Integrated Systems
Challenges and Opportunities

Giovanni De Micheli
Centre Systèmes Intégrés

Integrated systems

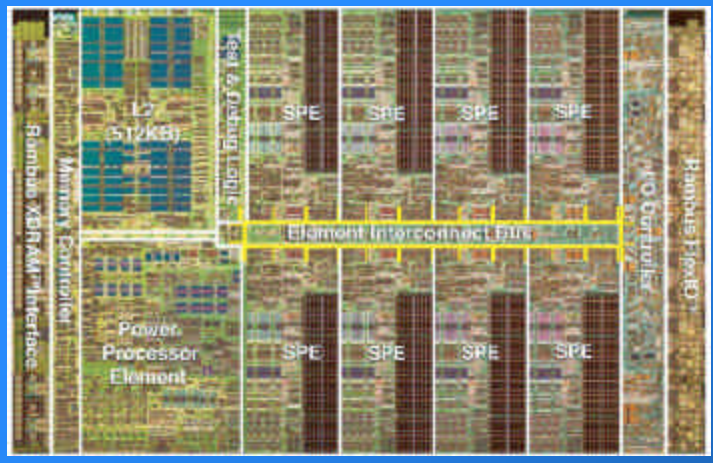
- Ubiquitous presence of integrated circuits and systems in products
- The *market pull:*
  - Run demanding SW applications with minimal energy consumption
- The *technology push*:
  - Pushing the physical limits of computational structures

Anecdotes

- I think there is a world market for maybe 5 computers - T. Watson – IBM, 1949
- There is no reason anyone would want a computer in their home - K. Olsen – DEC 1977
- I see no advantage whatsoever to a graphical user interface – B. Gates - Microsoft, 1983
- The cost of silicon in a car is higher than the cost of steel - circa 2000
- Communications of ACM dedicates a full issue to internet games - November 2006

Multi-processor
Systems on Chips

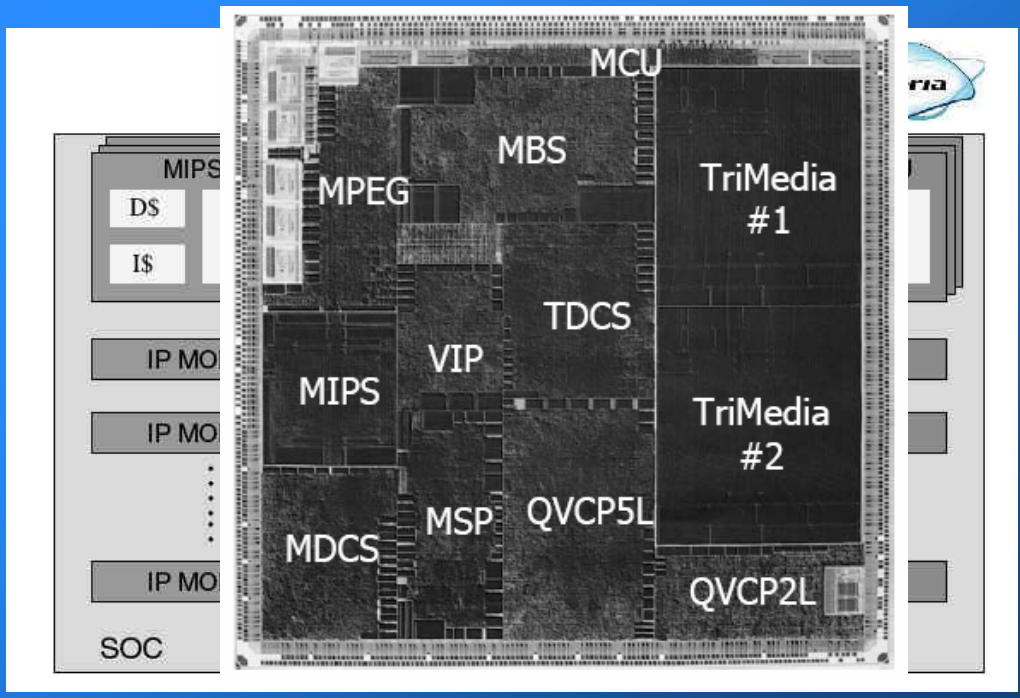
- Large-scale systems
  - Billion-transistor chips
  - Multi-cores, multi-threaded SW
  - Power-consumption limited
- Very expensive to design
  - Non recurring engineering costs
  - Require large market



IBM Cell Multi-Processor

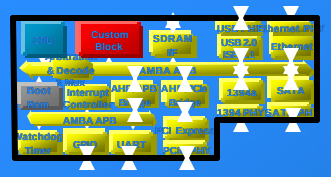
Platforms

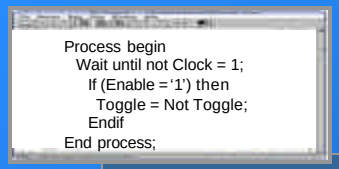
- Address *application-specific* needs
  - Domain-specific hardware
  - Differentiation via software
- Examples
  - Telecom:
    - Philips Nexperia
    - ST Nomadic
  - Automotive

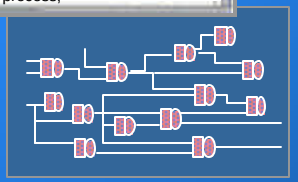


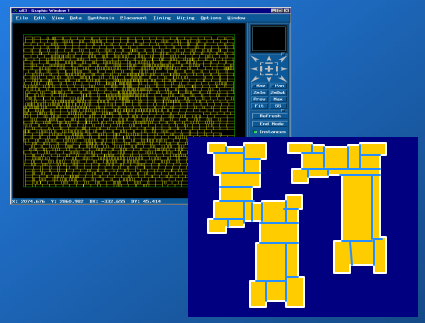
Designing a large chip

10^3	Blocks	System
10^5	RTL lines	RTL
10^8	Gates	Netlist
10^9	Transistors	Circuit
10^{12}	Polygons	Layout
10^{13}	Trapezoids	Mask







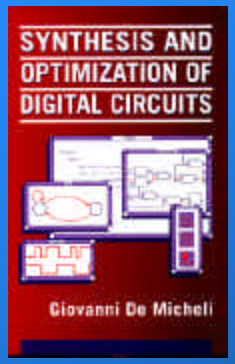


CMOS, mostly digital, 65nm, >200mm^2

Challenges & solutions

- Complexity (giga scale)
  - Intractable large scale problems
- Technology (nano scale)
  - Ever shrinking CMOS
  - New disruptive technologies
- Architectures
  - Multi-processing
  - Structured communication
- Objectives
  - High performance
  - Low-energy consumption
  - Small footprint - low cost
  - Dependable

- Synthesis technology
  - Model HW with languages
  - Compile into masks

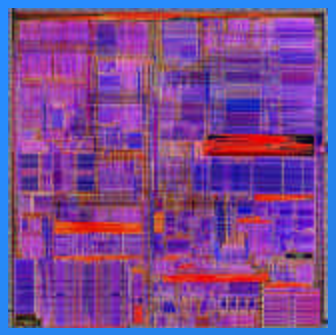


- Issues
  - Design closure
  - Handling new technologies
  - HW/SW co-design
  - Deal with multiple objectives
  - Verifying correctness
  - …

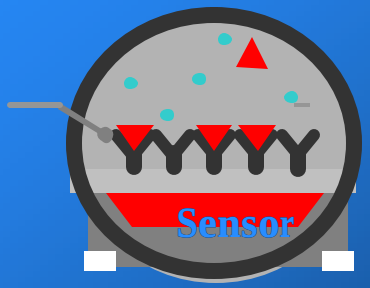
Outline

- The nanotechnology challenge
  - Variability management
  - Error tolerance
- The energy consumption challenge
  - Temperature management
- The communication bottleneck
  - *Networks on chips*
- A vision and conclusions

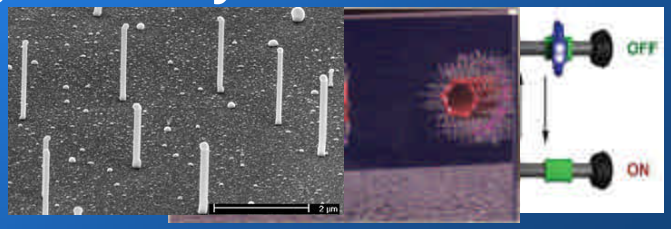
Where are we heading?



- Medium term:
  - *More Moore*
    - More scaling - More complex chips - Fewer players
  - *More than Moore*
    - Use silicon technologies beyond computational structures
    - Interaction with environment, sensors, etc…
    - System integration



- Long term:
  - A multi-furcation of Moore's scaling law beyond the 22 nanometer node
  - New technologies:
    - *There is plenty of room at the bottom*



Will a new nano-electronic technology prevail?

- The skeptical view:
  - Investments in CMOS silicon are huge
  - We will not need localized computing power beyond what is achievable with a 1 cm^2 die in 22 nm silicon CMOS
  - Wiring is the bottleneck: making transistor smaller does not help
- The optimistic view:
  - We will always need increasing computing power and storage capacity
  - We need to curb the increasing costs of manufacturing
  - We will invent new computing architectures, storage media and communication means

How is the transition path?

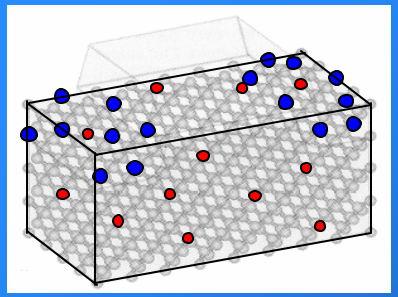
- When will current semiconductor technologies run out of steam?
- What factor will provide a radical change in technology?
  - Performance, power density, cost?
- Will the transition eliminate previous CMOS technologies?
  - Are the new nanoelectronic technologies compatible with standard silicon?
- How will we design nanoelectronic circuits:
  - What are the common characteristics, from a design technology standpoint?

Common characteristics of nano-devices

- Self-assembly can be used to create structures
  - Manufacturing paradigm is both bottom-up and top down
  - Attempt to avoid lithography bottleneck
- Combined presence of micro and nano-structures
  - Interfacing and compatibility issues
- More physical defects and higher failure rates
  - 10-15% defective devices according to recent estimates
  - Design must deal with nonworking and short-lived devices
- Advantage stems from the high density of devices
  - Two orders higher than scaled CMOS

Design issues

- Variability
  - Physical parameter variation
  - Molecular structural effects
- Reliability
  - Higher failure rate
  - Higher environmental exposure
  - Transient and permanent errors

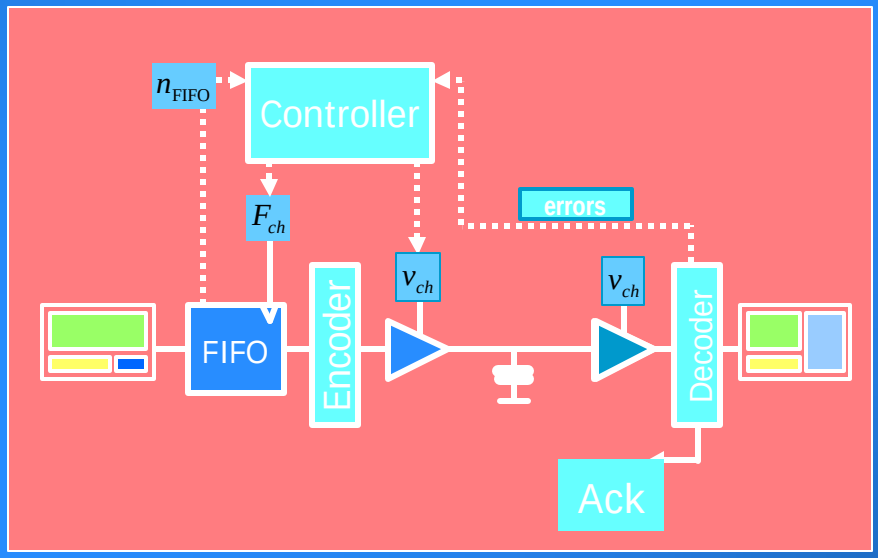


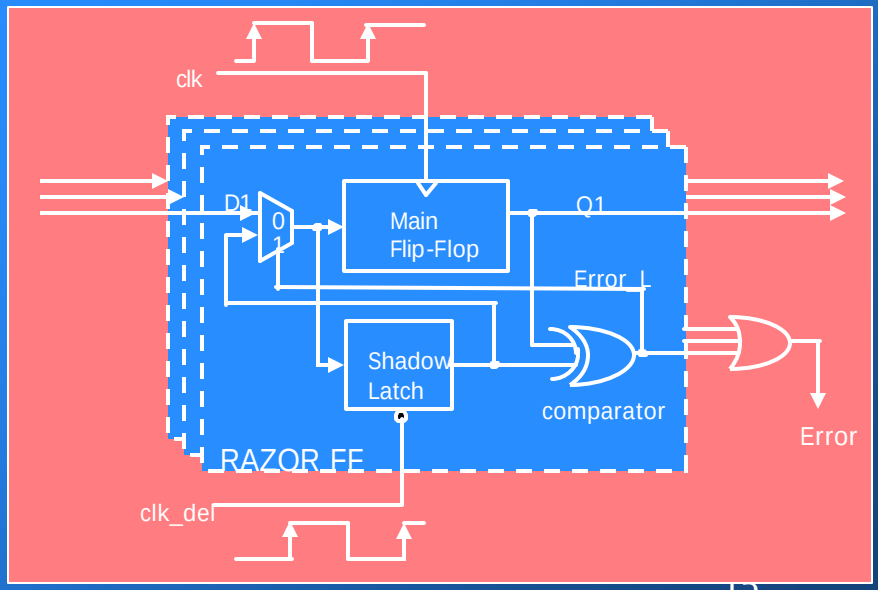
Variability

- Variations within/across chips
  - Fast/slow transistors and interconnect
- Design objective:
  - Achieve better than *worse-case performance*
- Solutions
  - Statistical timing analysis
  - Statistical logic synthesis
  - Asynchronous design
  - System-level approaches

Self-calibrating circuits

- Adapt to inter-chip variations and to environmental changes
  - Use *on-line* adaptation policy
- Examples:
  - Dynamic voltage scaling of bus swings [Worm,Ienne –EPFL]
  - Dynamic voltage scaling in processors
    - Razor [Austin – U Michigan]
  - Dynamic latency adjustment for NoCs
    - Terror [Tamhankar -Stanford]
- Autonomic computing
  - Systems that understand and react to environment [IBM]





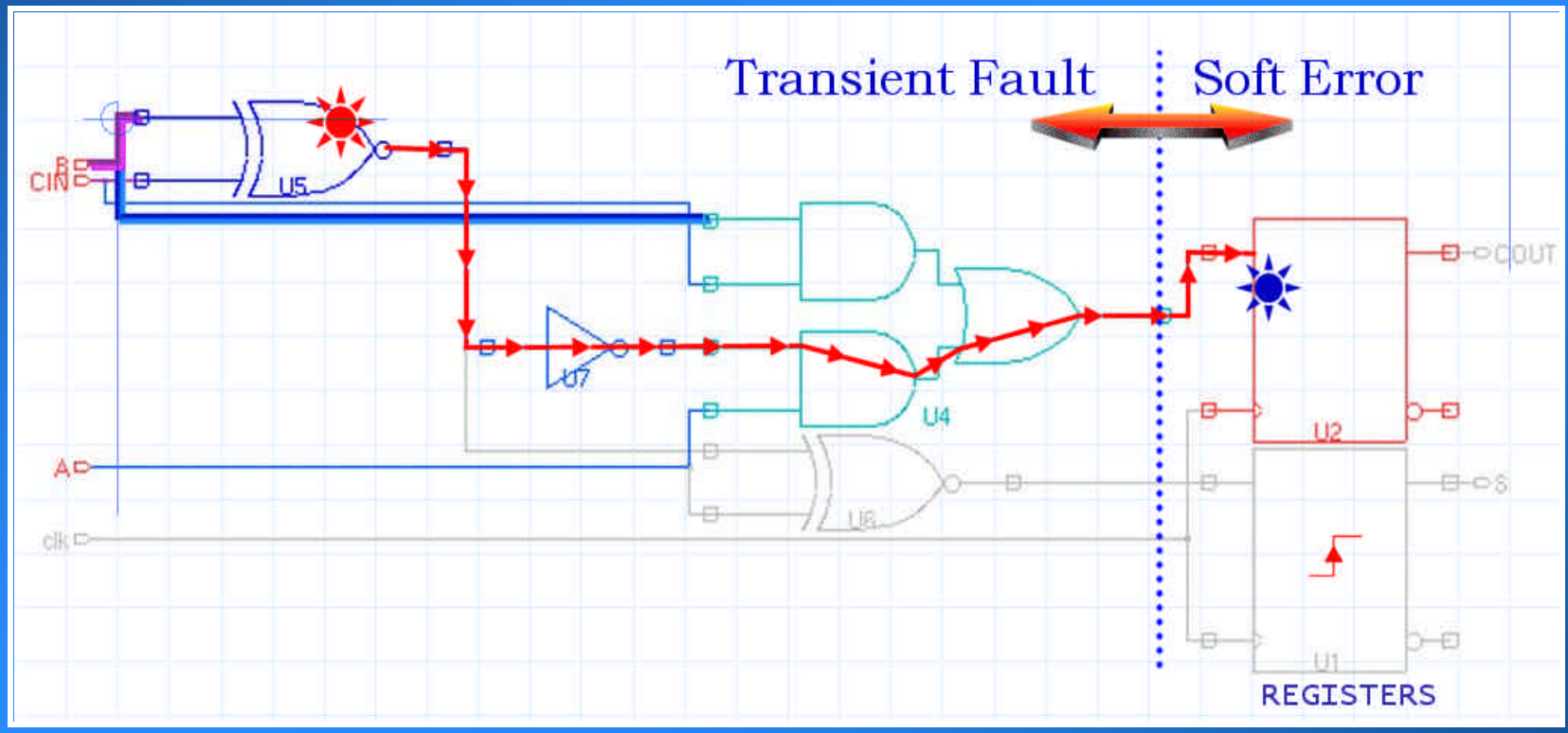
Reliability: coping with transient malfunctions

- Soft errors
  - Data corruption due external radiation exposure
- Crosstalk
  - Data corruption due to internal field exposure
- Both malfunctions manifest themselves as timing errors
  - Error containment

Soft error rates

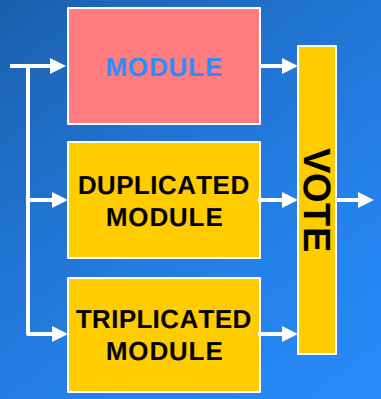
- Vary with altitude and latitude

Propagation of soft errors



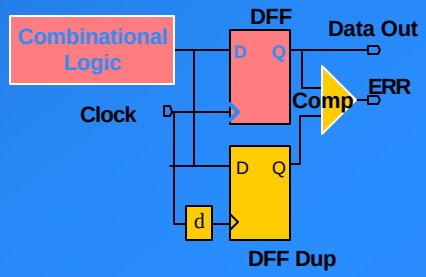
Logic protection techniques

Redundancy (TMR)



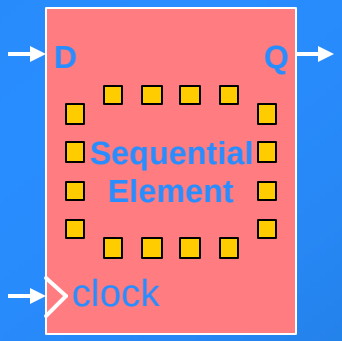
100% to 200% overhead

Detection + System Correction



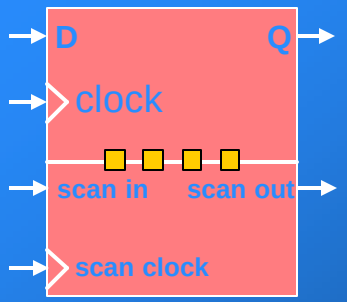
ERR signal used by system (Hardware or Software) to correct the error

Hardened Libraries



Protection Transistor embedded in the cell

Other Techniques



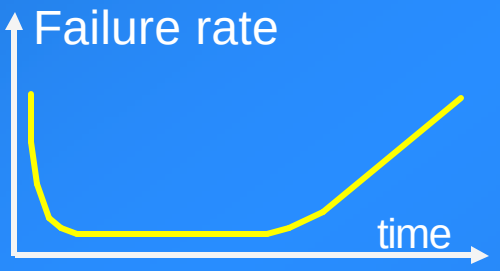
Scan Sequential Element

SCAN hardware used as DETECTION hardware in functional mode

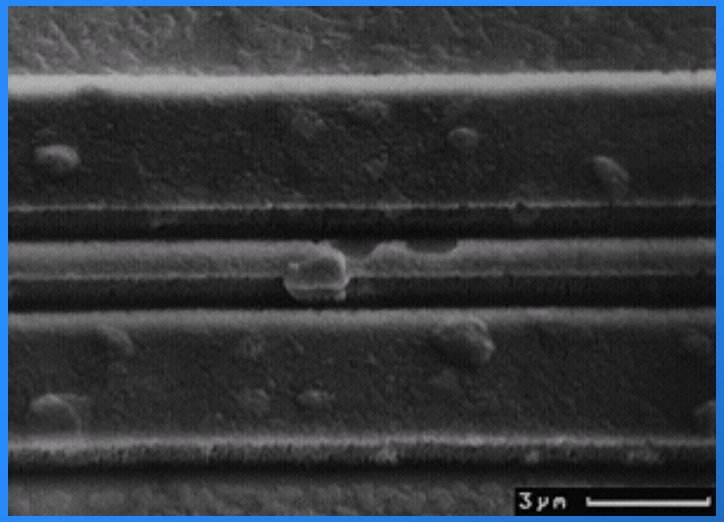
Redundancy ***Shielding*** ***Others***

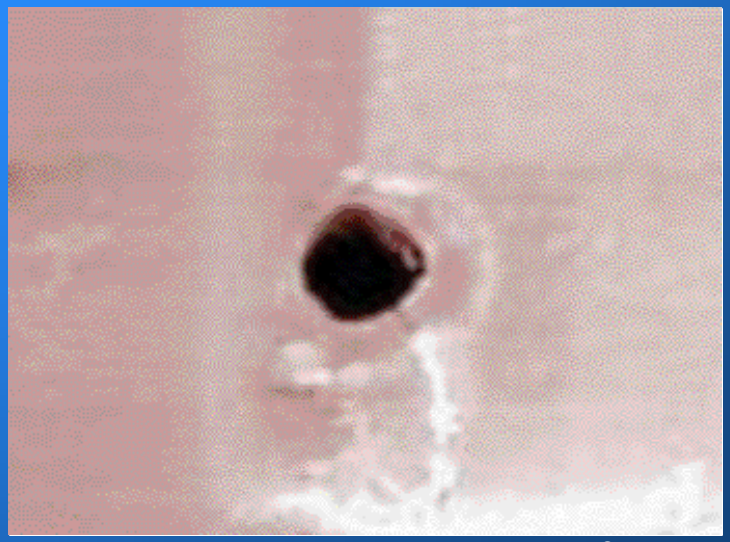
[Source IROC]

Reliability: aging of materials



- Failure mechanisms:
  - Electromigration
  - Oxide Breakdown
  - Thermo-mechanical stress
- Temperature dependence
  - Arrhenius law
  - Gradients





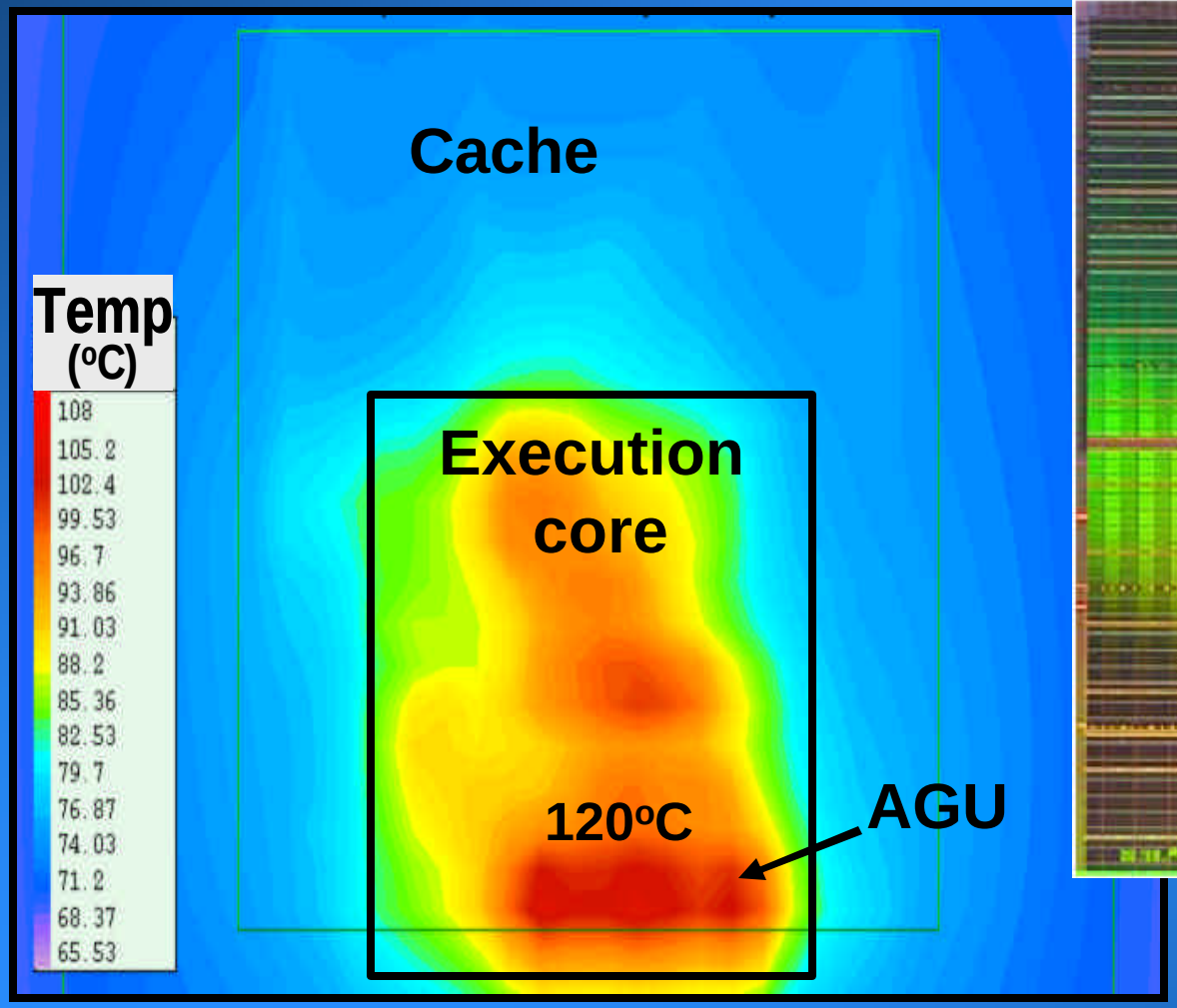
Summary:
coping with variability & reliability

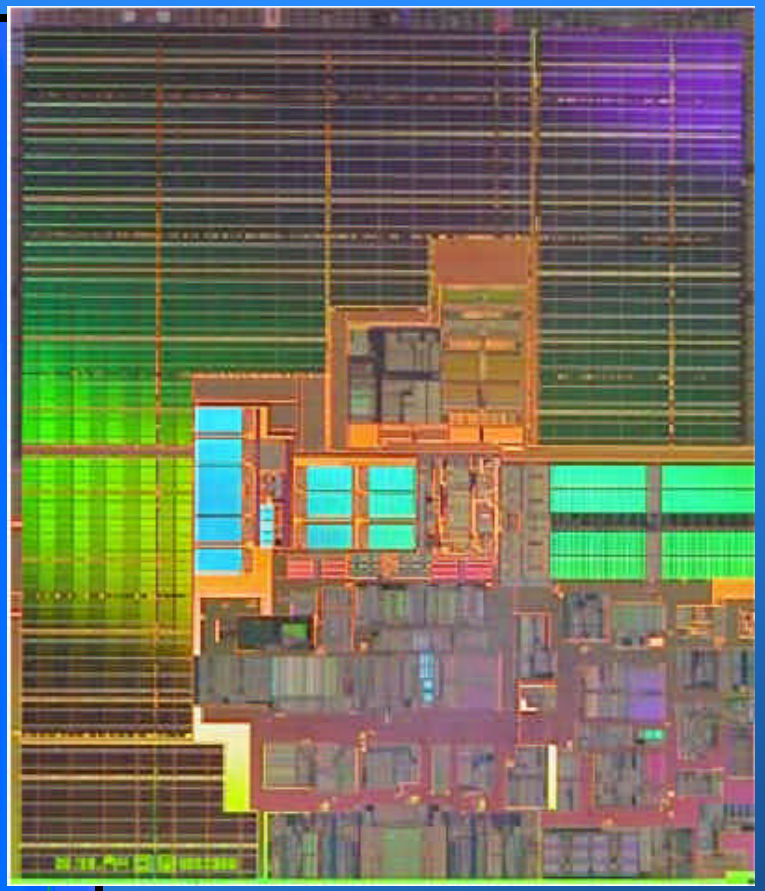
- Design chips so that they are *insensitive* to local timing variations
- Exploit *redundancy* to replace failing devices
- Manage *power consumption* and *temperature* of on-chip components

Outline

- The nanotechnology challenge
  - Variability management
  - Error tolerance
- The energy challenge
  - Temperature management
- The communication bottleneck
  - *Networks on chips*
- A vision and conclusions

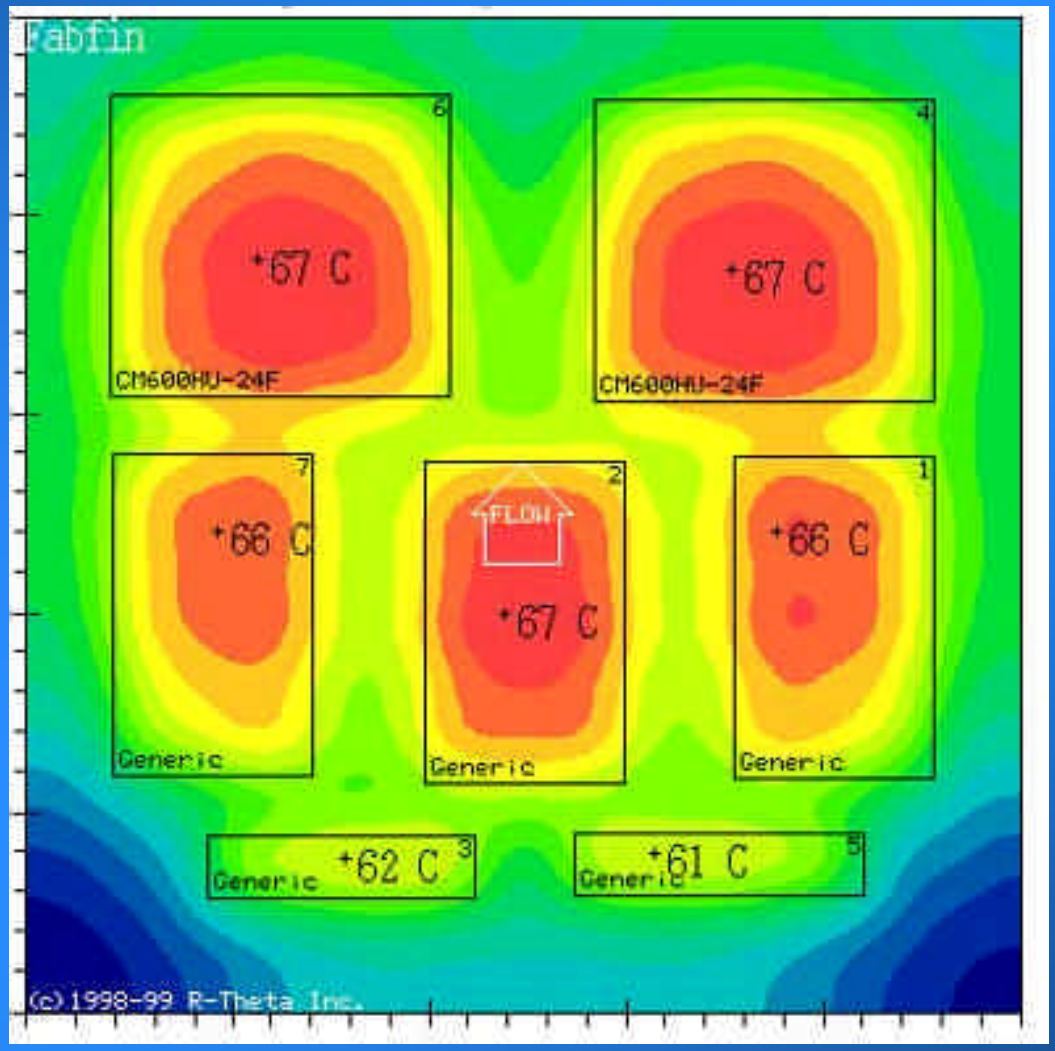
Thermal map
1.5 GHz Itanium-2



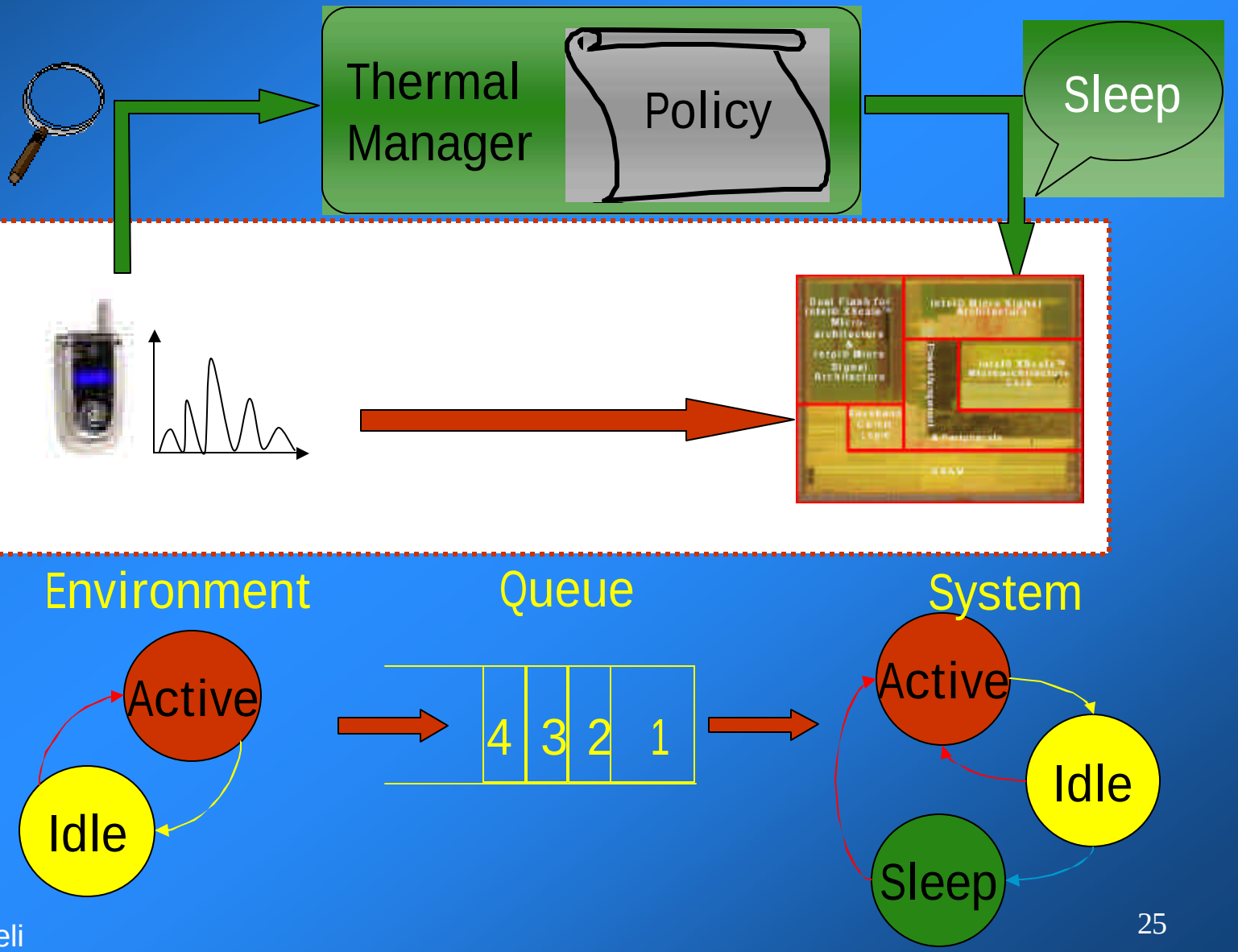


[Source: Intel Corporation and Prof. V. Oklobdzija]

Thermal map: multiprocessosr



Thermal management modeling



Thermal management policies

- Objective:
  - Increase energy efficiency and enhance reliability by controlling activity and temperature
  - Control *systems on chips* while running
- Mathematical problem:
  - Compute a policy that shuts down / slows down components
  - Policy complexity depends on requested accuracy
    - Markov, Semi-Markov, Time-Index Markov models, …
  - Under mild assumptions, the policy can be computed exactly by LP
  - Convex optimization can be used

$$\min \quad \sum_{c=1}^{N} \mathrm{cost}_{energy,\ c}$$

$$s.t. \quad \sum_{a\in A} f(s,a) - \sum_{a\in A}\sum_{s'\in S} M(s'|s,a)\, f(s',a) = 0; \quad \forall s, \forall c_s$$

$$\sum_{a\in A}\sum_{s\in S} T(s,a)\, f(s,a) = 1; \qquad \forall c_s$$

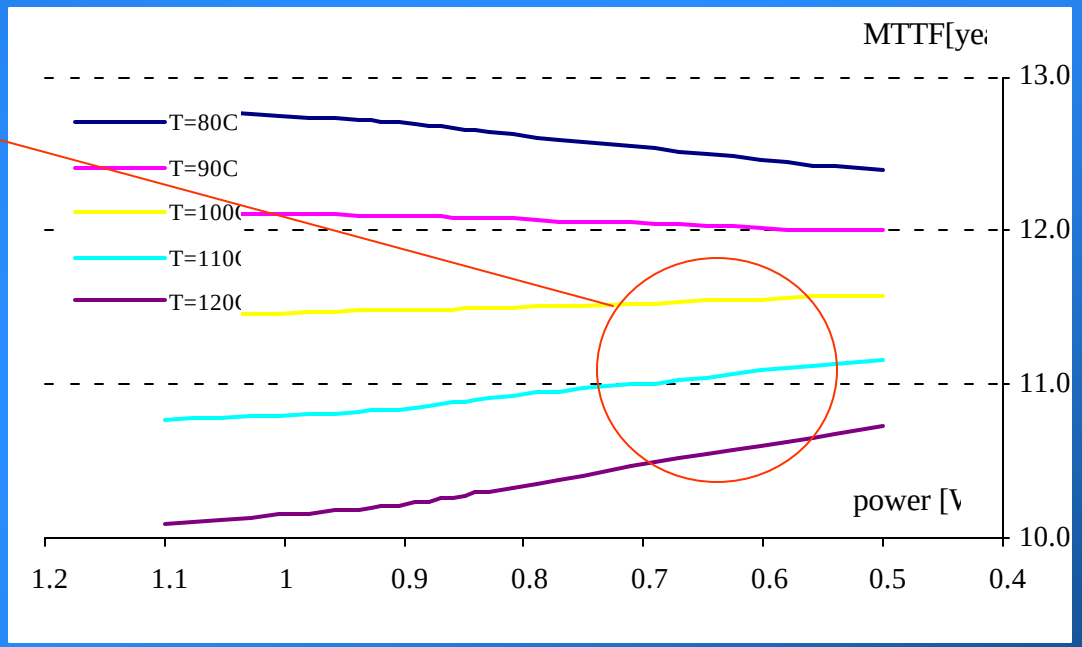
$$\sum_{c=1}^{N} \mathrm{cost}_{perf,\ c} < Perf_{const}; \qquad \forall c$$

$$Tpl\,(l_c) \le \mathrm{Re}l_{const}; \qquad \forall c_s$$

$$l_c = \sum_{i\in F}\sum_{a\in A}\sum_{s\in S} l^{i}_{core}\,(s,a)\, y(s,a)\, f(s,a)$$

Effect of power management policy on MTTF

- At high temperatures, EM and breakdown dominate
  - DPM helps reliability
- At low temperatures, thermal stress dominates
  - DPM lowers MTTF
- As the temperature gap between the active and sleep state widens, thermal stress tends to dominate



- Any management policy must address both power consumption and temperature management

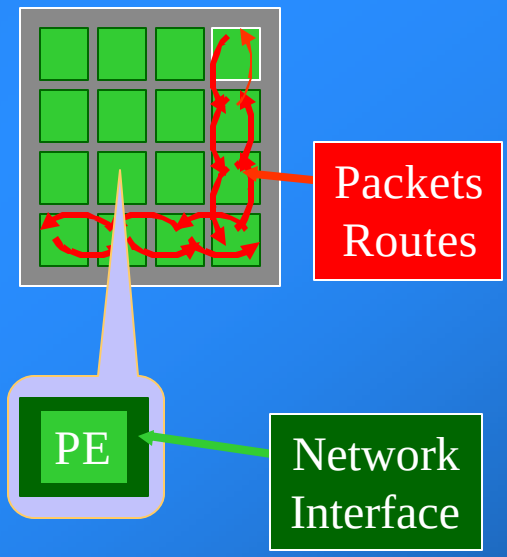
Summary: power and thermal management

- Power and temperature management addresses:
  - Battery lifetime extension for mobile systems
  - Component lifetime extension because of reliability enhancement
- Management policies become increasingly more important as geometries scale down:
  - More devices dissipate more power (per unit area)
  - Smaller devices are more prone to fail

Outline

- The nanotechnology challenge
  - Variability management
  - Error tolerance
- The energy challenge
  - Temperature management
- The communication bottleneck
  - *Networks on chips*
- A vision and conclusions

Why *on-chip networks?*

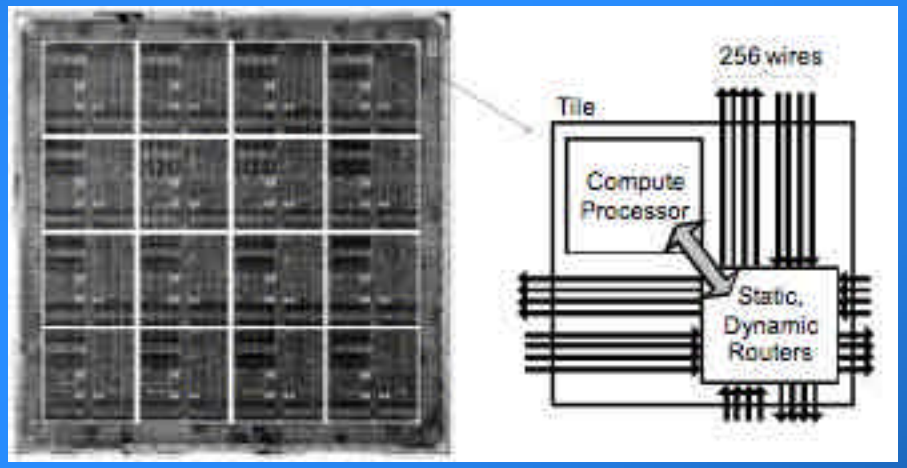


- Provide a structured methodology for realizing on-chip communication
  - Modularity
  - Flexibility
- Cope with inherent limitations of busses
  - Performance and power of busses do not scale up
- Support reliable operation
  - Layered approach to error detection and correction

NoC multi-processors: the RAW architecture

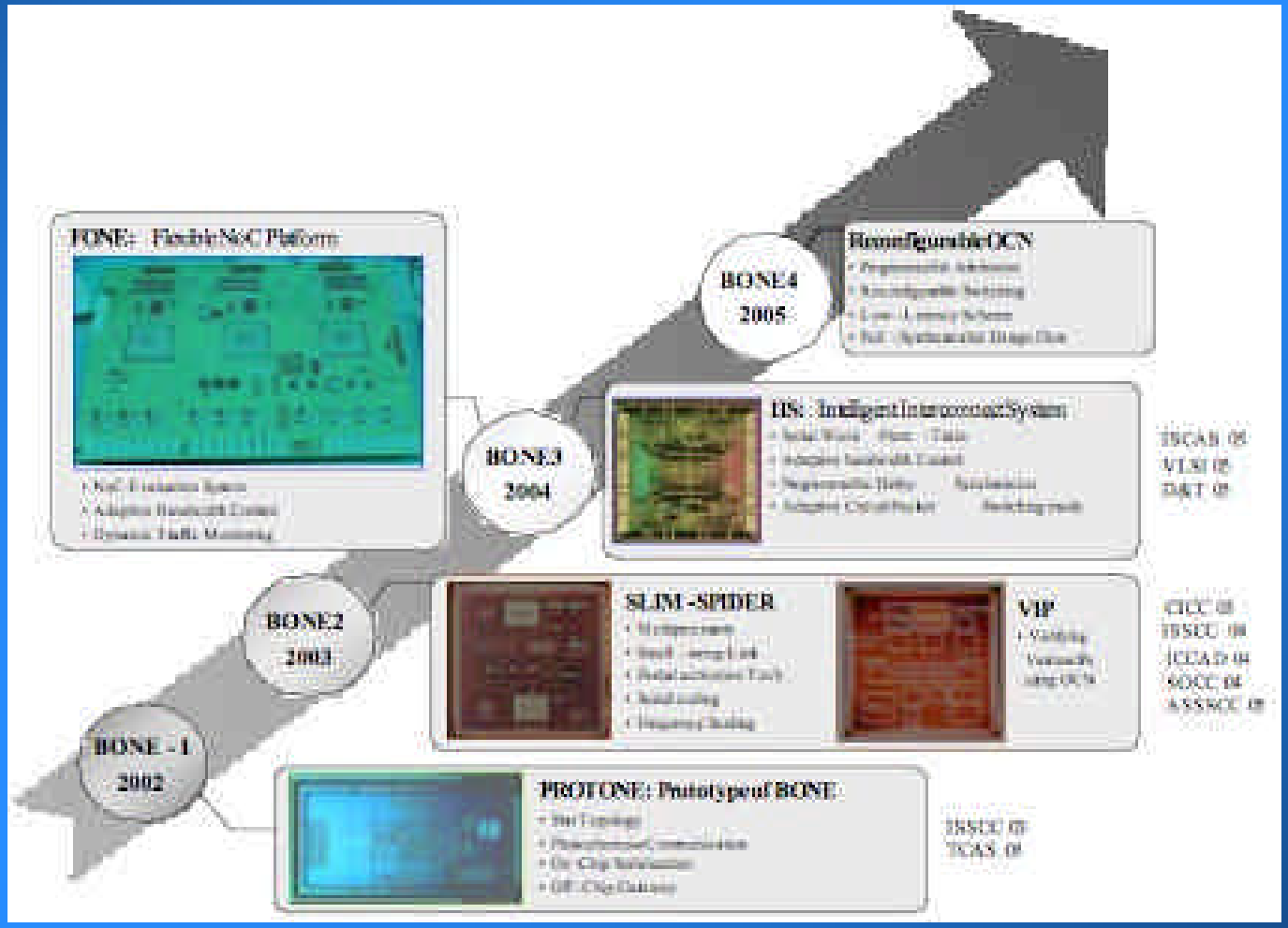
- Fully programmable SoC
  - Homogenous array of tiles:
    - Processor cores with local storage
    - Each tile has a router

    [Agrawal MIT]



- The raw architecture is exposed to the compiler
  - Cores and routers are programmable
  - Compiler determines which wires are used at each cycle
  - Compiler pipelines long wires

The BONE roadmap



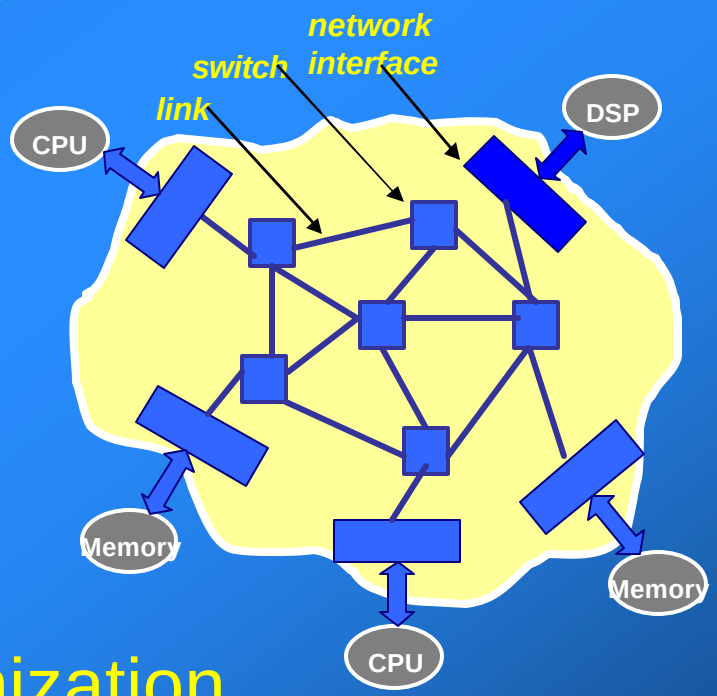
[Source: KAIST]

Metrics for NoC design

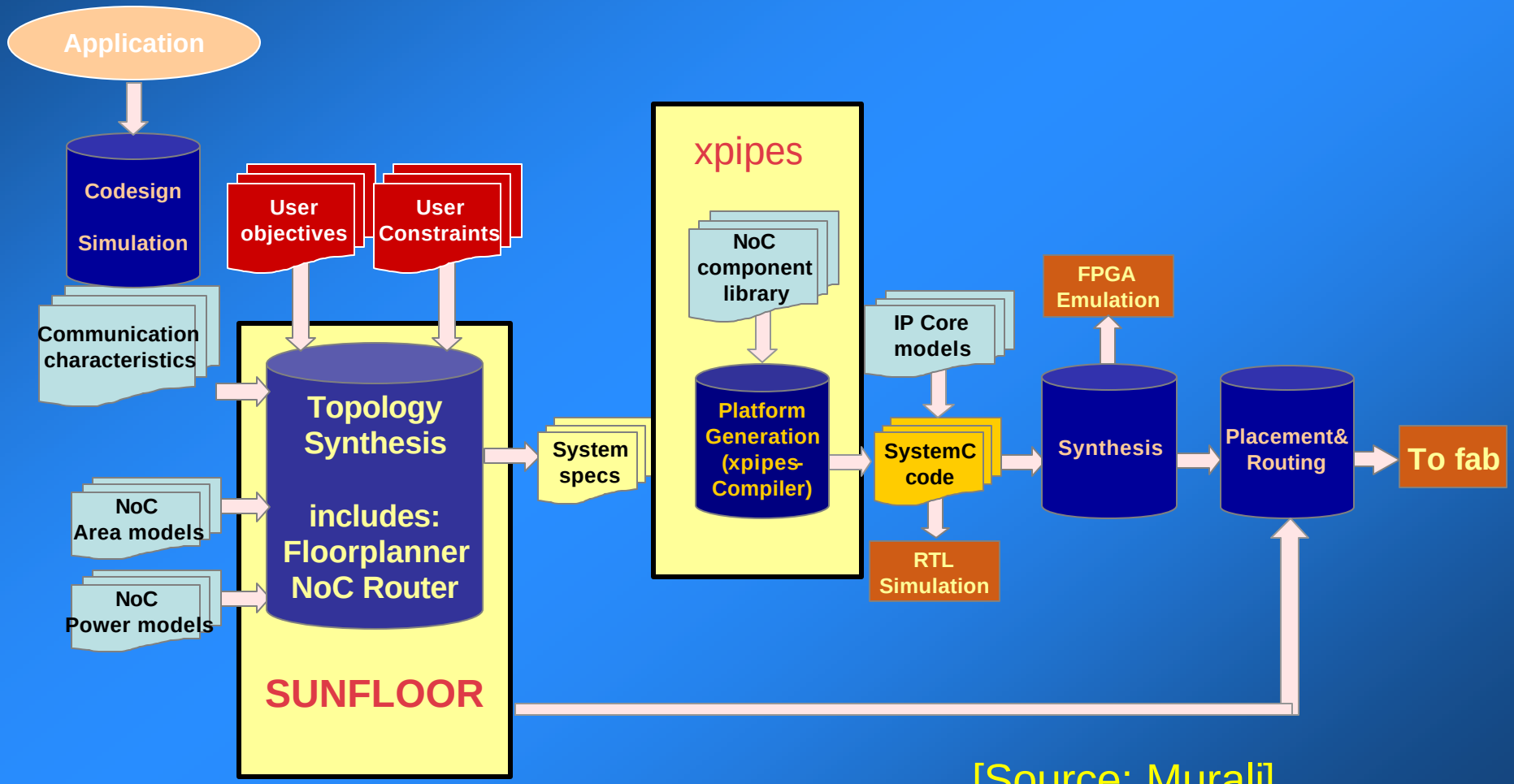
- Low communication latency
  - Streamlined control protocols
  - Data and control signals can be separate
- High communication bandwidth
  - To support demanding SW applications
- Low energy consumption
  - Wiring switched capacitance dominates
  - Local storage in register buffers is expensive
- Error resiliency
  - To compensate/correct electrical-level errors
- Flexibility and programmability

Flexibility in NoC design

- NoCs have modular structure
  - Core interfaces
  - Switches/routers
  - High-speed links
- NoCs can be tailored to applications
  - Topology selection
  - Switch/link sizing
  - Protocols
- Several parameters for optimization and a large design space
  - NoC synthesis and optimization



Netchip tool flow



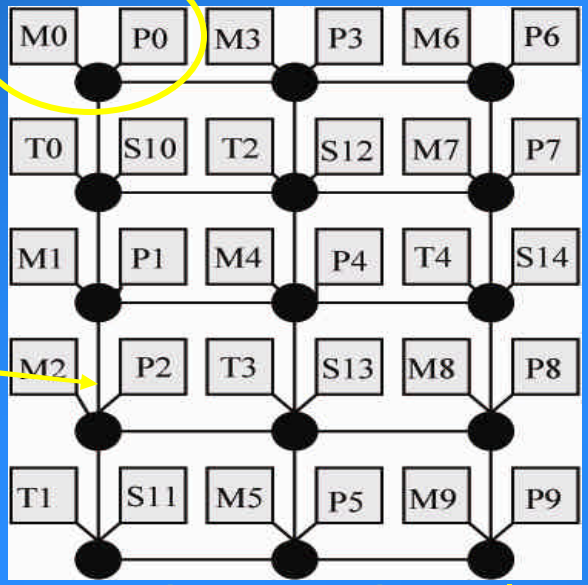
[Source: Murali]

SUNFLOOR vs. manual design

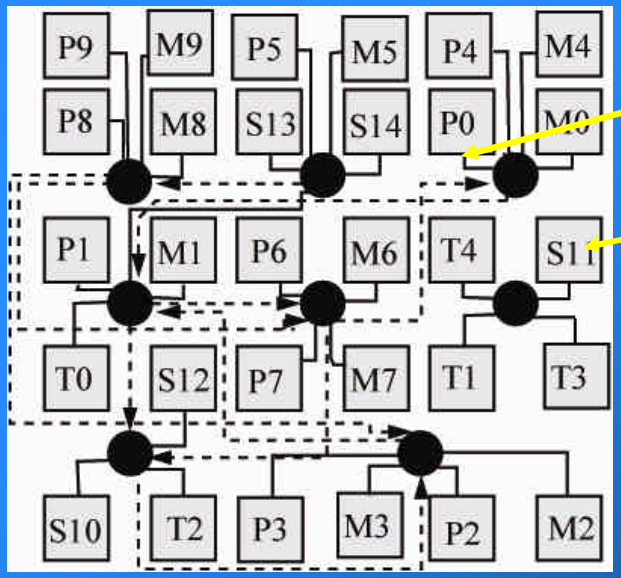
multimedia chip with 30 cores

Processor-memory cluster

Bi-directional links



Hand-mapped topology



Uni-directional links

Bi-directional links

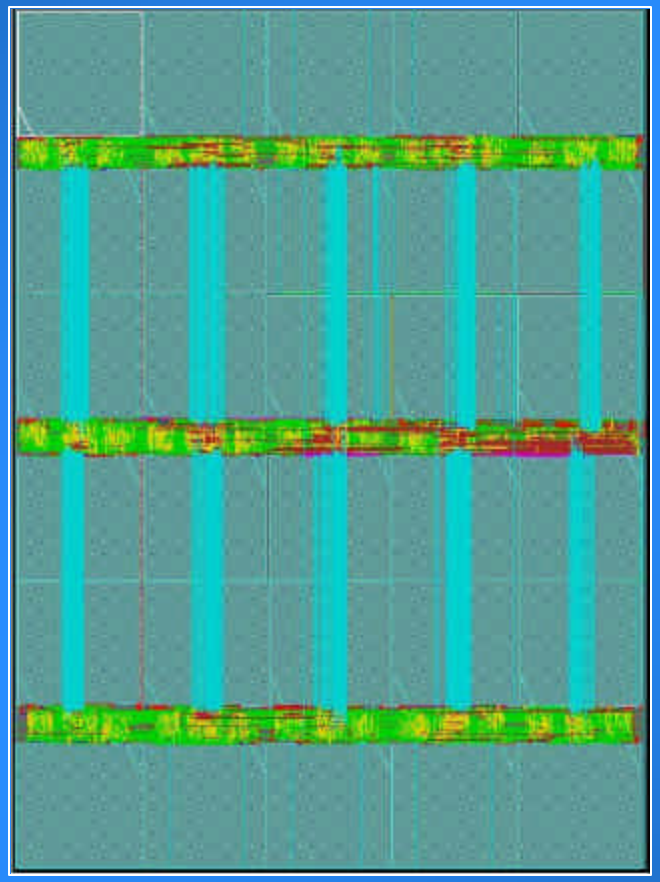
SUNFLOOR custom topology

P-processors, M-private memories,

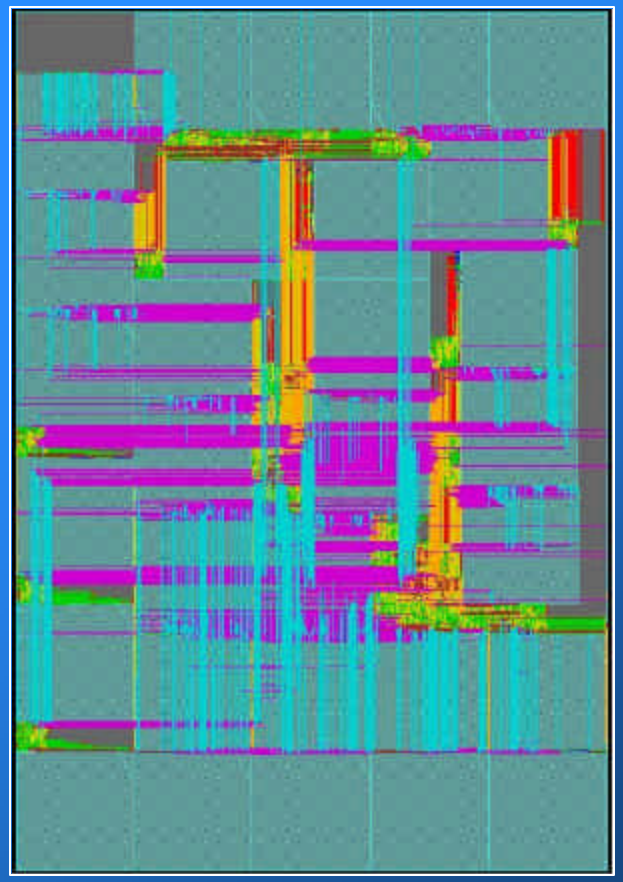
T-traffic generators, S-shared slaves

Design layouts

From Cadence SoC Encounter



Hand-design (custom mesh)



SUNFLOOR Design

SUNFLOOR vs. manual design

Manual design:

- Topology: 5x3 mesh (15 switches)
- Operating frequency: 885 MHz (post-layout)
- Power consumption: 368 mW
- Floorplan area: 35.4 mm^2
- Design time: several weeks
- 0.13 µm technology

SUNFLOOR:

- Topology: custom (8 switches)
- Operating frequency: 885 MHz (post-layout)
- Power consumption: 277 mW **(-25%)**
- Floorplan area: 37 mm^2 **(+4%)**
- Design time: **4 hours design to layout**
- 0.13 µm technology

Benchmark execution times comply with application requirements and, in fact, are even 10% better on the SUNFLOOR topology

Summary: *networks on chips*

- *Networks on Chips* are the structured interconnect of the future
  - NoCs exists in many forms and flavors
- *Networks on Chips* are necessary for chips designed in 45 nm technology and beyond
  - NoCs deal with wire delay variability
  - NoCs provide reliability enhancement

Outline

- The nanotechnology challenge
  - Variability management
  - Error tolerance
- The energy challenge
  - Temperature management
- The communication bottleneck
  - *Networks on chips*
- Vision and conclusions

A vision for the future

- Mobile, ubiquitous, pervasive computing
  - Ultra low-power demands *low-voltage* operation
- High performance requires *parallel* computation
- High reliability is achieved by *redundancy*
- New paradigm for computation:
  - Array-based computation (e.g., RAW)
  - Array-oriented communication (NoC)

Conclusions

- High-performance, ultra low-power, reliable circuits will be required by distributed embedded systems
- Novel nanotechnologies will provide us with unprecedented levels of functional integration and performance
- Novel design tools and methodologies will be needed to leverage the technology:
  - Cope with variability, reliability, thermal and other issues

It is only a beginning …
… the challenges are still ahead

[Source: Kubrick]