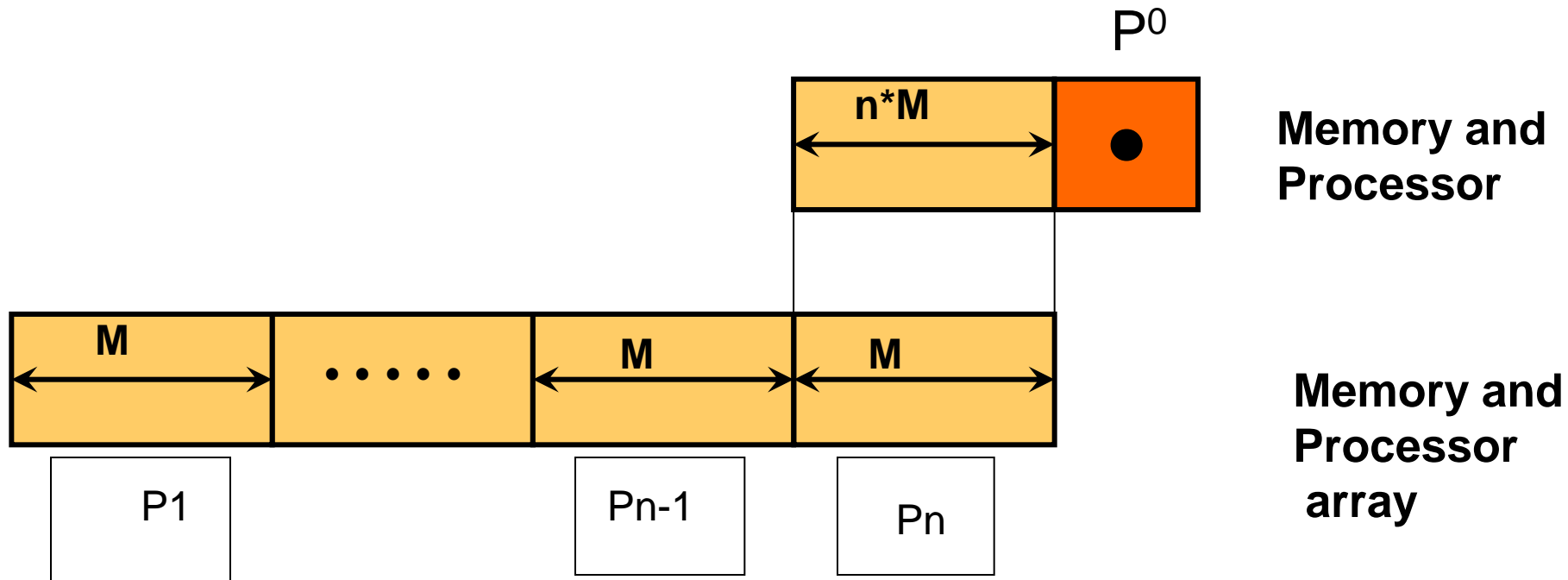


Separate sensory/memory plane

Flow architecture

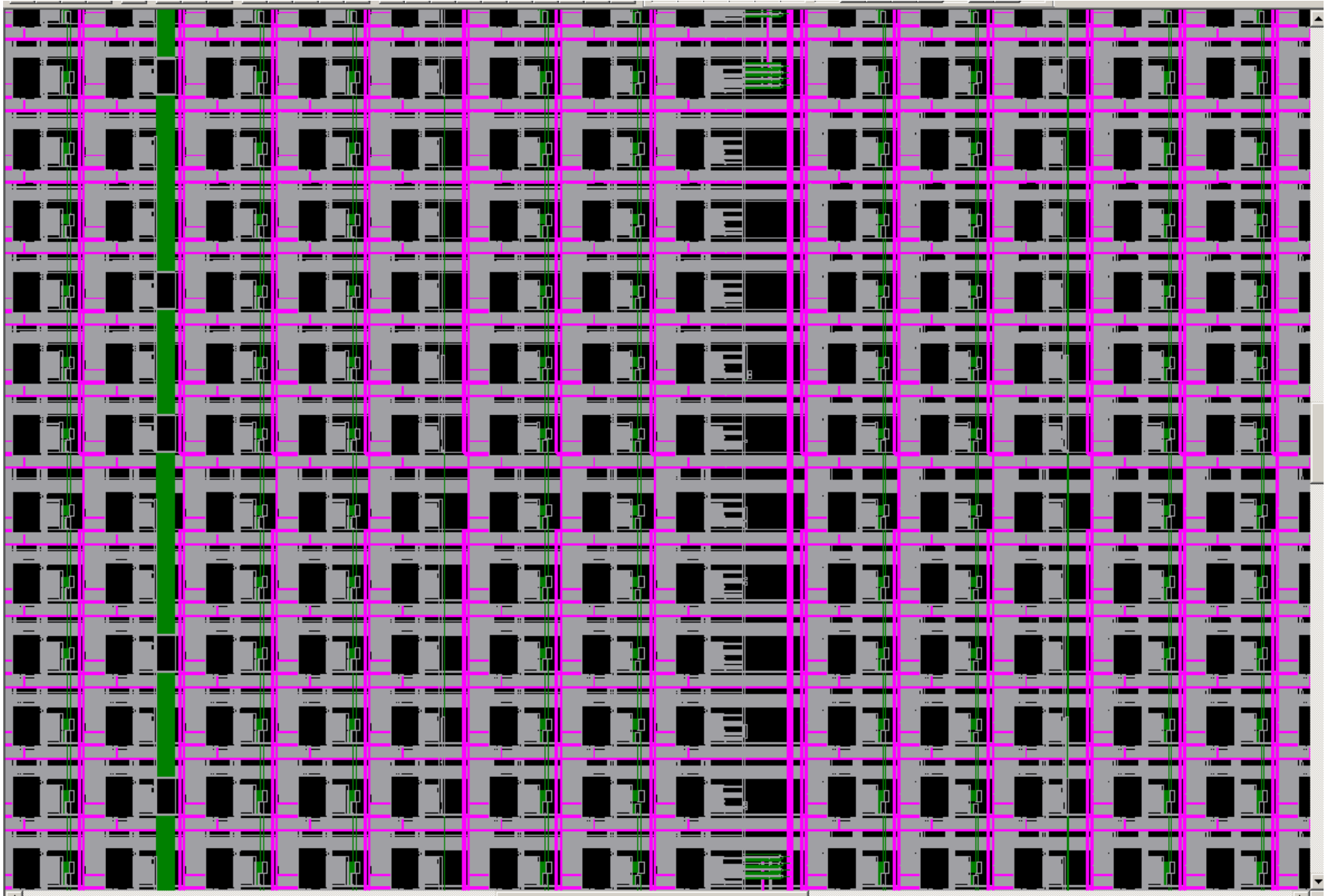


Time-to-space mapping algorithm

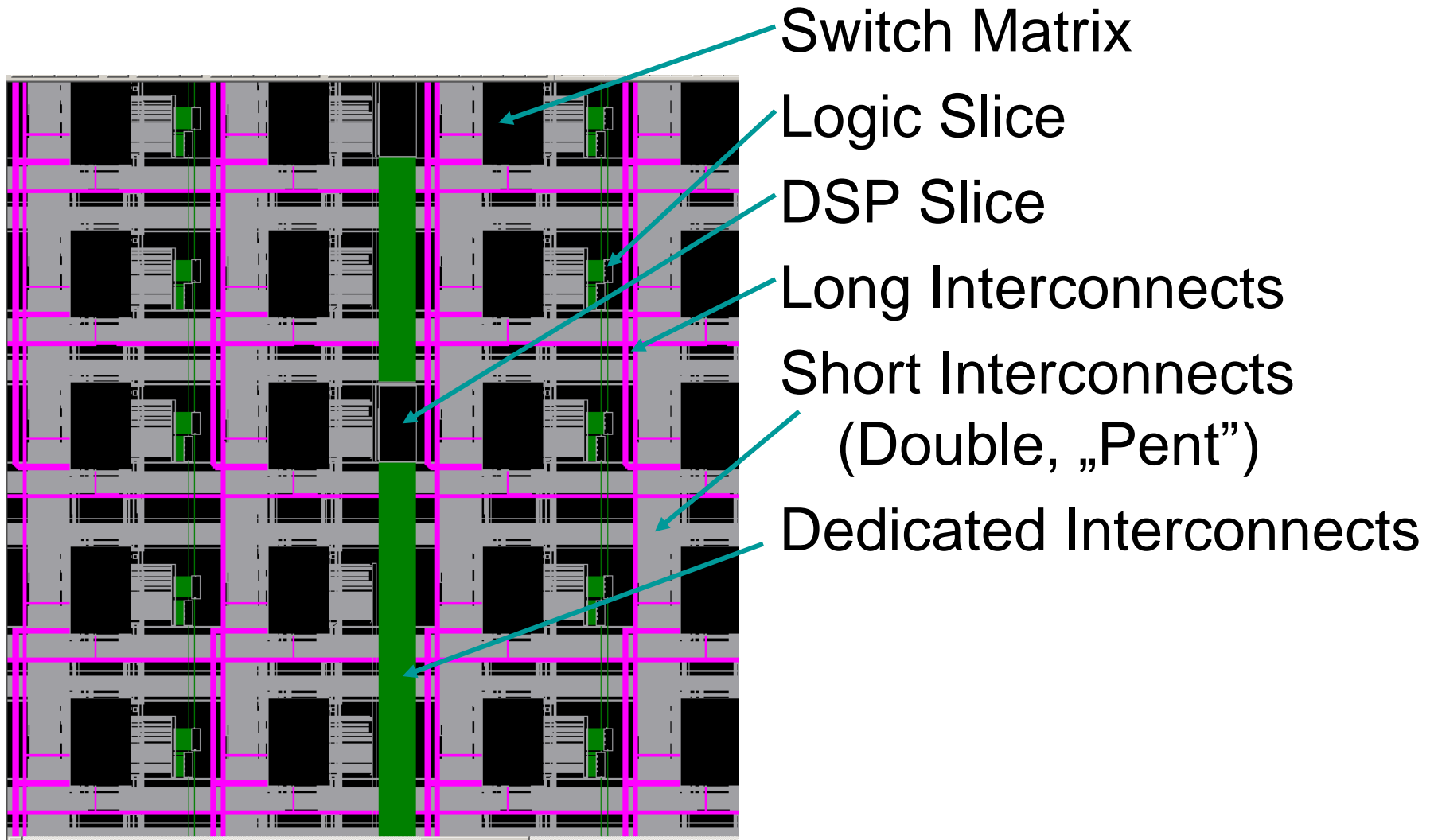


Pipeline or Parallel ... **More:** communication delay and power dissipation

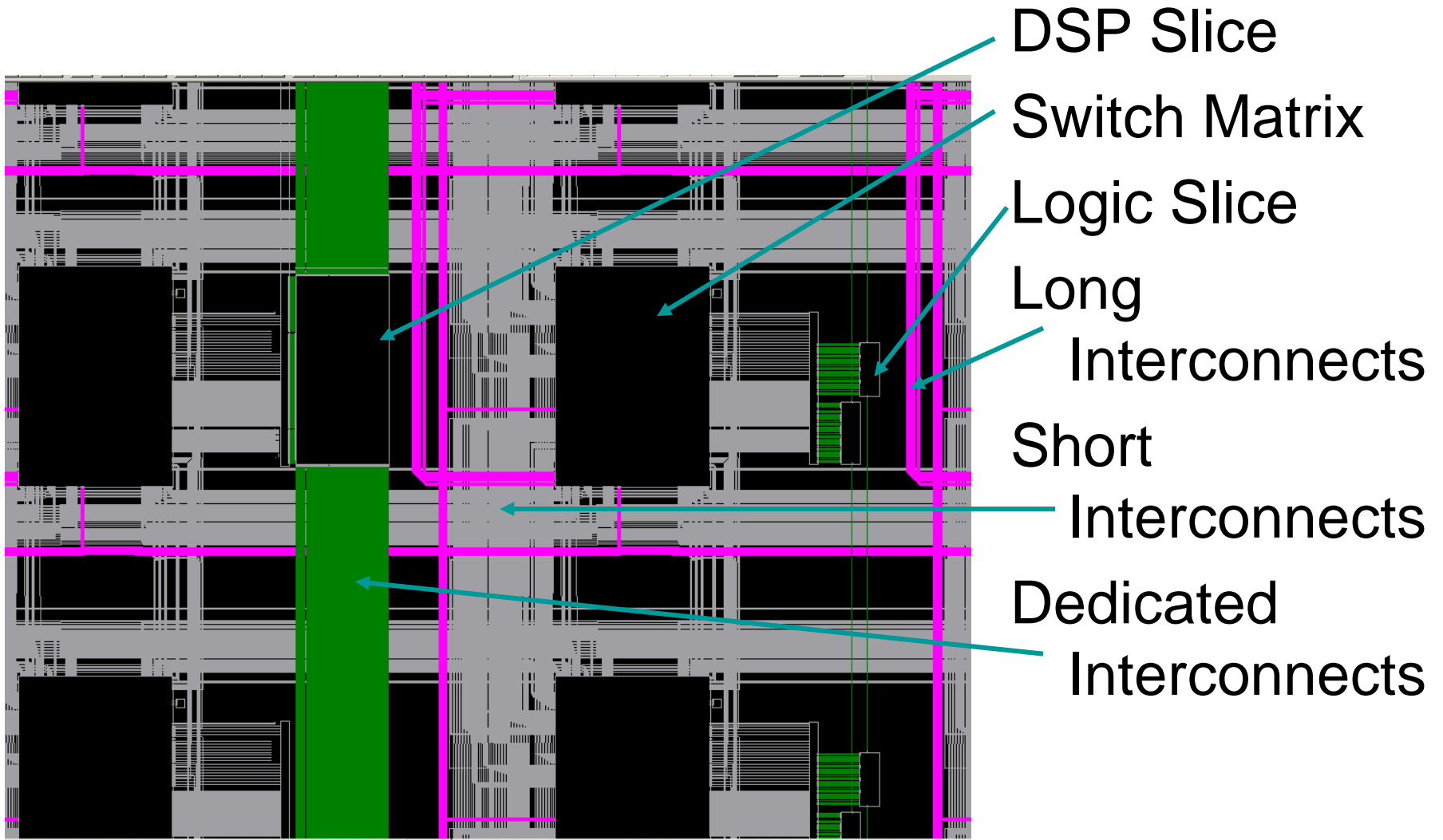
Virtex-5 FPGA



Virtex-5 FPGA



Virtex-5 FPGA



A compiler and equivalent transformations for FPGA implementation (Z.Nagy and P.Szolgay)

- Compiler: the FALCON architecture via the CNN Universal Machine.
- Special cases for
 - Navier Stokes 2D, 2½D, incompressible and compressible
 - Multilayer retina model
- Introducing special FIFOs to overcome the bandwidth constraint and the global control

New principles of Computational Complexity

- The **algorithmic and physical complexity measures** of these many core architectures will be eventually different compared to the single processor systems.
- With Ω cores and M total memory the *virtual algorithmic complexity* $C_a = C_a(\Omega, M)$
- The *physical computational complexity* of an algorithm is **measured by a proper mix of speed, power, and area.**

Two generic examples

- Generating true random binary patterns on an ACA 16k cellular visual microprocessor combining on chip algorithms and on chip physical noise (M.Ercsey-Ravasz et. Al.)
- Hyperacuity in time: time difference computation converting time to space – a custom chip (A. Mozsáry et.al.)

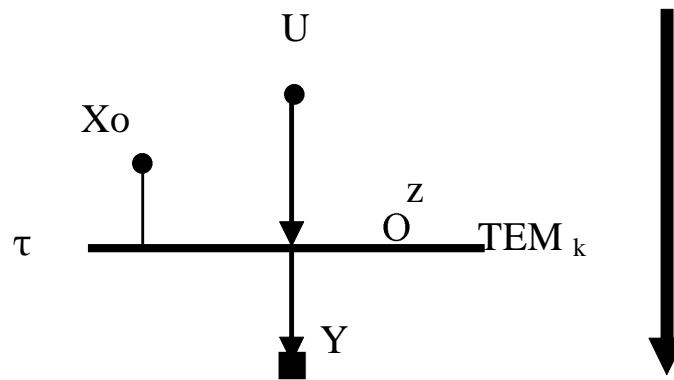
Dynamic operational graph and its use for acyclic UMF diagrams

- Nodes are memories
- Branches are either
 - the operators or
 - communication paths

Directed acyclic graphs representing UMF diagrams

- Genetic Programming with Indexed Memory (GP-IM)
- Extremal graph problems

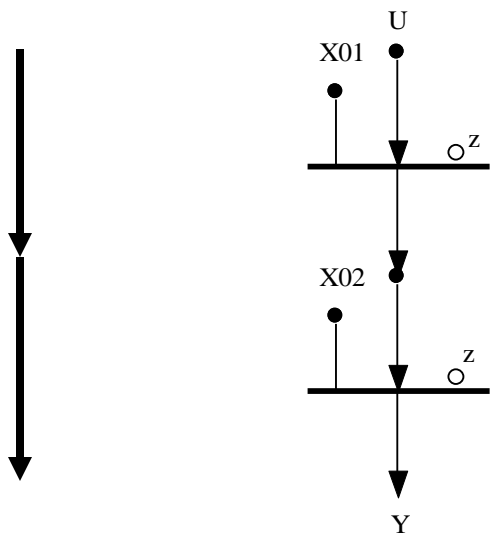
A single cellular array/ layer:



U: input array, X_o Initial state array, z : threshold or mask array, Y : output array,
 τ : time constant or clock time, TEM_k : local instruction

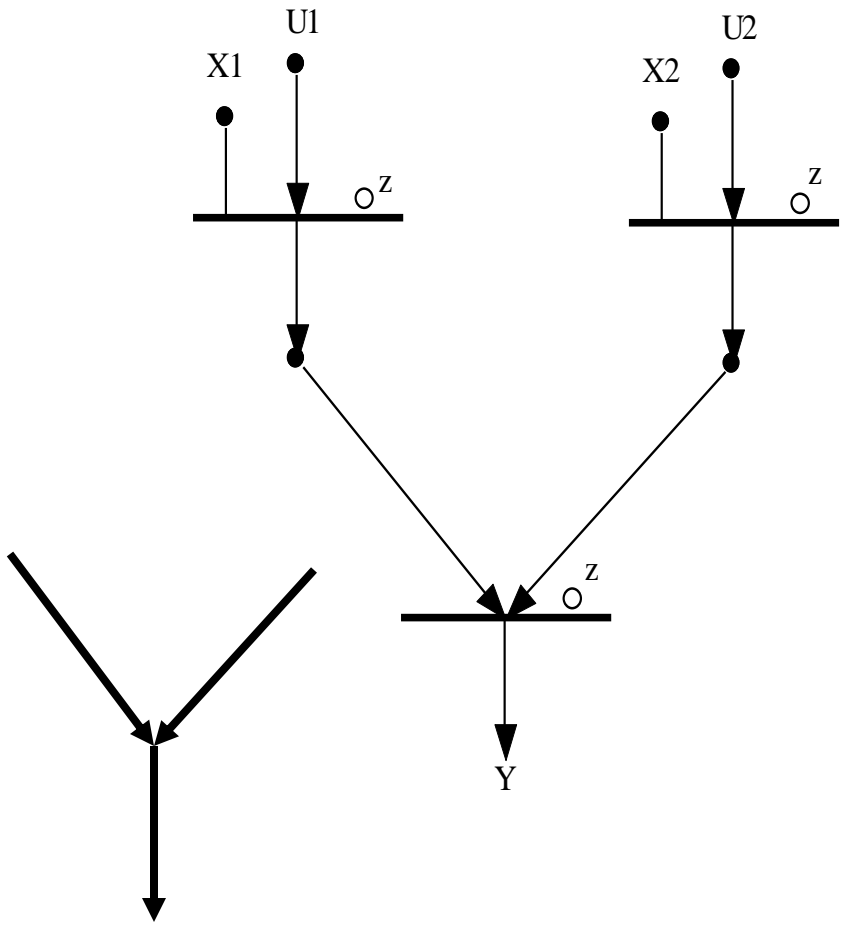
- *Algorithmic structures*
- *in terms of arrays/layers*

Cascade



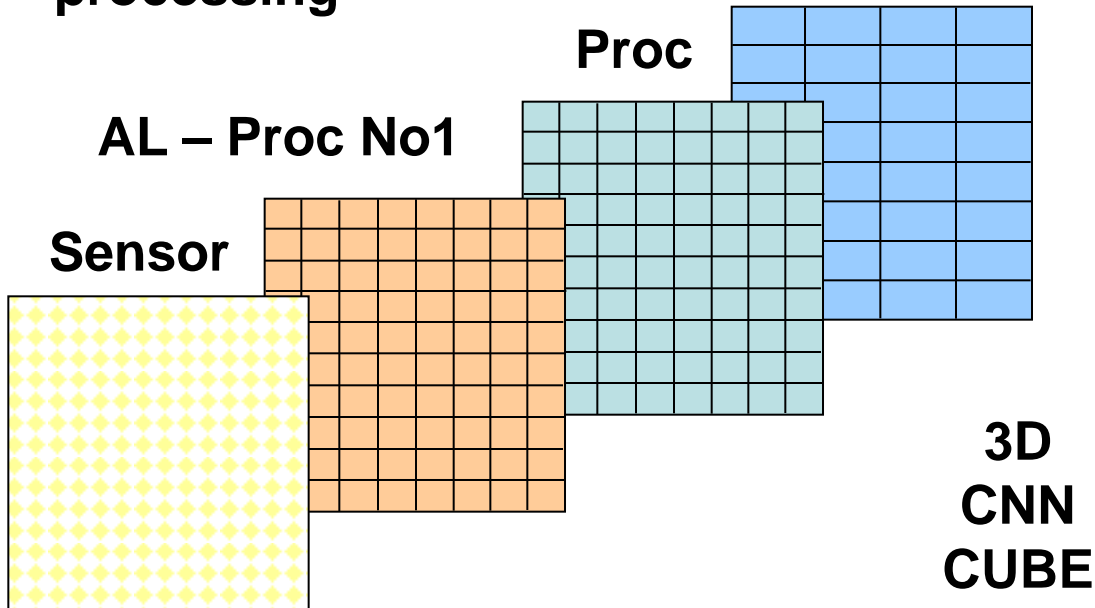
Parallel

A typical parallel structure with two parallel flows is shown below, by combining them in the final layer

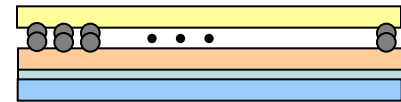


A NEW TECHNICAL APPROACH: 3D INTEGRATION

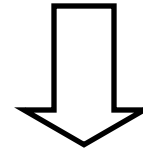
Topographic,
layered
sensing-
processing



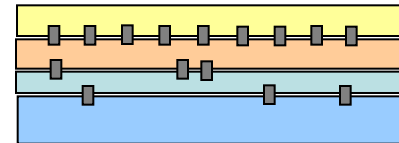
AFRL BB Process



Bump bonding:



3D stacking:

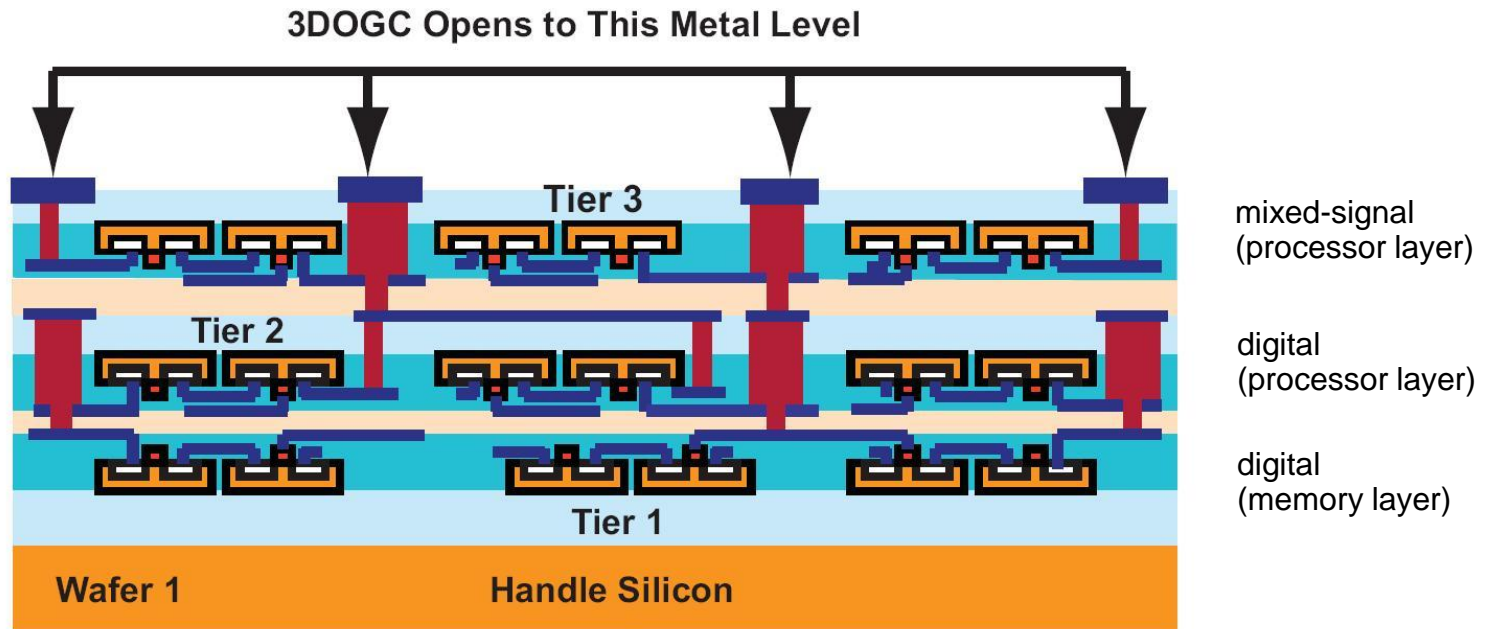


MITLL Low-Power
FDSOI CMOS Process

TECHNICAL APPROACH: 3D TECHNOLOGY

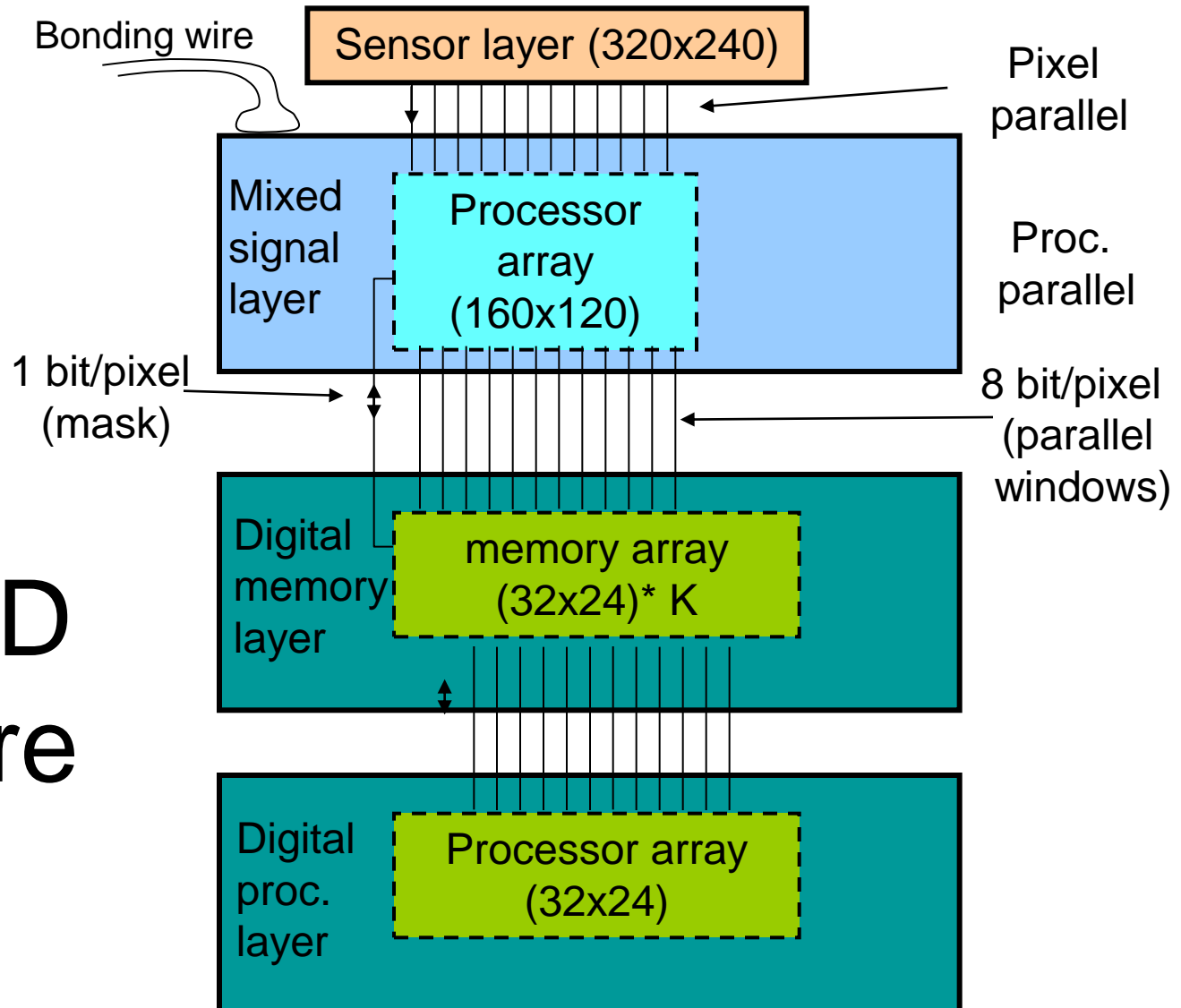
MIT Lincoln Laboratory Low-Power FDSOI CMOS Process

Three tiers will be integrated to form a 3D integrated circuit

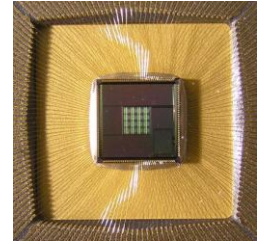


MITLL 3DM2 0.18 micron 3 SIO layers (+ BB Si/InGaAs sensor – not shown)

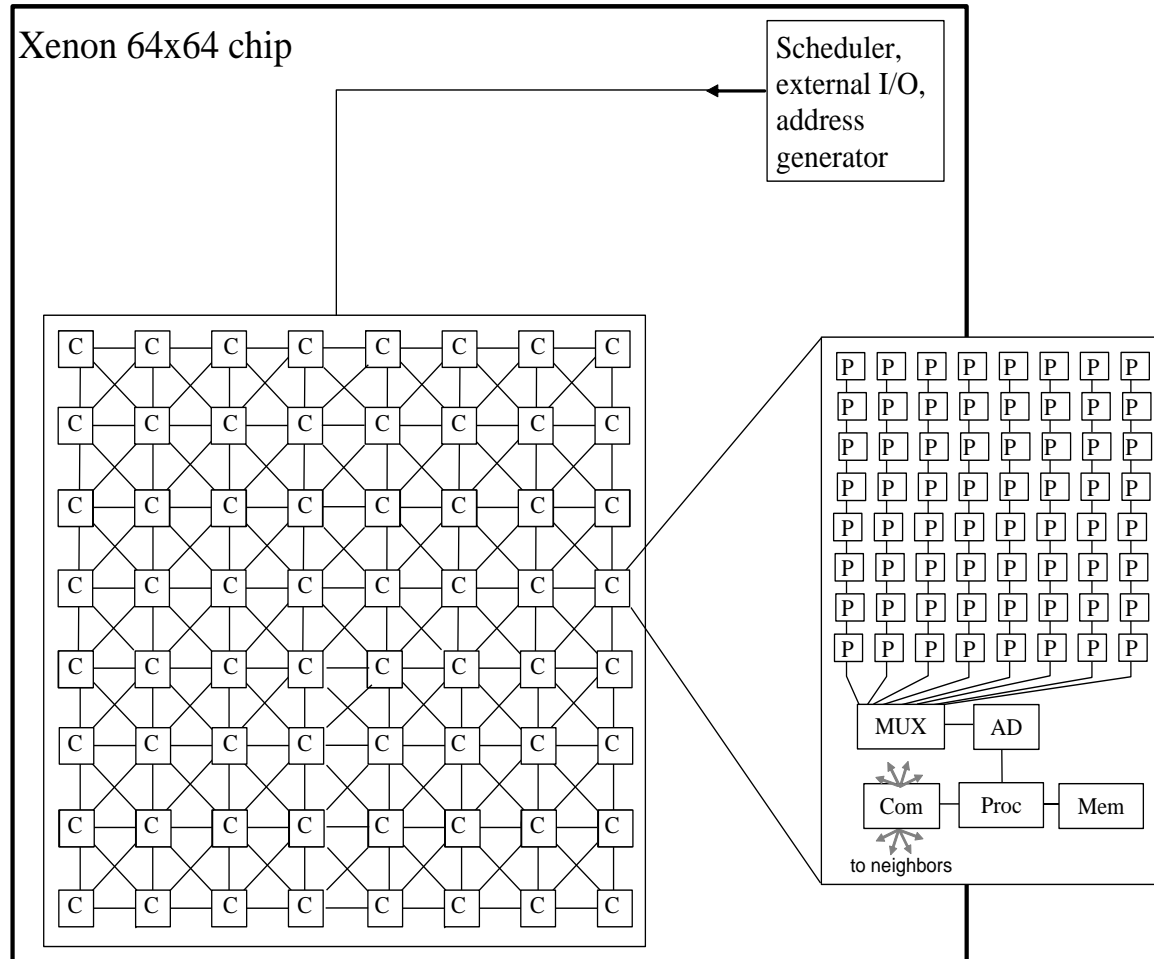
Viscube 3D architecture



Xenon Architecture



- 64x64 sensor-processor array
- Neighboring cells are directly interconnected
- Each cell is prepared to process 8x8 pixel array (scalable)
- SIMD
- 10GOPS, 20mW
- 500GOPS/W
- On-chip sensors



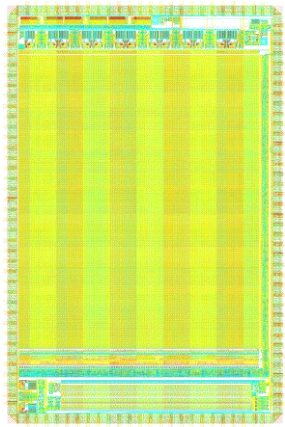
Q-Eye chip

AnaFocus Ltd, Seville

- Ancestor: ACE16k cellular visual microproc.
- 176x144 processor array
- 1 (4) sensors/cell (R,G, B +Gray)
- 8 LAM, 7 LLM
- ~30x30 micron pitch
- 1% output accuracy
- 20mW (~300mW) power consumption at 30 (1000) frame per sec

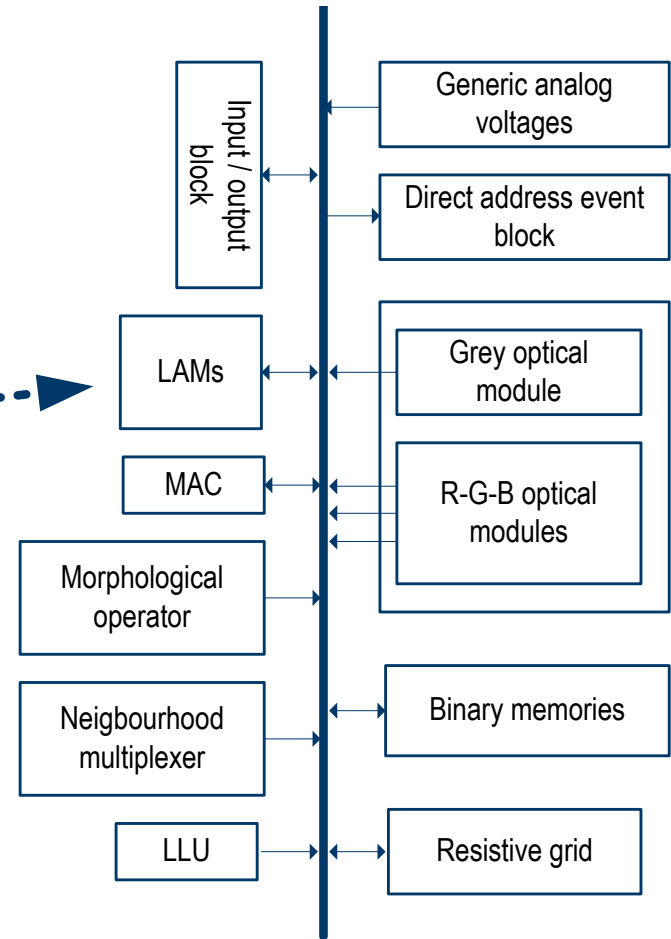
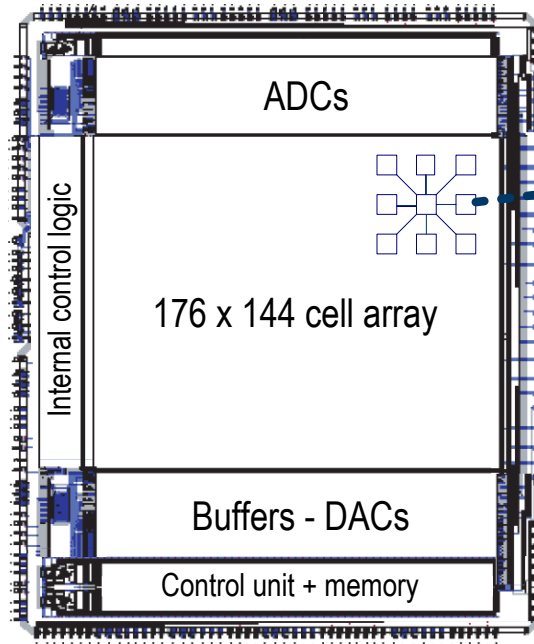
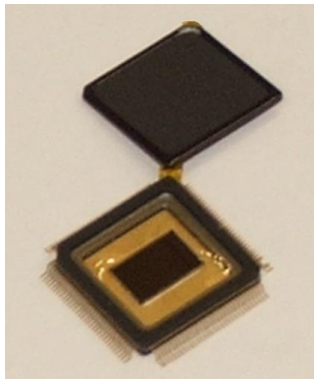
Q-EYE chip architecture

Q-Eye: mixed-signal cellular visual 25k microprocessor fabricated in 180nm CMOS



7mm

5mm



References

Journal Special Issues:

Int. J. Circuit Theory and Applications,
Jan./Feb.and July/Aug., 2006, July/Sept2008/May2009
IEEE Circuits and Systems Magazine, No. 2
2005 International Journal on Bifurcation and Chaos, February,
2004 (a bio issue, 3 papers related to CNN)
IEEE Trans, Circuits and Systems I, May, 2004

Book: Cellular Nanoscale Sensory Wave-Computers –
CNN Technology in Nanoscale, Springer, Sept. 2009

Conf. Series: IEEE CNNA Intl. Workshops, since 1990,
coming soon: CNNA 2010

web sites: http://cnaa2010.itk.ppke.hu/CNNA_2010/
<http://www.anafocus.com>
<http://www.eutecus.com>