SYSTEMS and NETWORKS on CHIP
Challenges and Solutions

Giovanni De Micheli
Centre Systèmes Intégrés
What is a System on Chip?

- What is a system?
Why on one chip?

- Cost reduction
- Integration issues
Multi-processor SoCs

- Large-scale
  - Billion-transistor chips
  - Multi-cores, multi-threaded software
- Power-consumption limited
- Very expensive to design
  - Non recurring engineering costs
  - Migration toward software

IBM Cell Multi-Processor
Application-specific SoCs

- Multi-core systems
- Embedded memory
  - Embedded software
- Heterogeneous structure
  - Flexibility vs optimality
- Example:
  - Telecom EDGE baseband and multimedia handset [STM]
Platforms

• Economic viability requires large production volumes
  – Domain-specific hardware
  – Differentiation via software

• Examples
  – Telecom:
    • Philips Nexperia
    • ST Nomadic
  – Automotive
Where are we heading?
Outline

• Evolutionary technologies
  – Variability and reliability
  – Thermal management and networking
• Revolutionary technologies
  – Nano and molecular electronics
• Summary and conclusions
[Diskobolos - Myron: 460BC]
Medium term trends

• Feature size downscaling
• Increasing transistor density and clock frequency
  ➢ Power and thermal management
• Lower supply voltage
  ➢ Reduced noise immunity
• Increasing the spread of physical parameters
  ➢ Inaccurate modeling of physical behavior
  ➢ Variability and reliability
Variability: physical motivation

Dopant Atoms

20 nm MOSFET (2010 ?)
50 Si atoms along the channel

4 nm MOSFET (2020 ?)
10 Si atoms along the channel
Prediction vs reality

- Models used for design are not accurate enough
- Statistical spreading makes design over-conservative

De Micheli
Design space exploration
worst case analysis

Pareto points on w.c. curve

Delay

min
typ
max

Voltage
Adaptive design space
worst case analysis

As parameters spread, w.c. design is too pessimistic
Self-calibrating circuits

- Address variability and robustness
- Design self-calibrating circuits operating at the edge of failure
- Examples:
  - Dynamic voltage scaling of bus swings [Ienne – EPFL]
  - Dynamic voltage scaling in processors
    - Razor [Austin – U Mich]
  - Dynamic latency adjustment for NoCs
    - Terror [Stanford]
Outline

• Evolutionary technologies
  – Variability and reliability
  – Thermal management and networking

• Revolutionary technologies
  – Nano and molecular electronics

• Summary and conclusions
De Micheli [Laocoonte: Hagesandros, Athanadoros and Polydoros: 1st century BC]
Dealing 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
Sources of soft errors

Strong (nuclear) interaction

Electromagnetic interaction (Silicon reaction)

Solar wind (source)

Neutrons @ sea level

Silicon

Ions

25Mg+α
28Al+p
24Mg+n+α

Transistor

Electrons

Transient pulse

Silicon reaction

25Mg+α
28Al+p
24Mg+n+α

De Micheli
Soft error rates

- Vary with altitude and latitude
Propagation of soft errors
De Micheli

Memory protection techniques

Standard ECC

Low Area ECC

High Speed ECC

Embedded Protection

System Level Protection Techniques

Error Correcting Codes

[Source IROC]
Logic protection techniques

Redundancy (TMR)

- MODULE
- DUPLICATED MODULE
- TRIPlicated MODULE

100% to 200% overhead

Detection + System Correction

- Combinational Logic
- DFF
- Clock
- Data Out
- ERR
- Comp

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

Hardened Libraries

- DFF Dup
- Q

Protection Transistor embedded in the cell

Other Techniques

- Scan Sequential Element
- Scan in
- Scan out
- Scan clock
- D
- Q
- clock
- D
- Q

SCAN hardware used as DETECTION hardware in functional mode

[Source IROC]
Outline

• Evolutionary technologies
  – Variability and reliability
  – Thermal management and networking
• Revolutionary technologies
  – Nano and molecular electronics
• Summary and conclusions
[Charioteer of Delphi - Sotades: 474 BC]
Aging of materials

- Failure mechanisms
  - Electromigration
  - Oxide Breakdown
  - Thermo-mechanical stress

- Temperature dependence
  - Arrhenius law
Thermal maps
1.5 GHz Itanium-2

Cache

Execution core

AGU

120°C

Temp (°C)

[Source: Intel Corporation and Prof. V. Oklobdzija]
Thermal maps multiprocessor
Thermal effects

• Keep chip as cool as possible
  – Reduce failure rates and power consumption
• In multi processor (core) system, power management shuts down idle cores
  – The temperature distribution will change in time
  – Thermal stress may increase
• Balance temperature reduction and thermal stress
Component redundancy

• Use stand-by components to replace faulty ones
  – Provide for temporary or permanent back-up
  – Exploit power/thermal management
  – Load sharing
• A programmable and flexible interconnection is required
From power to system management

• Analyze system-level reliability
  – as a function of a power management policy

• Determine a system management policy
  – to maximize reliability (over a time interval) and minimize energy consumption

• Determine a system management policy and system back-up topology
  – to maximize reliability (over a time interval) and minimize energy consumption
System-level management

- Reliability and energy management can be modeled by stochastic processes
  - Stochastic optimum control for policy design
  - As more accurate models are required, policy design is harder
- Simulation of system management policies is useful for assessing effectiveness of redundancy and energy cost
  - Simulation results show dominant effect of temperature and its cycling on system reliability
- Optimal policy design is also possible
Effect of DPM policy on MTTF

- Power and temperature gap between active and sleep state
  - Small gap
    - Thermal cycle effects dominate EM and TDDB only in the lower temperature spectrum
    - MTTF decreases/increases as DPM gets more aggressive
  - Wider gap
    - Thermal cycle effects dominate
    - MTTF decreases always as DPM gets more aggressive

[Simunic – UCSD]
Outline

• Evolutionary technologies
  – Variability and reliability
  – Thermal management and networking
• Revolutionary technologies
  – Nano and molecular electronics
• Summary and conclusions
Component-based SoC design

- SoCs are designed (re)-using large macrocells
  - Processors, controllers, memories…
  - *Plug and play* methodology is very desirable
  - Components are qualified before use

- Design challenge:
  - Provide a **functionally-correct, reliable operation** of the interconnected components

**Critical issue:**
- Design of the communication fabric
Example:
Motorola MSC8126 platform

- Communication scalability is the bottleneck
Entire Chip is Not Reachable in One Clock Cycle!

[Source: Leblebici]
Why on-chip networking?

- 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 [MIT]

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

• 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

[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
  – Great match to stream computing

• Low energy consumption
  – Wiring switched capacitance dominates

• Error resiliency
  – To compensate/correct electrical-level errors

• Flexibility and programmability
Error resiliency

- Several implementation styles:
  - Local link-level
    - ECC in switches
  - Global *end to end*
    - ECC at core interfaces
  - Transaction level
    - Software approach
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
xPipes NoC Synthesis Flow

- Support for several topologies and routing functions
- Area, power, delay optimization
Comparisons

- 130nm UMC library
- Cores: 1mm² obstructions (ARM cores, 32kB SRAM)
- Wire routing over the cores was forbidden

Summary of results:
- 2.7% vs. 17% post P&R timing degradation → Much improved physical scalability
- Clock frequency 885 vs. 400 MHz → Much faster
- 16% application speedup (longer latency, but more effective bandwidth)
- 7x more area and 5x more power (mostly due to flip-flops in buffers)
- Overall better energy efficiency for >4 Watts Proc&Mem power

Predictability is highly enhanced
Efficiency: competitive with state-of-the art interconnect even in 130nm technology

[Source: Benini]
Outline

• Evolutionary technologies
  – Variability and reliability
  – Thermal management and networking

• Revolutionary technologies
  – Nano and molecular electronics

• Summary and conclusions
[Scopas: Torso]
New computing materials

• When will current semiconductor technologies run out of steam?
• What factor will provide a radical change in technology?
  – Performance, power density, cost?
• Several emerging technologies:
  – Silicon nanowires, carbon nanotubes, single-electron devices, molecular switches, quantum devices, biological computing, …
• Are these technologies compatible with silicon?
  – What is the transition path?
• What are the common characteristics, from a design technology standpoint?
Common characteristics of nano-devices

• Self-assembly used to create structures
  – Manufacturing paradigm is bottom-up

• Significant presence of physical defects
  – Massively fault-tolerant design style

• Competitive advantage stems from the high density of computing elements
  – Two orders up as compared to scaled CMOS
Nanoscale and molecular electronics

• Nano-scale electronics
  – Includes scaled-down traditional CMOS
  – Challenges induced by nanometric scale
  – Scaling limit?
• Molecular electronics
  – Devices exploit molecular structure
• There is plenty of room at the bottom
Micro/Nano scale compatibility

PolySi nanowire integrated with CMOS

[Source: Leblebici]
Reliable nano-design

- Massive parallelism and redundancy
  - Local and global configuration

- Regular layout
  - Exploit properties of crosspoint architectures
    - E.g., Programmable Logic Arrays (PLAs)
  - Wiring delay are predictable
  - Regular structure support redundant logic design
Reliable nano-design: logic synthesis

- Device level redundancy
  - Duplicate transistors to achieve broader coverage
  - Cover Boolean implicants more than once
- New paradigm for testing
  - Circuit with faulty devices may still be OK
  - Exploit, rather than remove, redundancy
- Objective is overall yield
Reliable nano-design
Weighted Averaging

Fault tolerant architecture based on multiple layers

General weighted averaging and re-scaling function used in the third layer

\[ y = \frac{V_{fs}}{\sum_i k_i} \sum_i k_i x_i \]

[Leblebici and Schmid]
Architectural implications

- Modularity, redundancy, regularity
- Cellular approach to computation
  - Cellular nonlinear networks
  - Stream computing
- Programming paradigms
  - Designer need to think “parallel” to exploit these architectures at best
Outline

• Evolutionary technologies
  – Variability and reliability
  – Thermal management and networking
• Revolutionary technologies
  – Nano and molecular electronics
• Summary and conclusions
Summary and conclusions

• The electronic market is driven by embedded applications where performance and reliability are key figures of merit

• Hardware systems are more prone to fail
  – Variations in manufacturing
  – Hard and soft malfunctions

• Reliability can be enhanced by component and communication redundancy
  – System management is critical for long-lasting operation
  – On-chip networks support redundancy

• Massive parallelism and redundancy are key to design highly-dependable circuits with nano-technologies
  – Sub 45nm CMOS technologies
  – Novel silicon and non-silicon based nano-technologies
Thank You