



*The Engine of SOC Design*

# Everything you know About Microprocessors is Wrong!

Steve Leibson  
Technology Evangelist  
Tensilica, Inc.  
[sleibson@tensilica.com](mailto:sleibson@tensilica.com)

October 2, 2007



# Everything you know about microprocessors and systems design is wrong

- **Moore's Law**
  - It no longer gives us faster chips that consume less energy each year
- **Processors**
  - They're no longer expensive or slow
- **Amdahl's Law**
  - Does not apply to embedded system design
- **Wires (no longer fast)**
  - Therefore, buses are no so good for on-chip use
- **We need more design verification**
  - Soon, there won't be enough engineers in the world to verify the largest designs

- **New Imperatives for SOC Design**
- **CPUs, Complexity and Efficiency**
- **Inside the SOC: Control vs. Data-plane Processors**
- **SOC Processor Convergence**
- **MPSOC Examples**
- **Tools for Unified SOC Design**
- **The Efficiency-Generality Paradox**
- **Closing Thoughts**

**Change #1:** Products go mobile, connected, always-on, and media-rich, placing more demands on performance

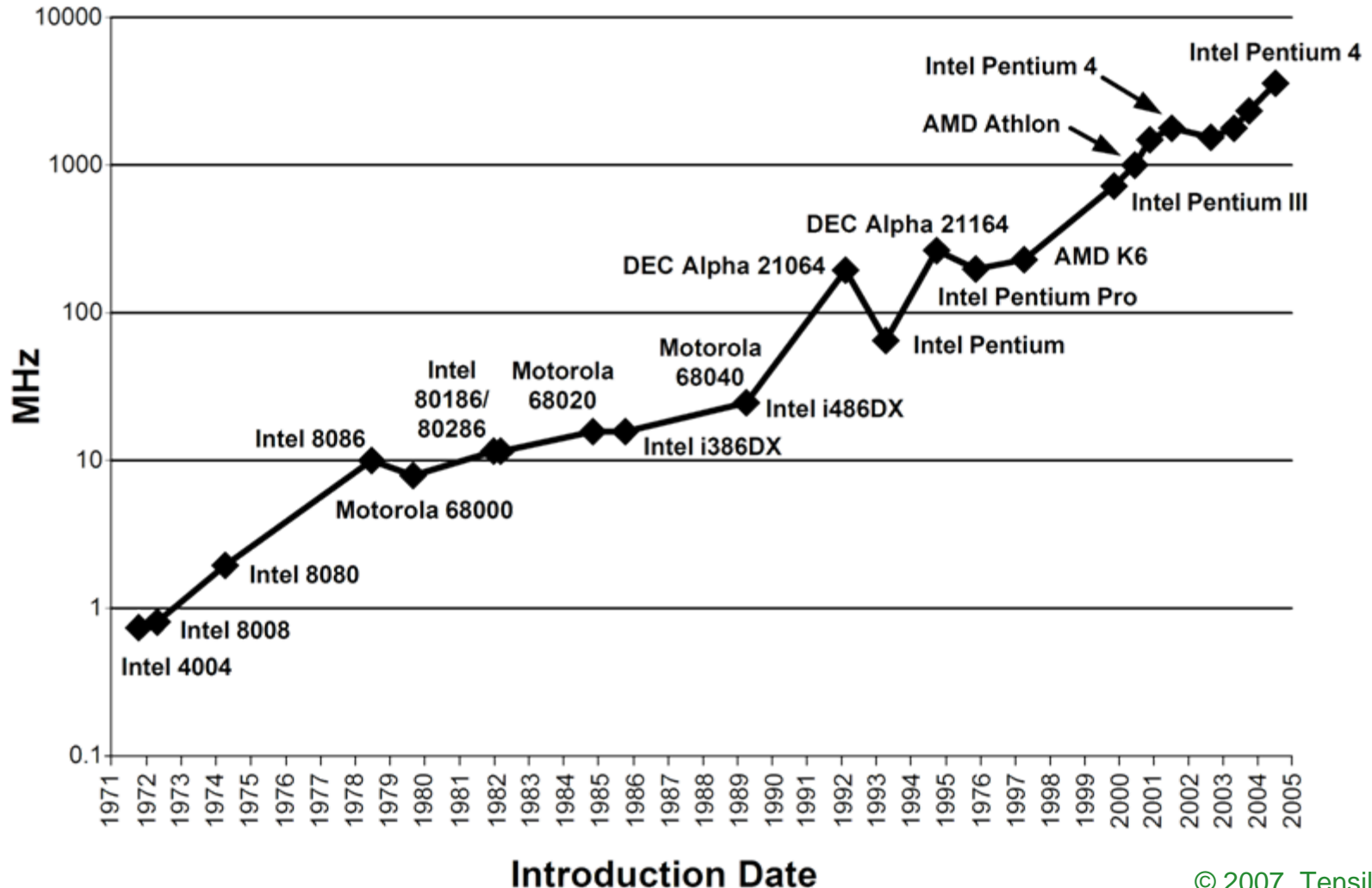
**Change #2:** Increasing product complexity drives up development cost and risk while volume per IC design goes down

**Change #3:** Moore's Law drives lower silicon cost but no longer delivers lower power (Denard scaling died at 90nm)

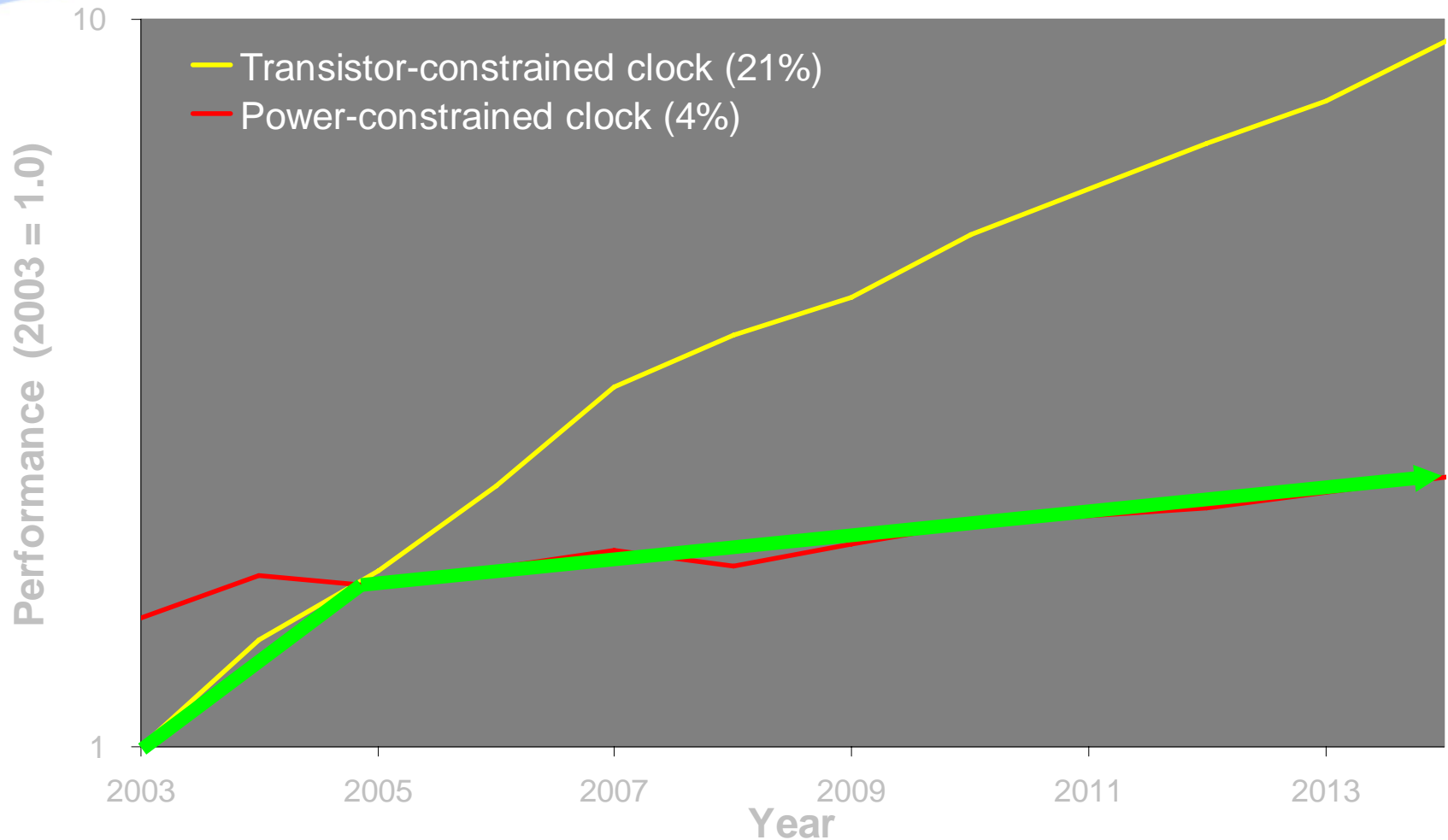


# Rise in Packaged Microprocessor Clock Rate Over Time

Microprocessor Clock Rate over Time



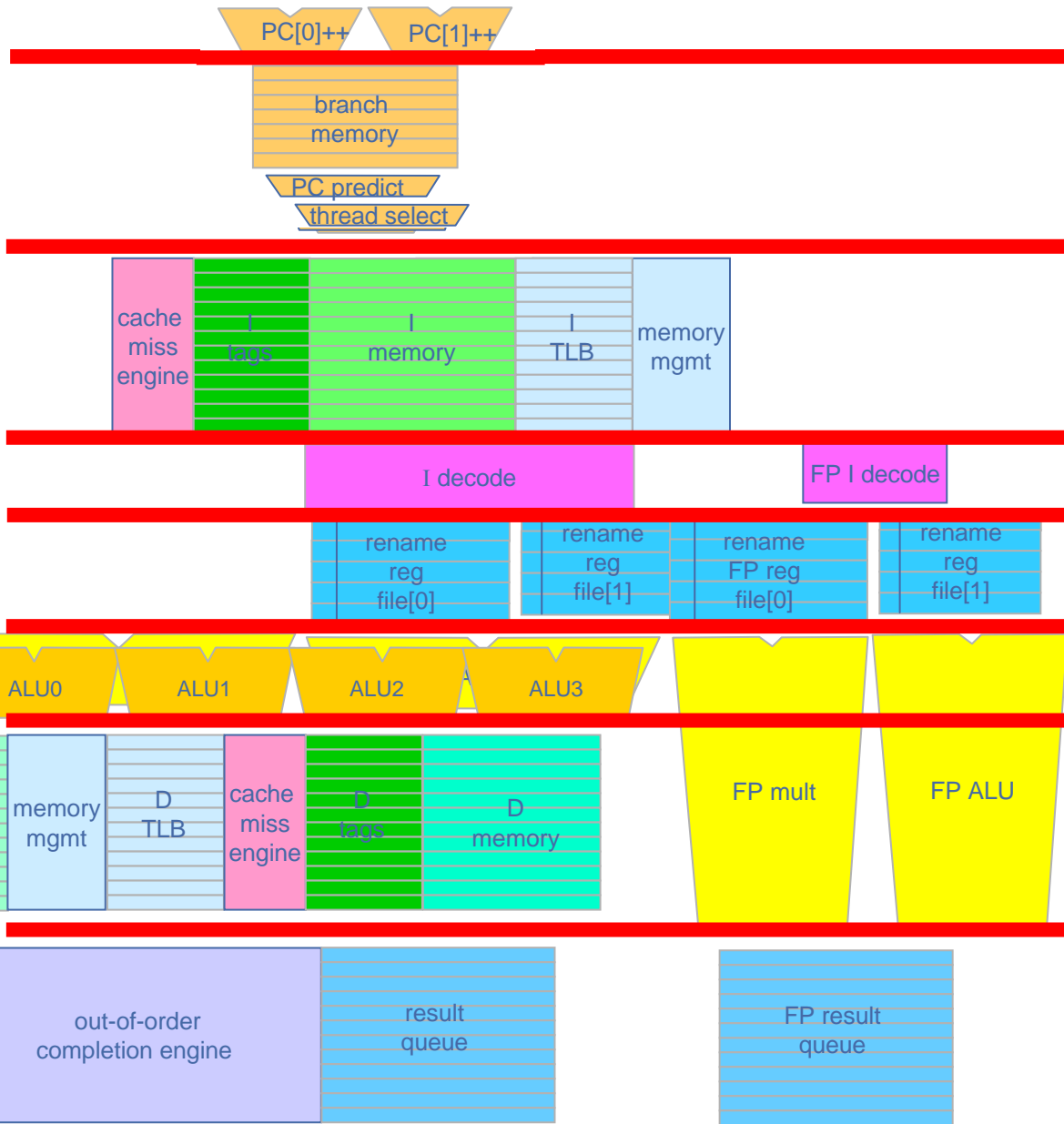
# Clock Frequency Inflection Point



# Relative Impact of Processor Features

- thread integer performance
- processor area and power

- Basic micro-controller
- Micro-code
- Data width:4→8→16→32...
- General register file
- Cache and memory protect
- Pipelined load/store arch
- Floating point
- Superscalar (static or dyn.)
- SIMD multimedia ALU
- Branch prediction
- Symmetric multi-processing
- Out-of-order execution
- Simultaneous multi-thread

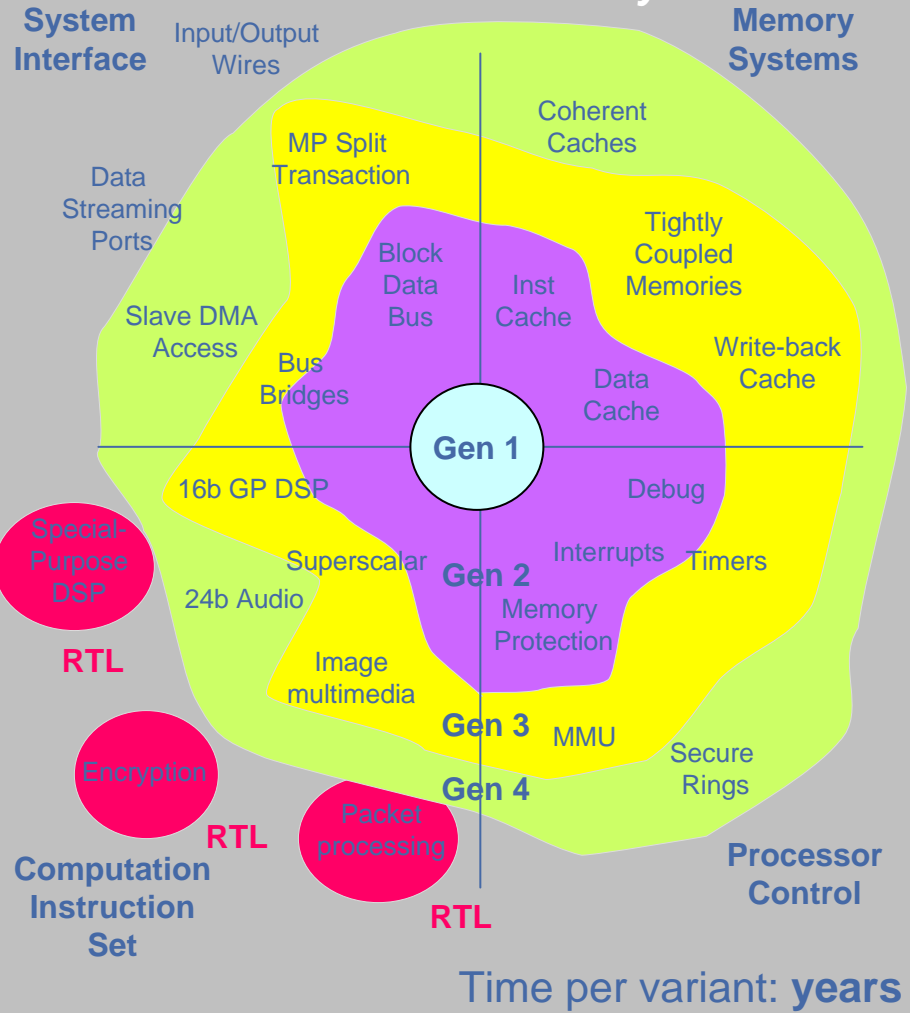


*complex*  
**The history of microprocessors**

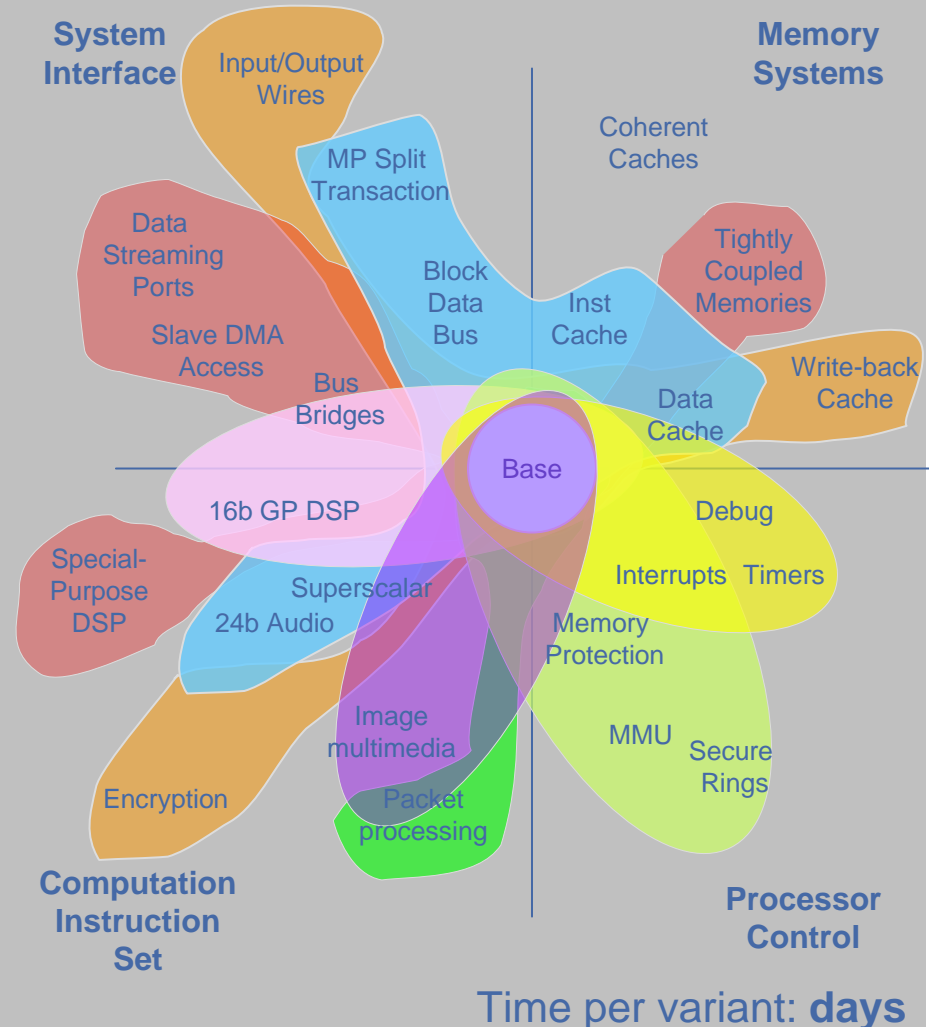


# Two models of processor evolution

## Traditional General-Purpose Processor Family



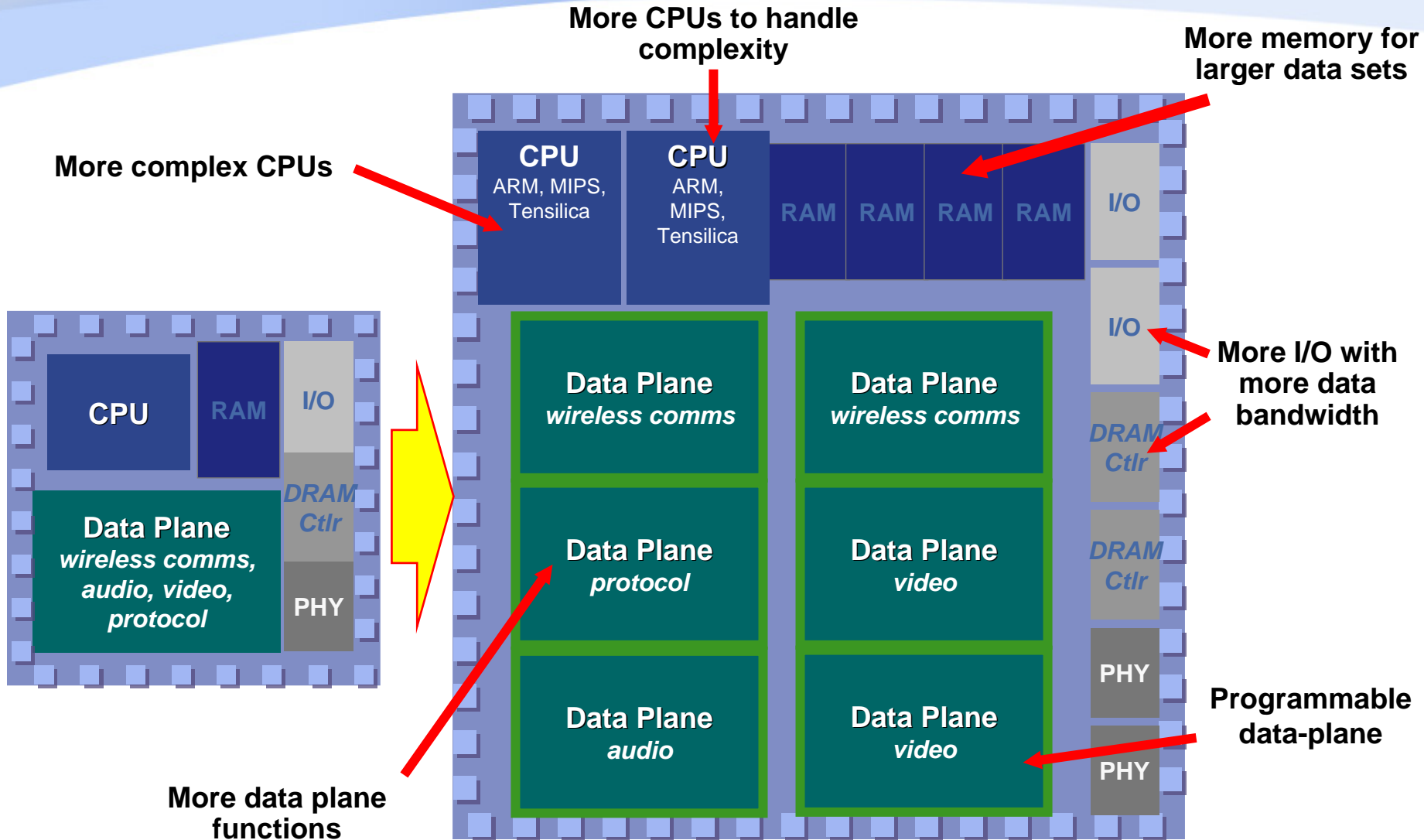
## Configurable Processor Family



Area = silicon cost and power

- **Optimize data-plane processors to provide hardwired-logic performance**
  - **Optimize registers and function units for a particular data type:**
    - 24-bit audio
    - 3x8-bit video
    - 56-bit encryption keys
    - 20/40-byte packet header data structures
    - In addition to full 32-bit RISC instruction set
  - **Specialized instructions reduce clock rate and lower power:**
    - One optimized instruction = 5-50 RISC instructions
  - **Better than standard RISC + hardwired accelerator blocks**
    - No bus delays for interblock communications
    - Equivalent performance and area
    - Easier verification (automated hardware generation, correct by construction)
    - Instantly upgradeable with software

# Silicon Scaling Drives Multi-Core SOCs

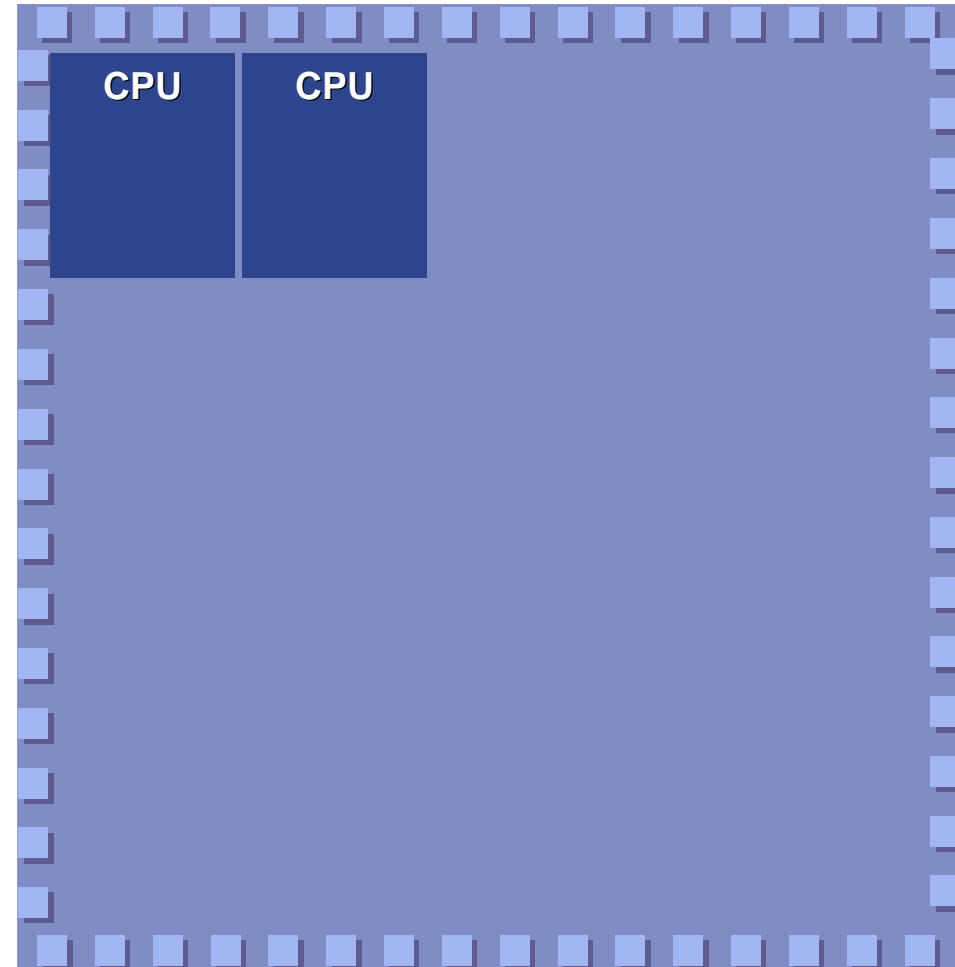


# The CPU Challenge for SOCs

## *Silicon Scaling No Longer Enough*

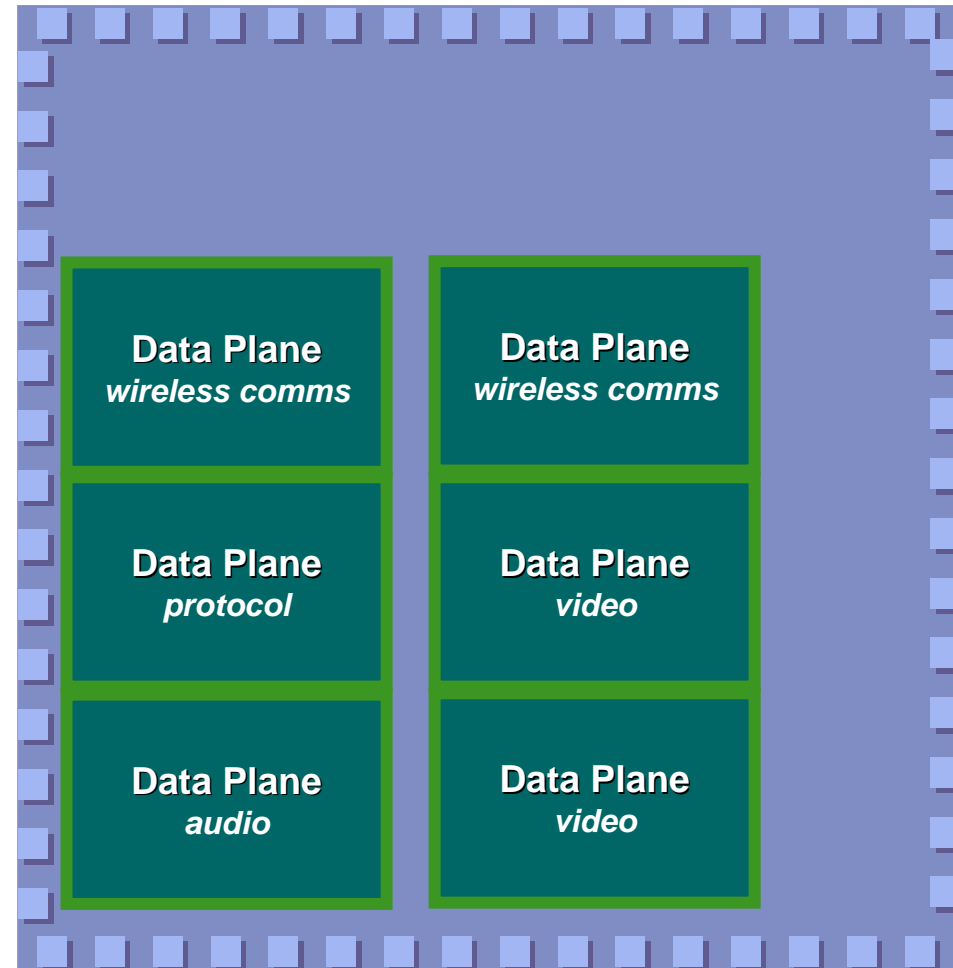
### Multiple CPU limited by software

- **CPU performance demand scales with complexity**
- **Multiple small CPUs more energy efficient than single large CPU**
- **Symmetrical CPUs suitable for control tasks**
- **CPU technology moving down from computer systems to embedded systems (advanced OS, cache-coherency, SMP clusters)**
- **Multi-processor CPUs can mean programmer retraining and legacy software re-engineering**



Multiple data-plane functions limited only by feature demands

- **End markets demand richer, more complicated mix of communications and multimedia functions**
- **Evolving functions need more efficient, specialized processors for each function**
- **Diversity of data-processing needs drives diversity of data-plane processors**



# How Does Processor Extension Work?

## Example: Tensilica Instruction Extension (TIE)

```
operation SIMDadd {in AR a, in AR b, out AR c} {} {assign c = {a[31:16]+b[31:16], a[15:0]+b[15:0]};}
```

Register files include automatic addition of load/store ops, data-types, compiler register allocation. Arbitrary width, depth and (inferred) ports

Any number of state registers, arbitrary width and ports

Any number of additional interfaces: queues, wires, special memories

New operations: any number of source and destination operands [5 source, 3 dest]. Automatic port generation and encoding

Explicit operands visible to register allocation, assembly

Implicit operands take no encoding space, include memory ports

Body of operation uses Verilog syntax plus rich TIE function libraries

Optional scheduling. Automates pipelining of hardware, code scheduler, simulators

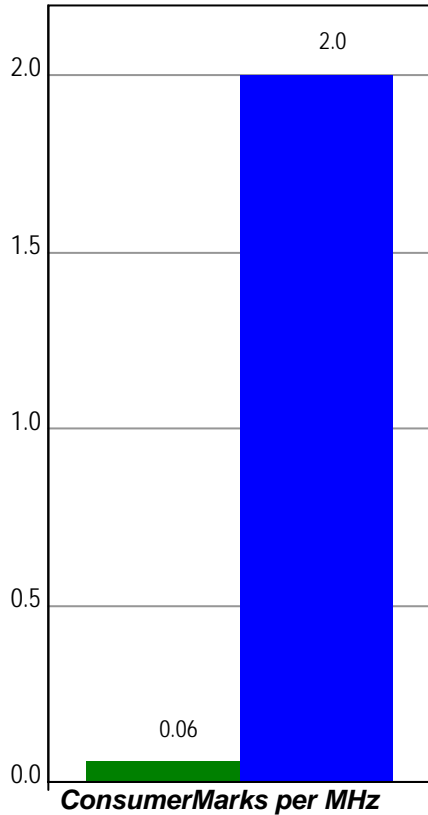
```
regfile XR 64 8 x
state Accum 80
queue realq 40 out
function [31:0] sat ([39:0] x) {assign sat = |x[39:32]?32'hfffffff:x[31:0];}
immediate_range im4 0 120 8
operation ComplexMAC {in AR a, in im4 i, in AR b, in AR c, out XR d}
{inout Accum, out VAddr, in MemDataIn64, out realq} {
assign VAddr = a + i;
wire [39:0] real = b[15:0]*c[15:0]-b[31:16]*c[31:16]+ Accum[39:0] + MemDataIn[31:0];
wire [39:0] imag = b[15:0]*c[31:16]+b[31:16]*c[15:0]+ Accum[79:40] + MemDataIn[63:32];
assign Accum = {imag,real};
assign realq = real;
assign d = {sat32(imag),sat32(real)};
}
schedule scCM {ComplexMAC} {def Accum 2; def realq 2; def d 3;}
```



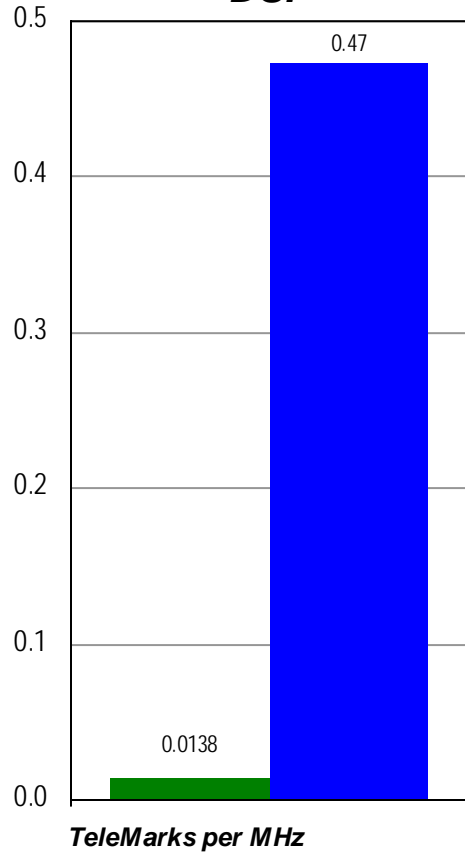
# Benefit of Processor Generation

*Big performance gains from domain specialization*

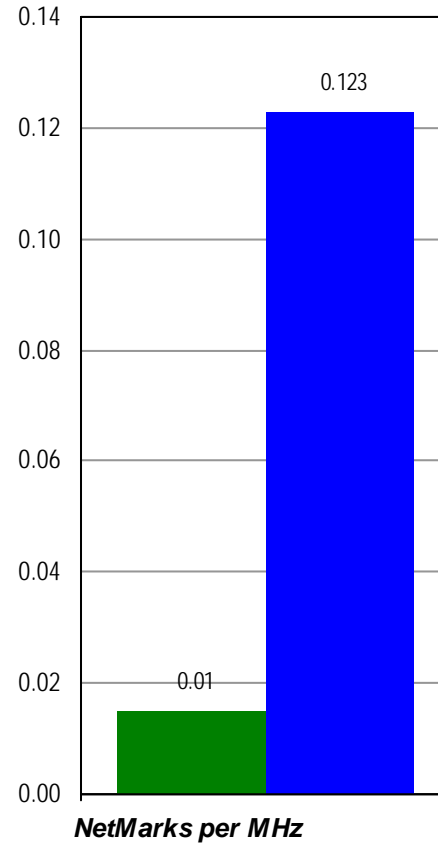
### Consumer



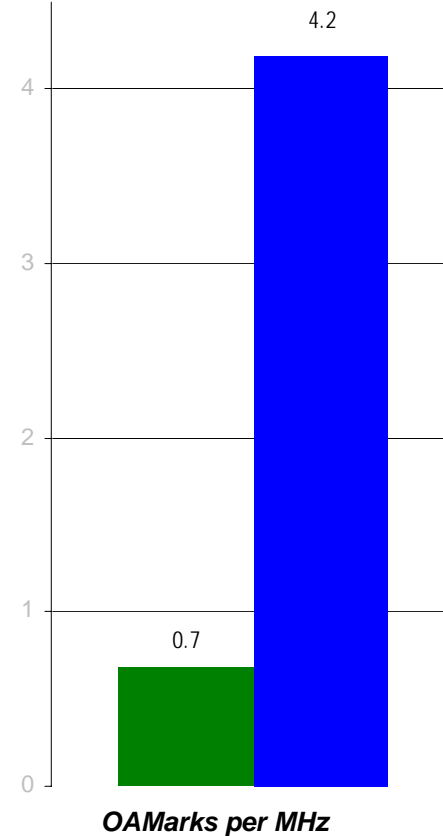
### DSP



### Networking



### Office Automation



■ Industry average RISC processor cores

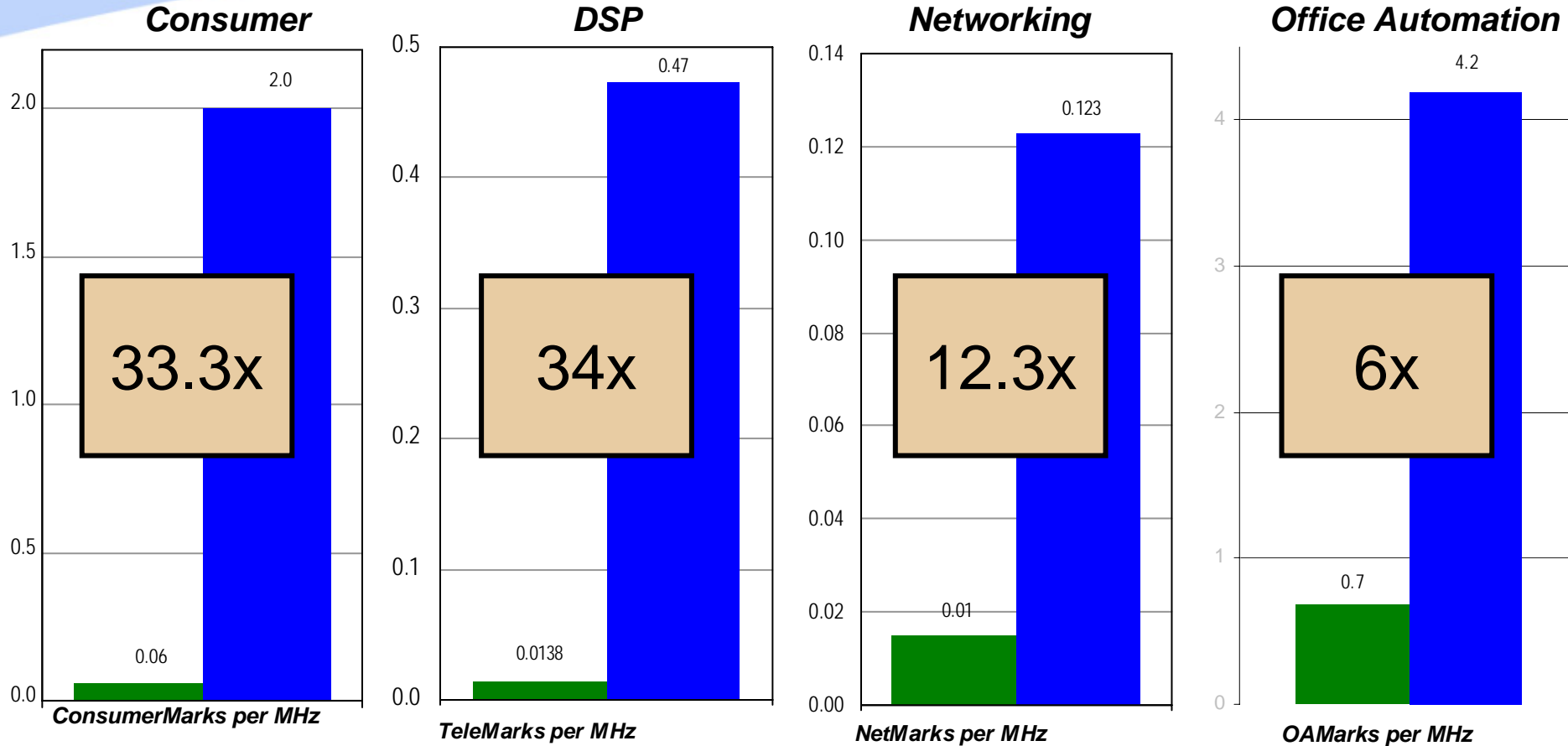
■ Optimized processor cores

Source: EEMBC Certified Benchmarks



# Benefit of Processor Generation

*Big performance gains from domain specialization*



■ Industry average RISC processor cores

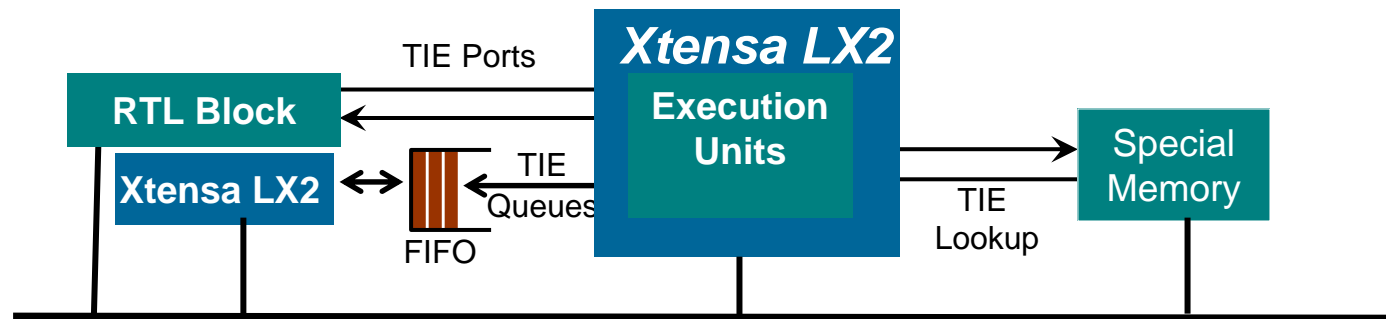
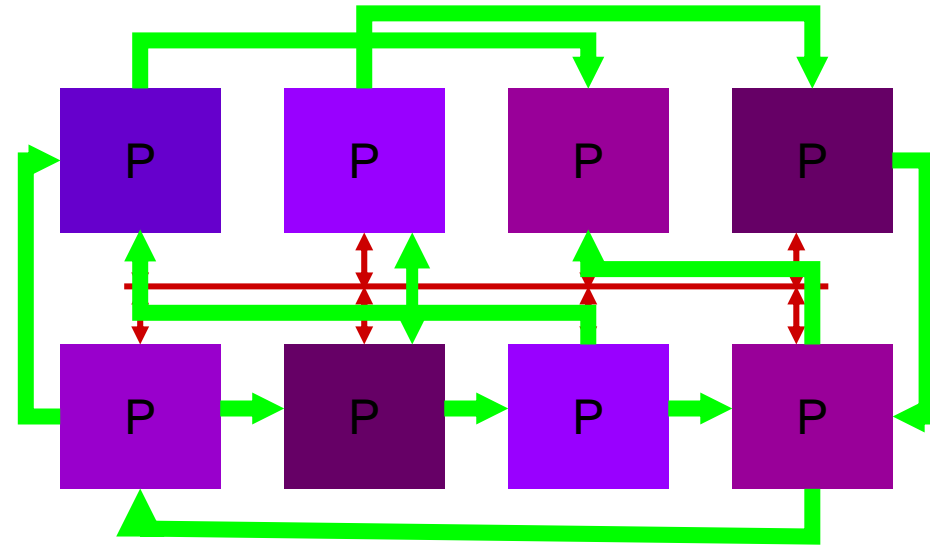
■ Optimized processor cores

Source: EEMBC Certified Benchmarks

# Beyond the Bus

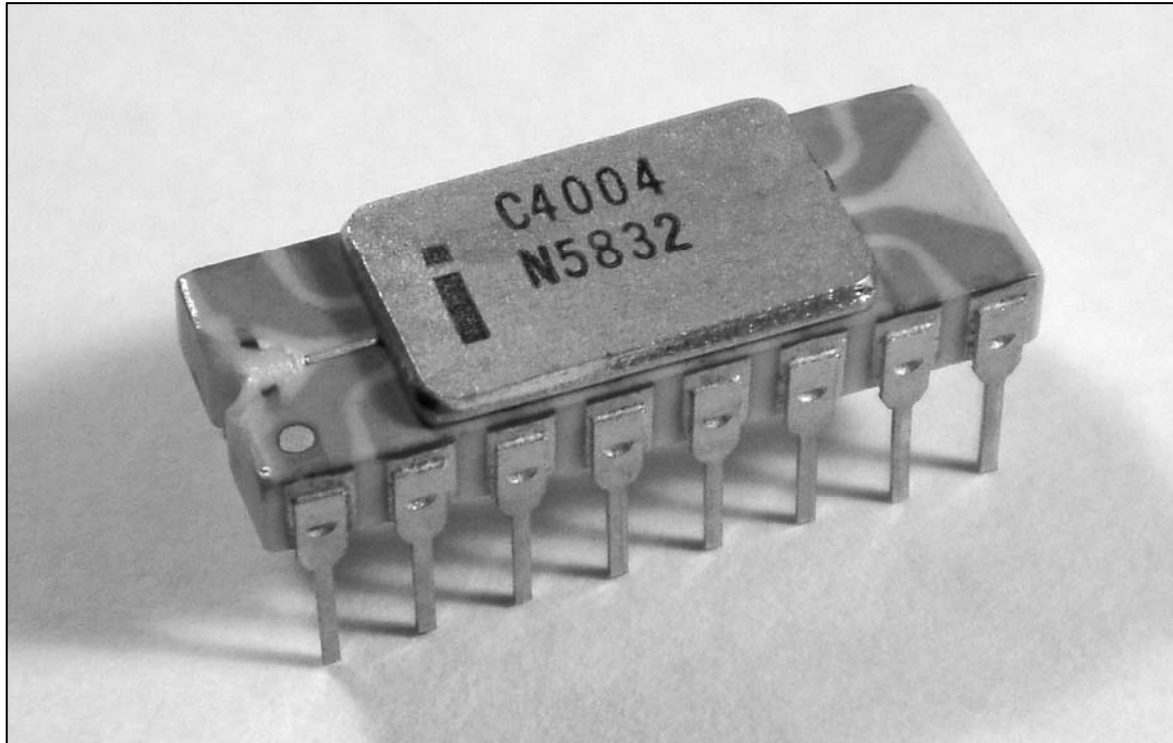
## Direct Port, Queue, & Lookup Interfaces

- More processors require more bandwidth
- Shared bus interfaces create significant bottlenecks
- Replace bus interface for connections that need high bandwidth, low latency
- Extensible I/O wires, direct queue interfaces, RAM tables connect directly to processor's data path
- Queue interfaces directly implement synchronization (message send-receive)
- Application-specific processors complements application-specific interconnect



# World's First Commercial Microprocessor

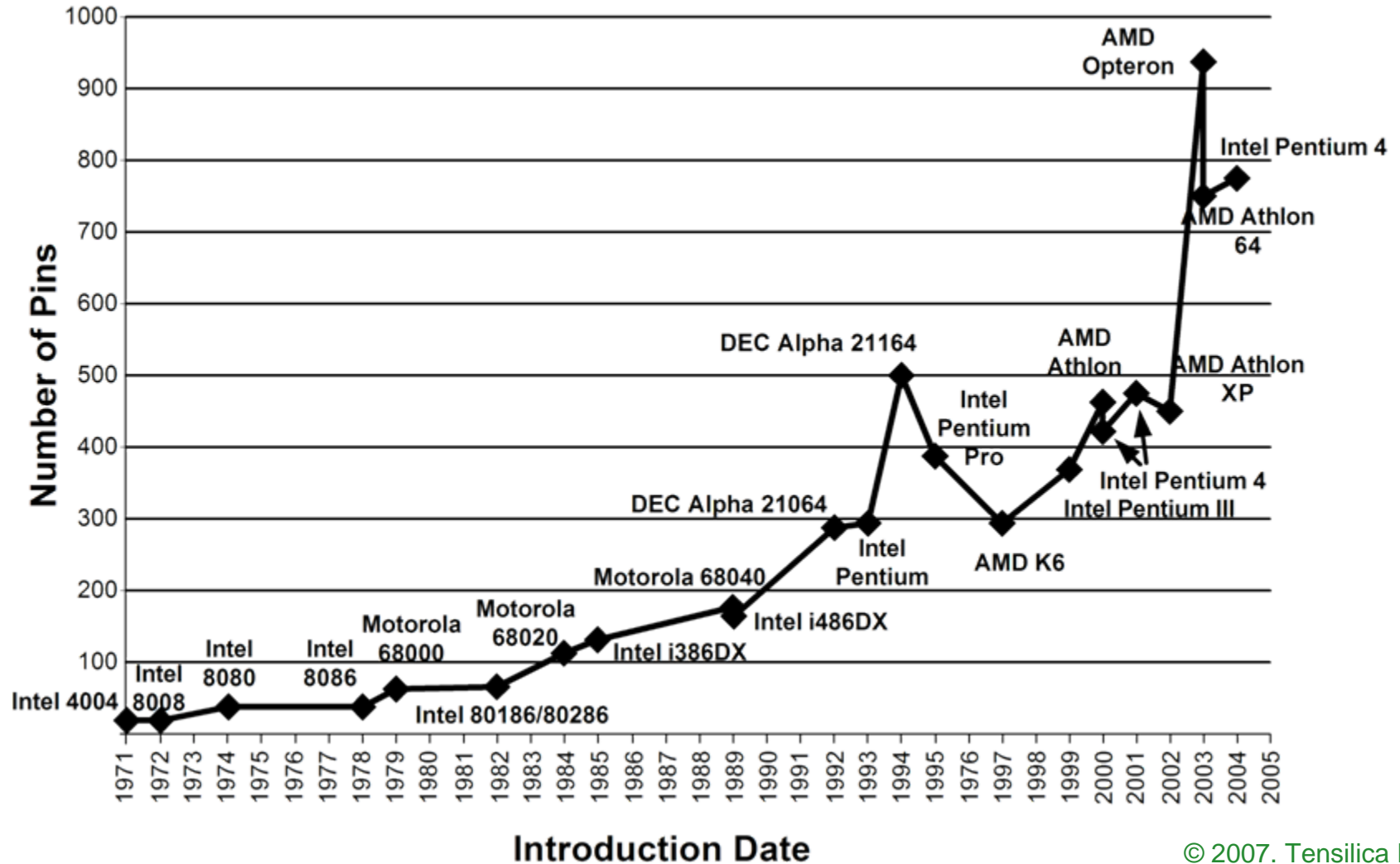
*The Intel 4004, Circa 1971*



**16 pins – necessitating a multiplexed bus**  
**And processors have been pin-limited ever since**

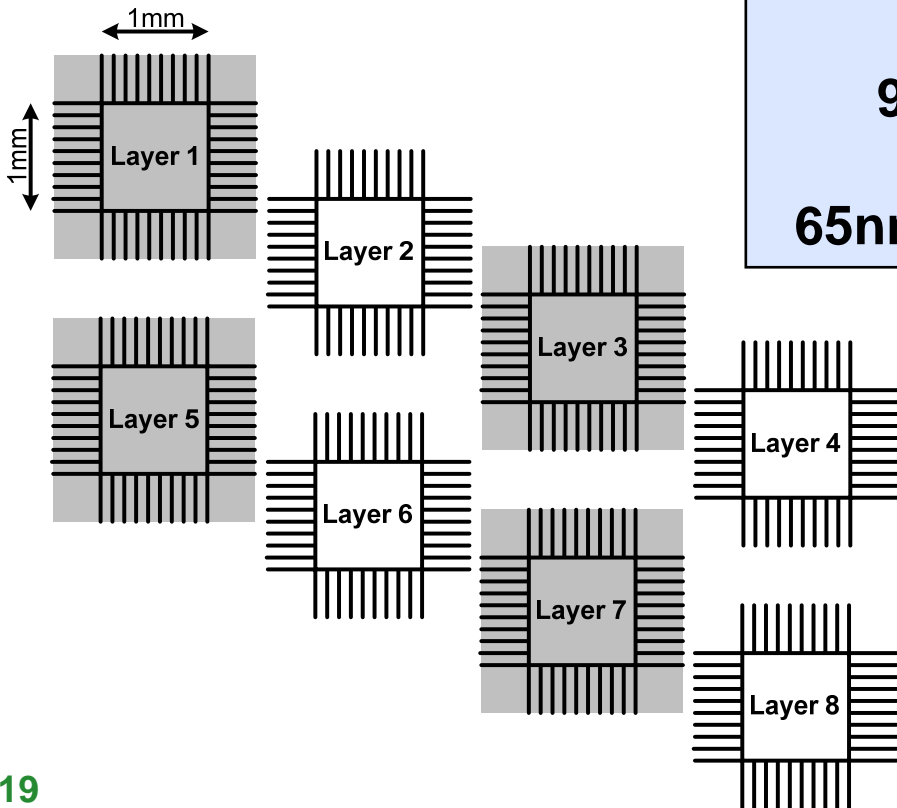
# Rise in Packaged Microprocessor Pin Count Over Time

Microprocessor Pin Count over Time



# Wide Interconnect and Wire Density: *What's Practical?*

- **Do the Math:**



**8-10 Metal Layers, ITRS 2006 wire spacing**

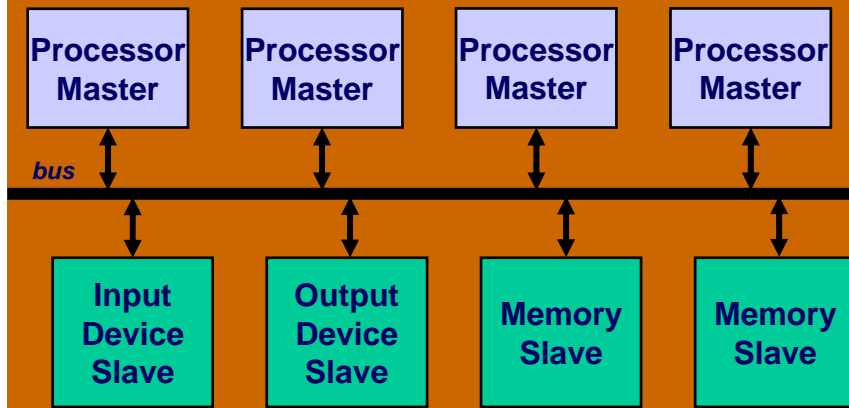
**90nm: 100,000+ wires/square mm**

**65nm: Almost 200,000 wires/square mm**

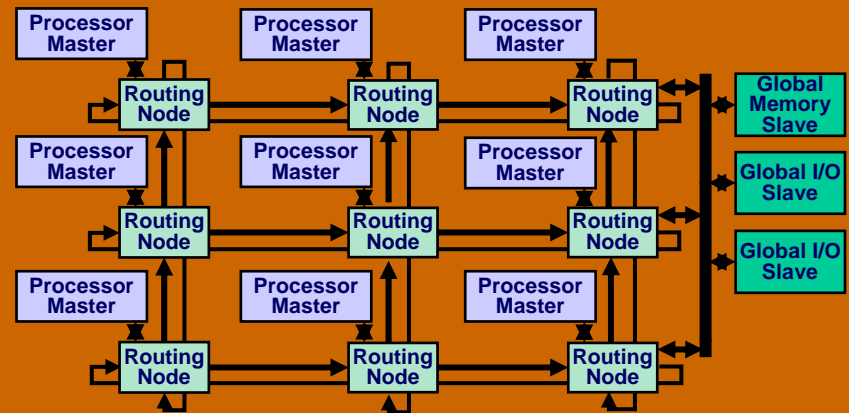
# Keys to Efficient MP

*Flexible range of system-interconnect topologies*

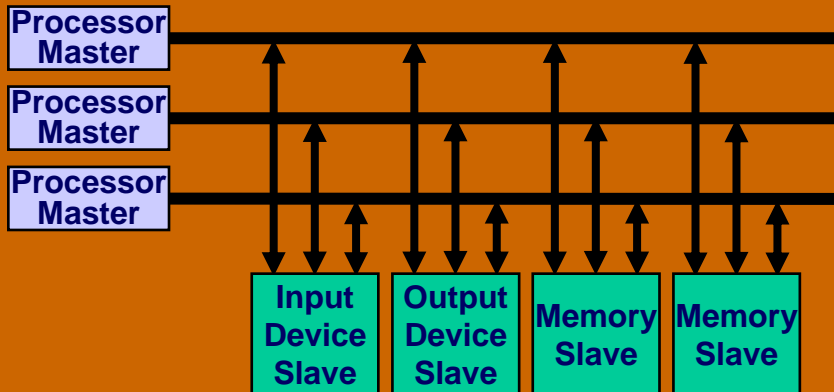
## Shared Bus



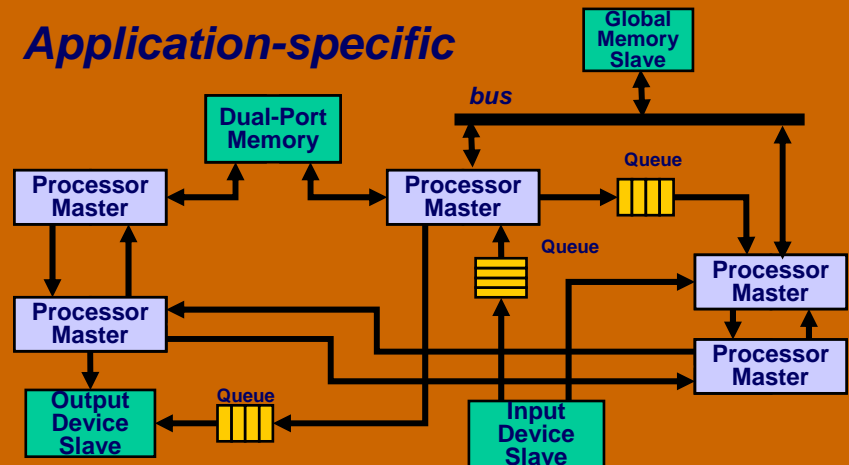
## On-chip Routing Network



## Cross-Bar

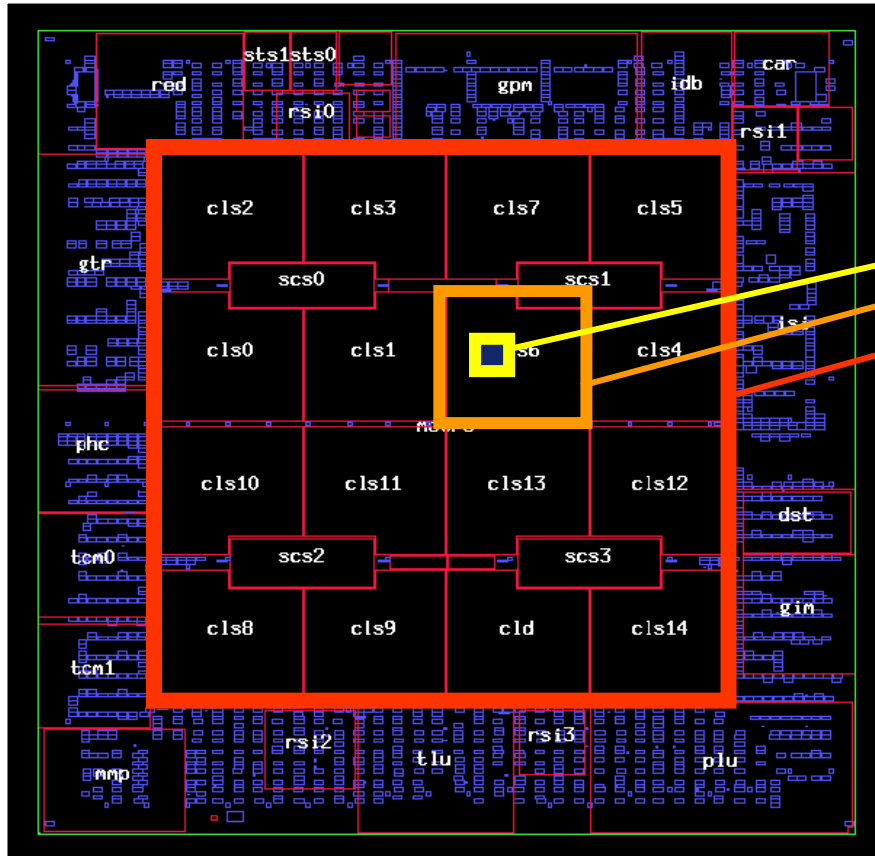


## Application-specific



# True “Many-Core” System-on-Chip

## 192 Xtensas Per Chip in Cisco CRS-1 Terabit Router



Complete 32-bit processor: 25K gates  
 12 processors per cluster  
 16 processor clusters per chip

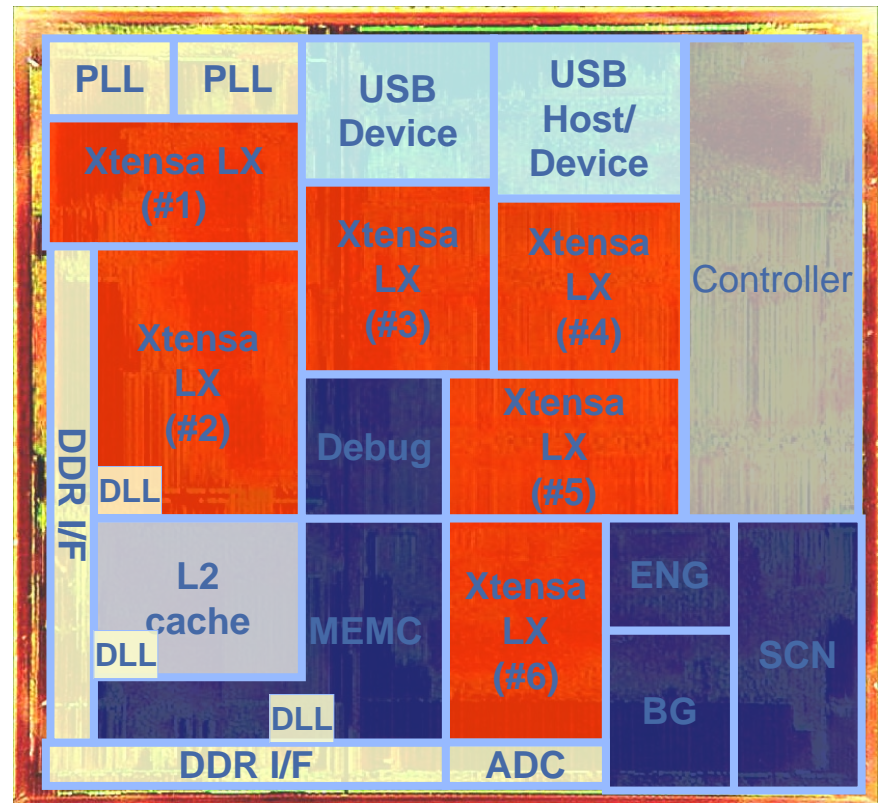


- 192 Xtensa network-processor cores per Silicon Packet Processor
- Up to 400,000 processors per system

# Complex Consumer Products

## EPSON's REALOID-based Printers

- EPSON's printer controller chip with six heterogeneous, asymmetric, configurable VLIW cores + legacy controller + I/O



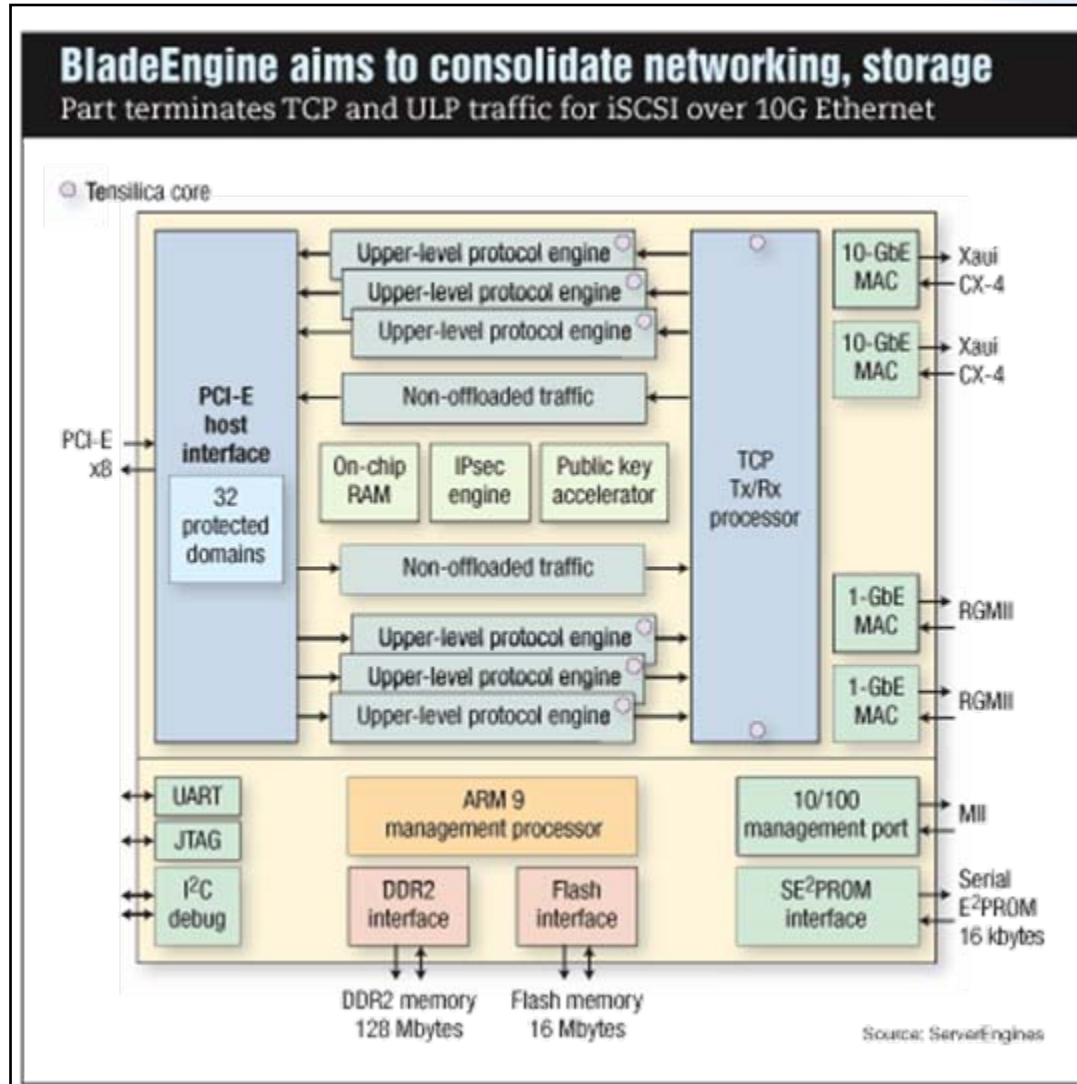
Epson REALOID IC Block Layout

# Data-plane is More than DSP

## *A Recent Example: Server Engines*

From EETimes: July 31, 2007

- Next generation high-volume server chipset
- Processor-based acceleration of TCP/IP and network/storage protocols
- 8 configurable data processors + ARM management processor

# Closing Thoughts

- 1. Processor paradigms from board-level days no longer valid! Ever-climbing clock rate increases and pin-count limitations are archaic limitations no longer in effect.**
- 2. Silicon scaling makes multi-processor SOC design inescapable! There's no other clear way to get more performance.**
- 3. On-chip and off-chip inter-processor communications are just as important as computation**
- 4. As processor frequency improvement slows, system-architecture optimization becomes the only open path to more performance**
- 5. It's a heterogeneous world – with increasingly diverse, application-driven processor and interconnect needs**