The Engine of SOC Design

The Reinvention of the Microprocessor for MPSOC

Chris Rowen
President and CEO
Tensilica, Inc.

Outline

- Basic changes in the electronics environment
- Moore Law gives us less
- The Old World: A short history of the microprocessor
- The New World:
  - Enter the configurable processor
  - Concurrency and multiple processors
  - Design flow for multiple processor embedded systems
- Wrap-up: The future of system on chip

Basic Changes in Industry Drive Innovation

Change #1: Products go mobile, connected, always-on, and media-rich

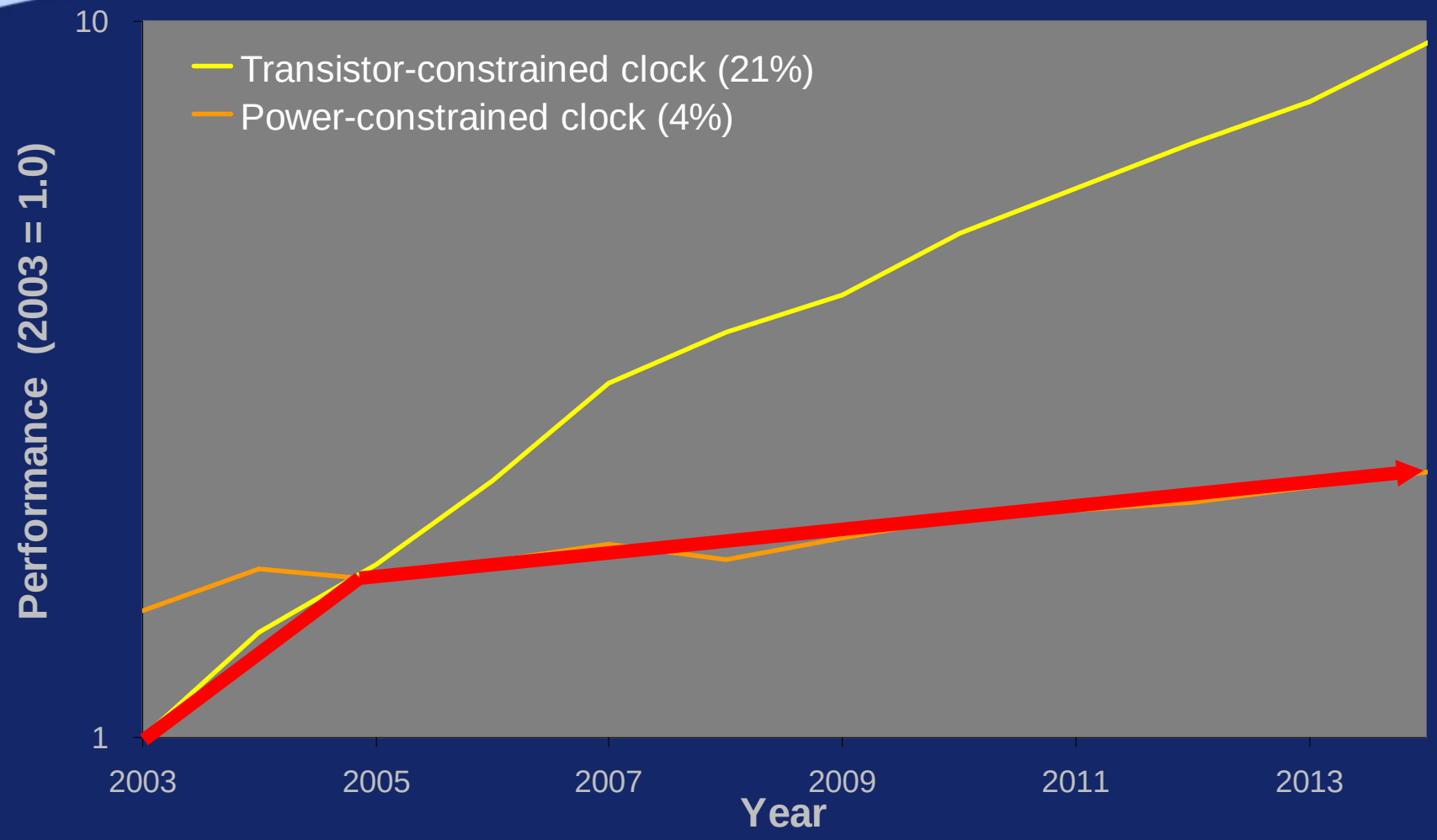
Change #2: Product complexity drives up development cost and risk

Change #3: Moore's Law drives lower silicon cost but not lower power

The Old World

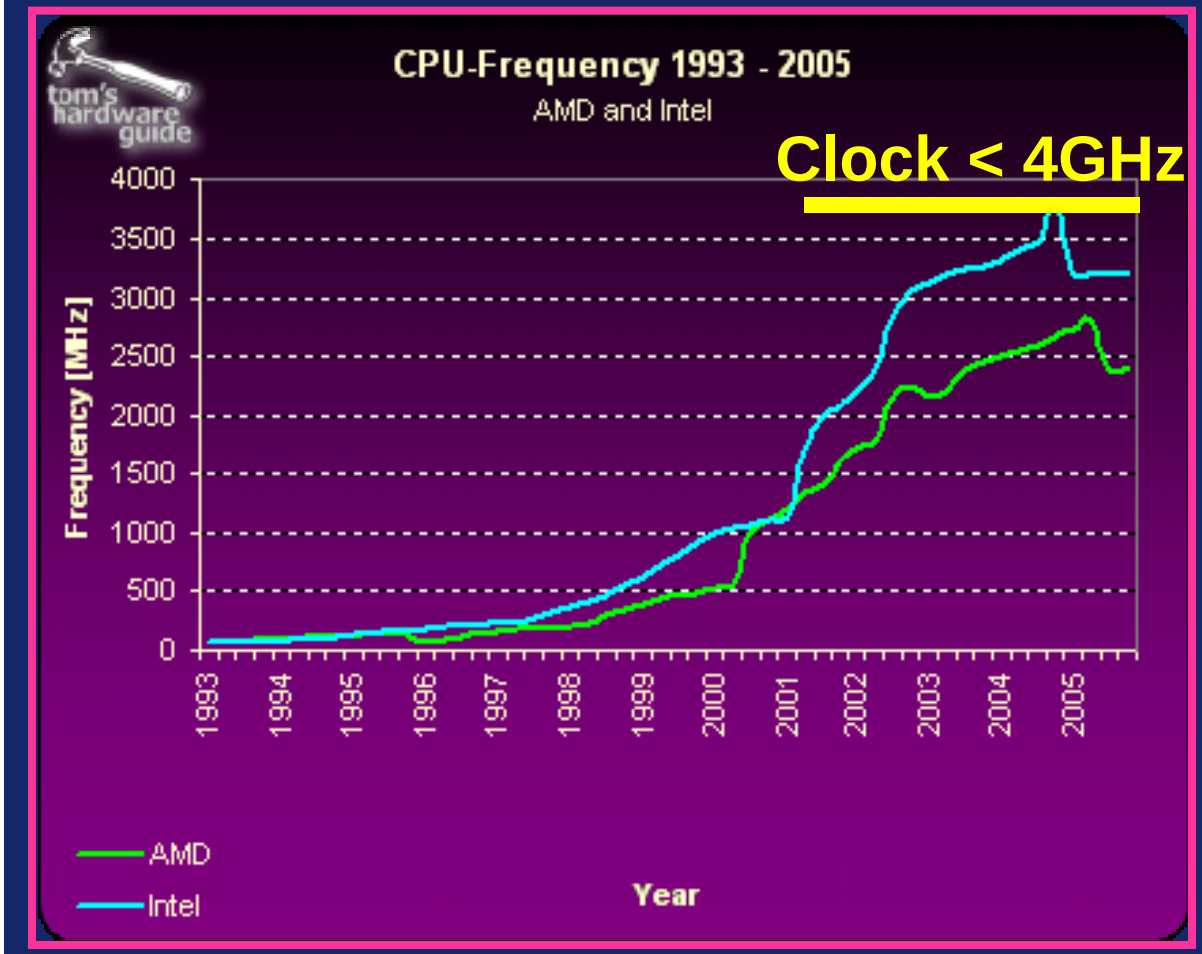
1. **Board-level using standard ICs, FPGA and ASIC**
   - Modest total hardware and software complexity
   - Pin-count-limited interfaces between sub-systems
   - Processor requires full IC
2. **Manual development of processor hardware and software tools forces one-size-fits-all processors**
   - Limitation of processor performance and energy efficiency
   - Long development cycles for new programmable platforms
3. **Deep sub-micron silicon design limited to advanced chip-makers**
   - High schedule and risk in hardwired functions

Inflexion Point in Clock Frequency



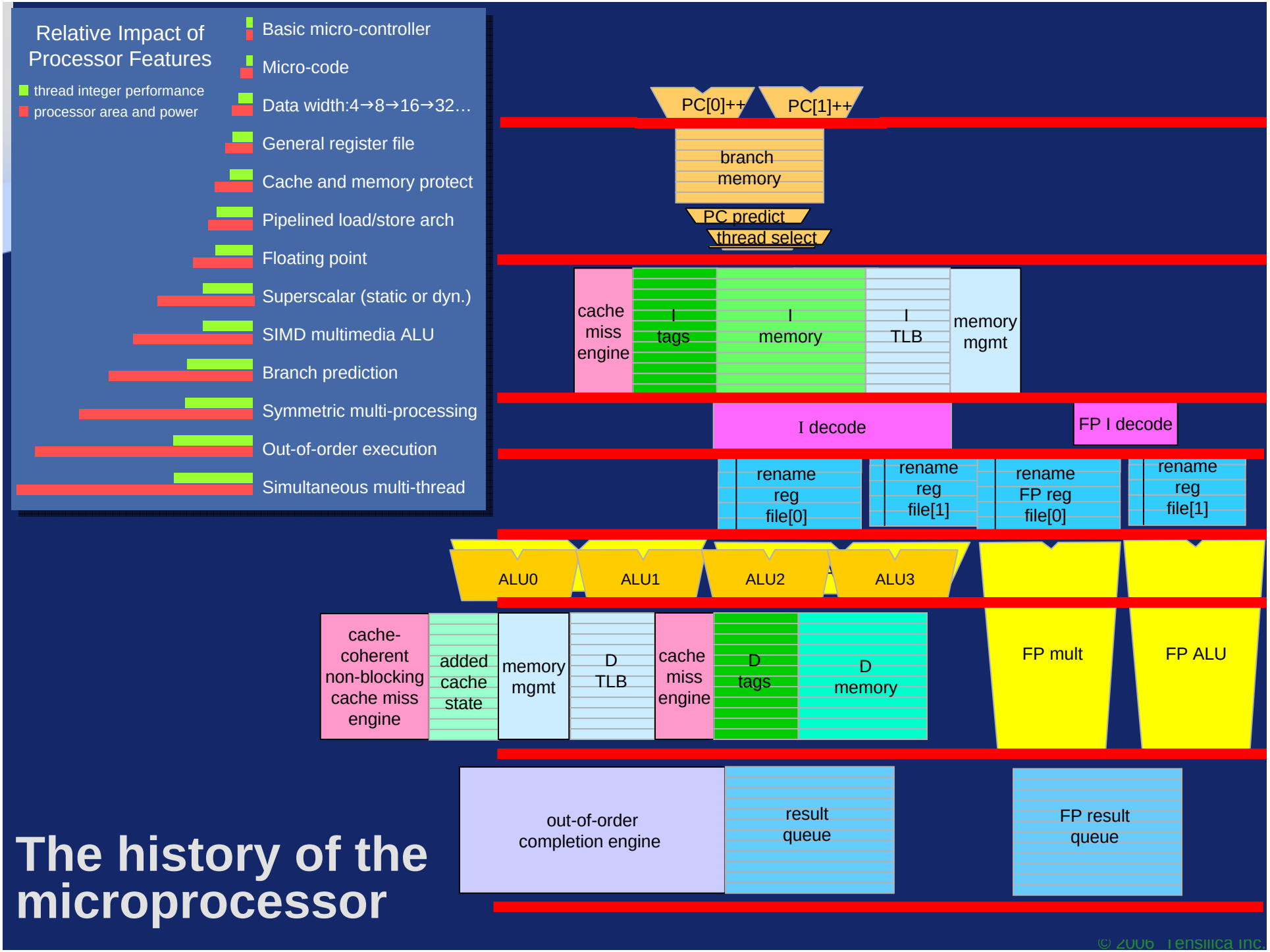
Based on International Technology Roadmap for Semiconductors 12/03

High-End Processors hit MHz ceiling



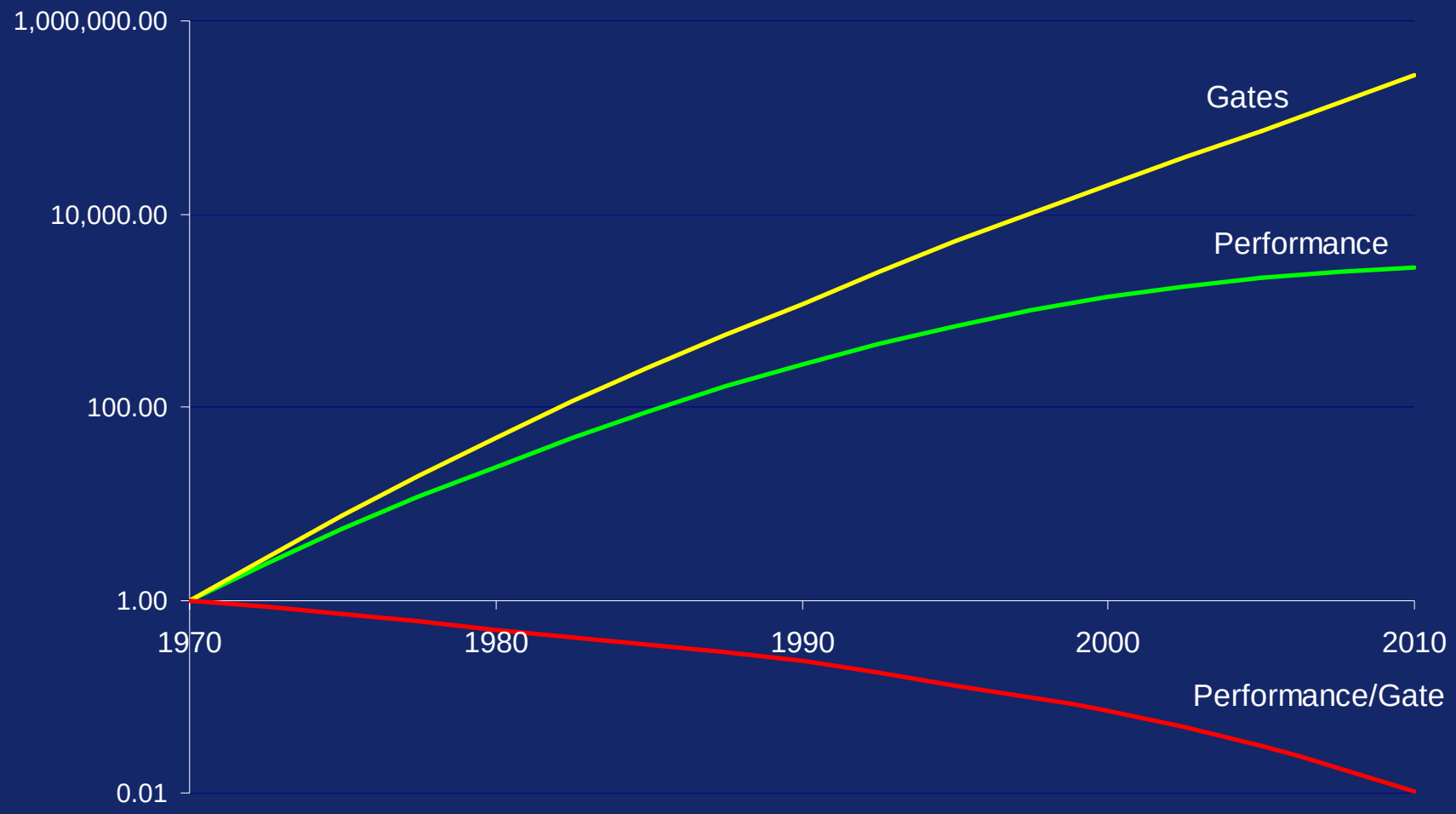
Basic Implications:

1. Get performance from better architecture instead of more MHz
2. Use multiple processors


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
PC[0]++
PC[1]++
branch memory
PC predict
thread select
cache miss engine
I tags
I memory
I TLB
memory mgmt
I decode
FP I decode
rename reg file[0]
rename reg file[1]
rename FP reg file[0]
rename reg file[1]
ALU0
ALU1
ALU2
ALU3
cache-coherent non-blocking cache miss engine
added cache state
memory mgmt
D TLB
cache miss engine
D tags
D memory
FP mult
FP ALU
out-of-order completion engine
result queue
FP result queue
The history of the microprocessor
© 2006 Tensilica Inc.

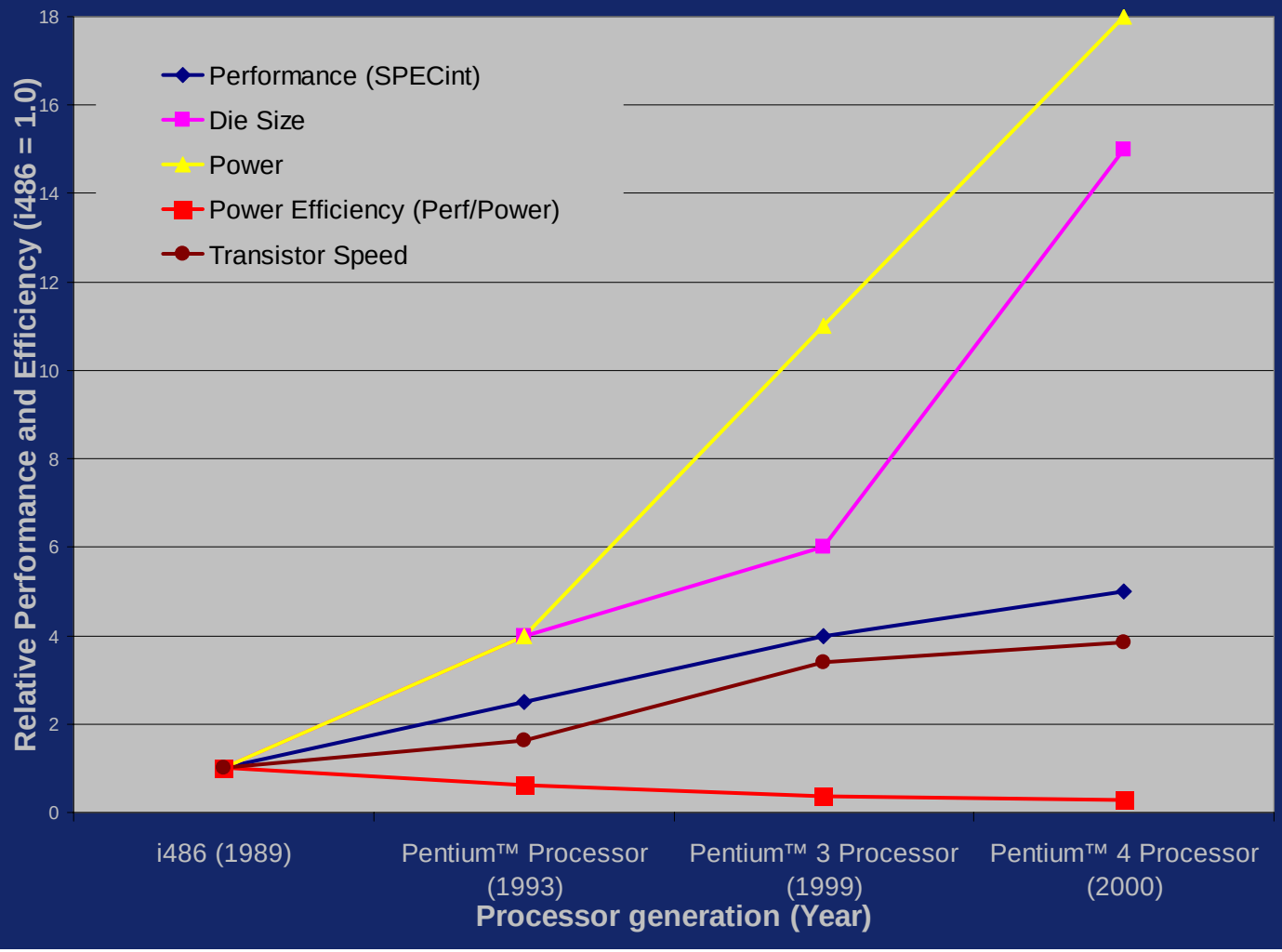
Long-term uni-processor scaling trend

…What Happens When you Don't Have Enough MIPS?

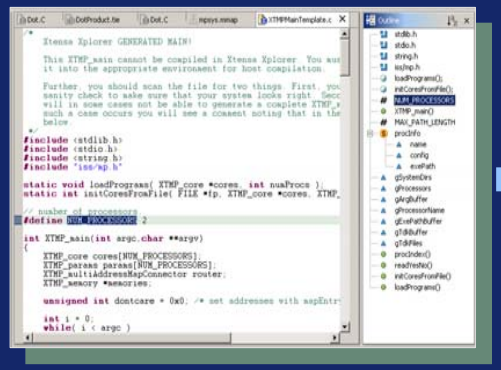


Intel's Own Assessment

Power and Area increase more rapidly than Performance



The New World
Automatic Processor Generation

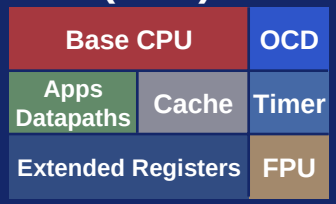


Processor configuration

1. Select from menu
2. Automatic instruction discovery (XPRES Compiler)
3. Explicit instruction description (TIE)

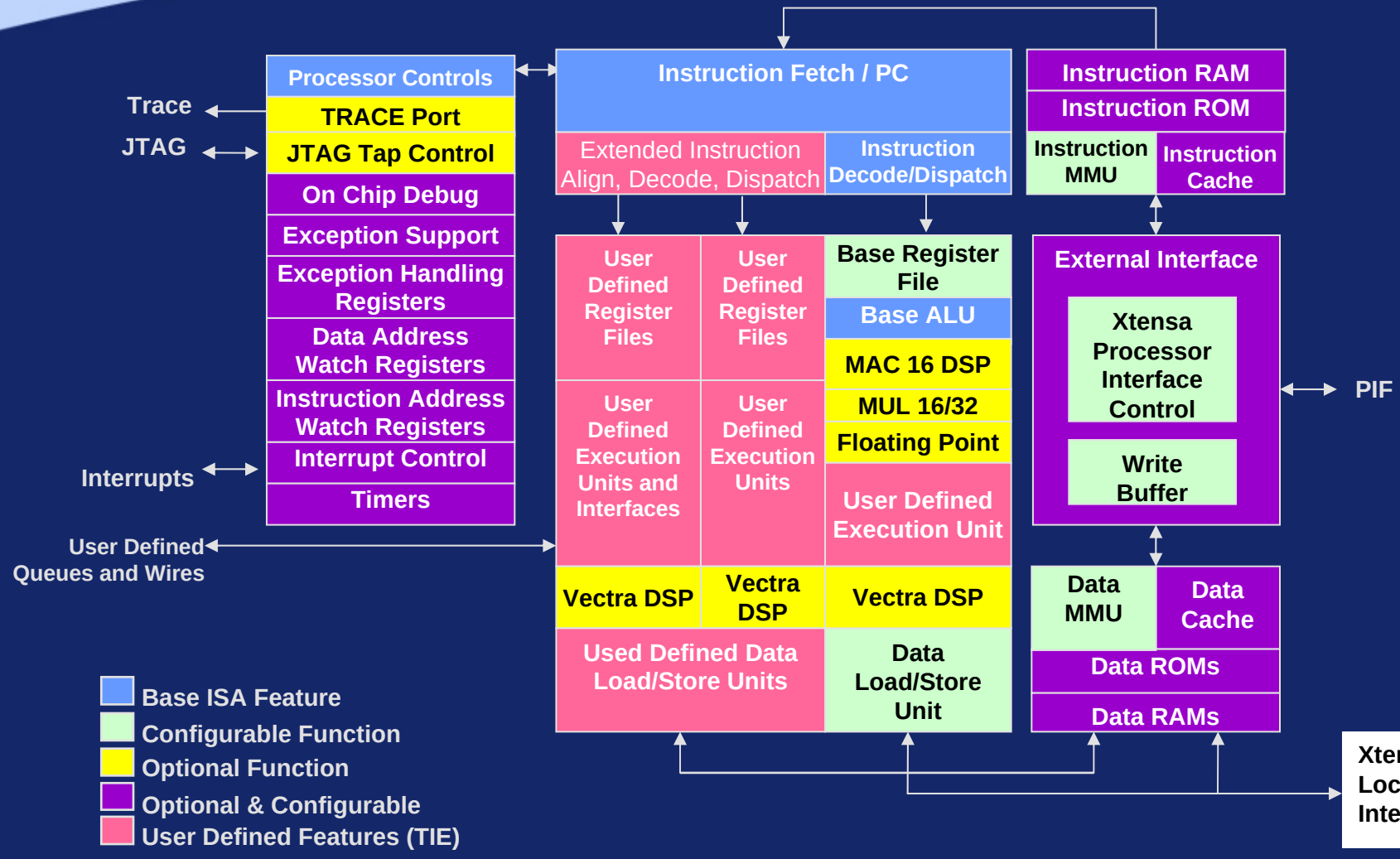
Tensilica Processor Generator

Application-optimized processor implementation (RTL)

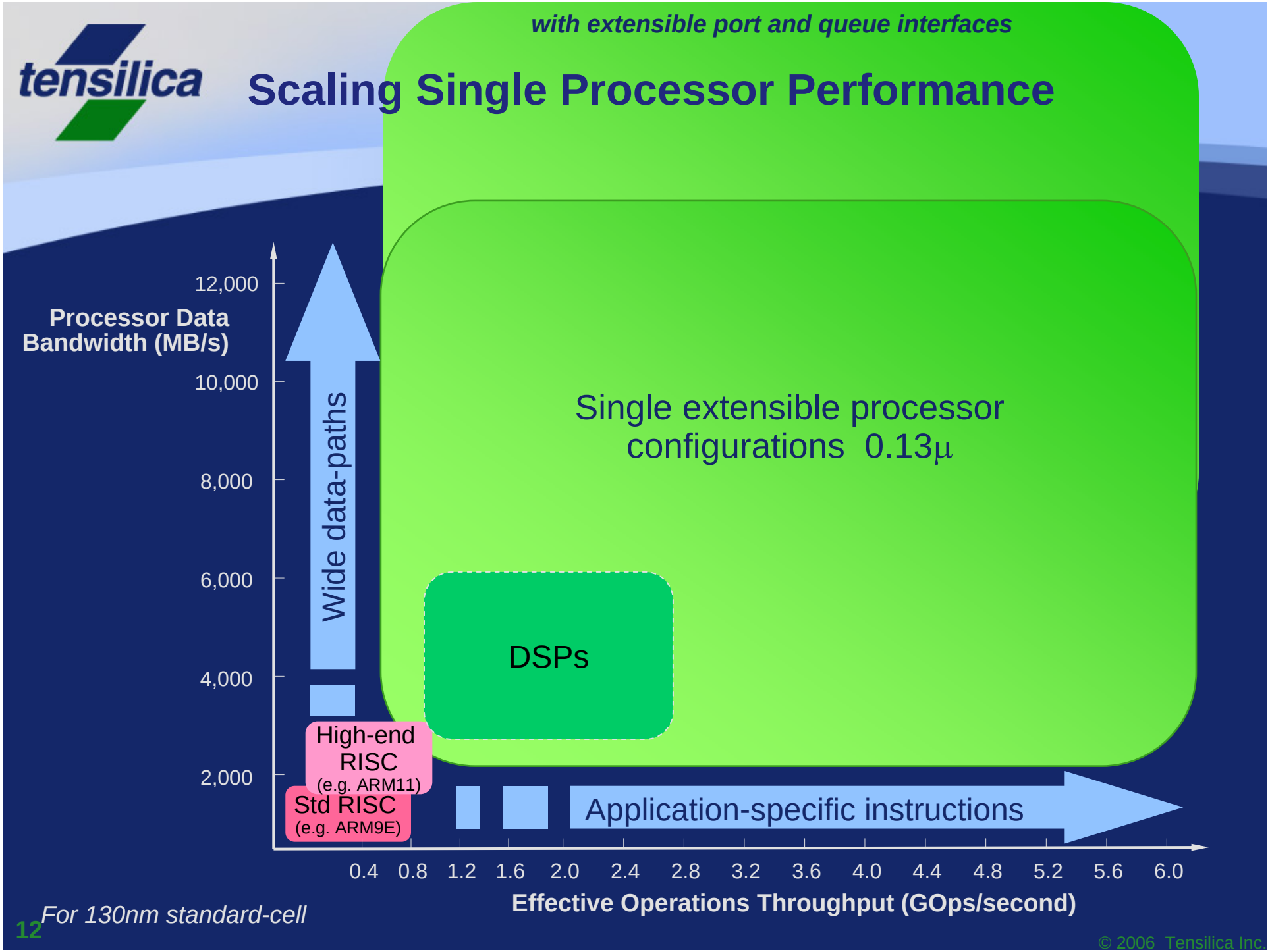


Tailored SW Tools: Compiler, debugger, simulators, OS ports

Build Almost Any Processor



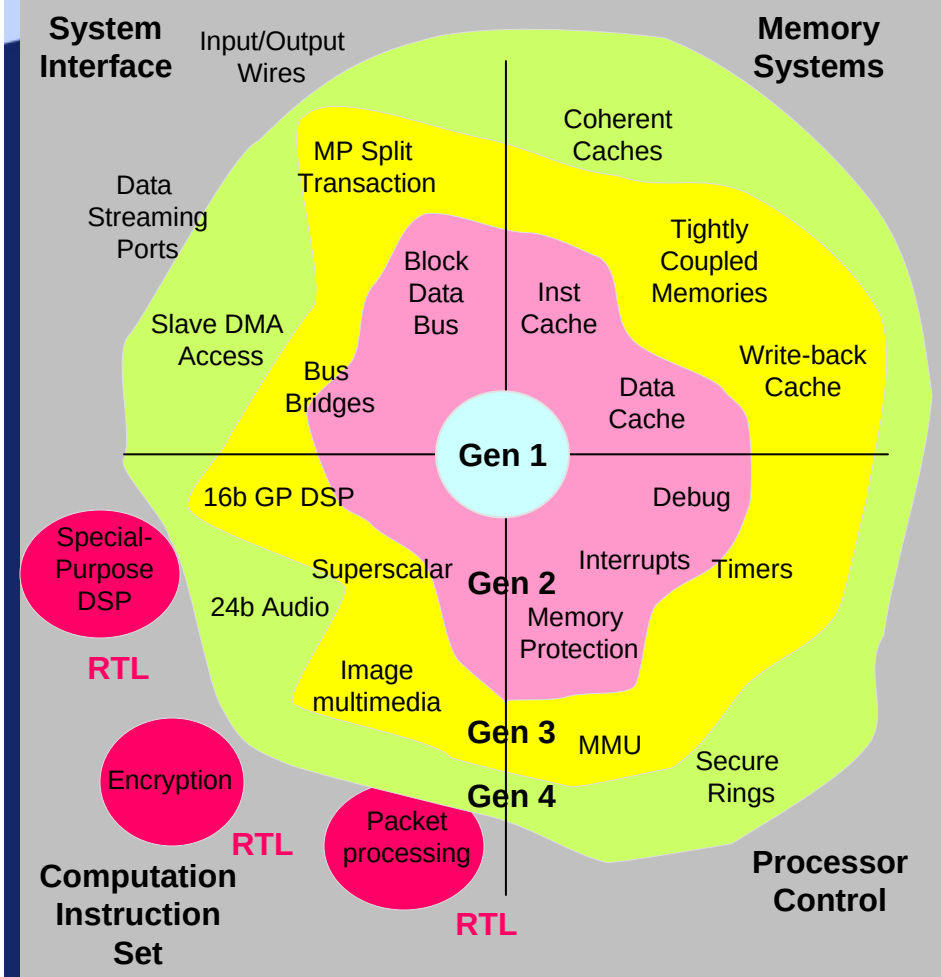
Scaling Single Processor Performance



For 130nm standard-cell

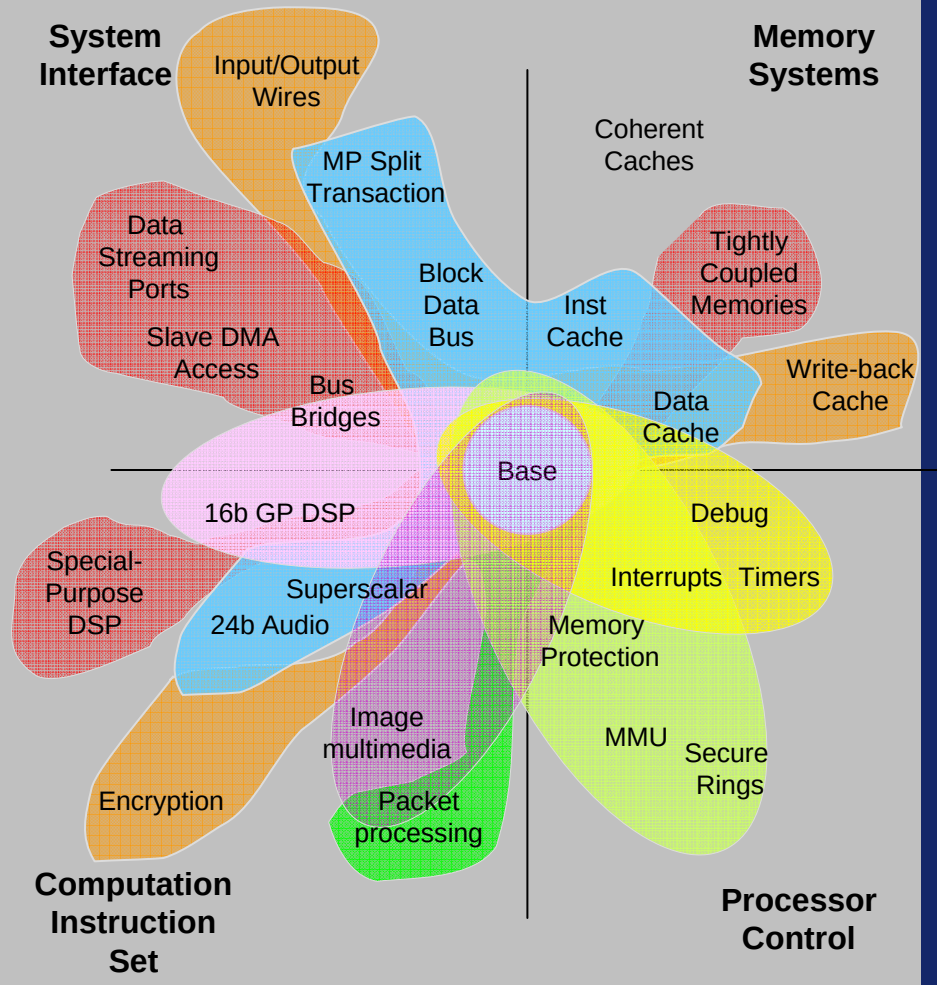
Two models of processor evolution

Traditional Processor Family



Time per variant: **years**

Configurable Processor Family



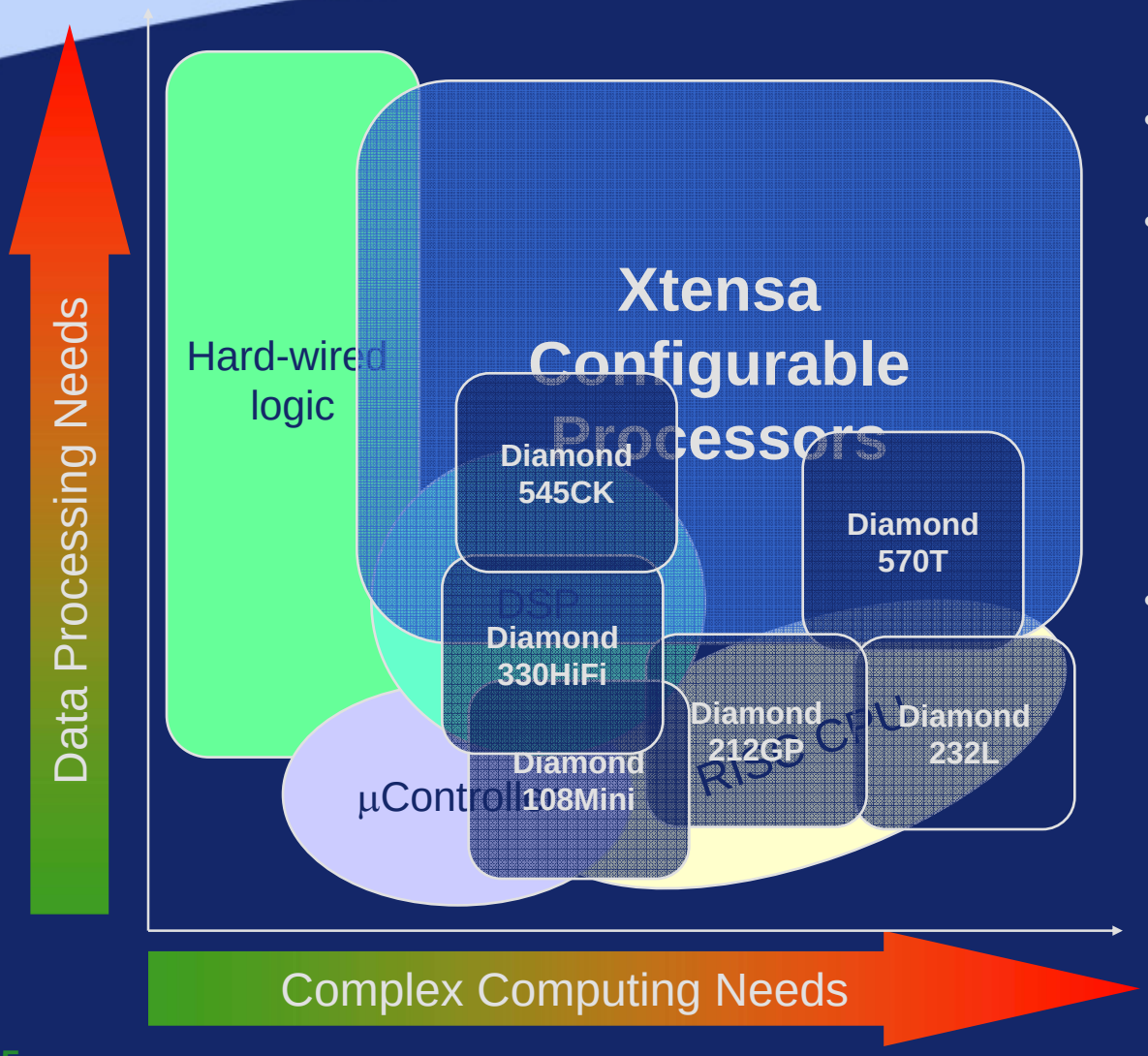
Time per variant: **days**

Area = silicon cost and power

Why Configurability Now?

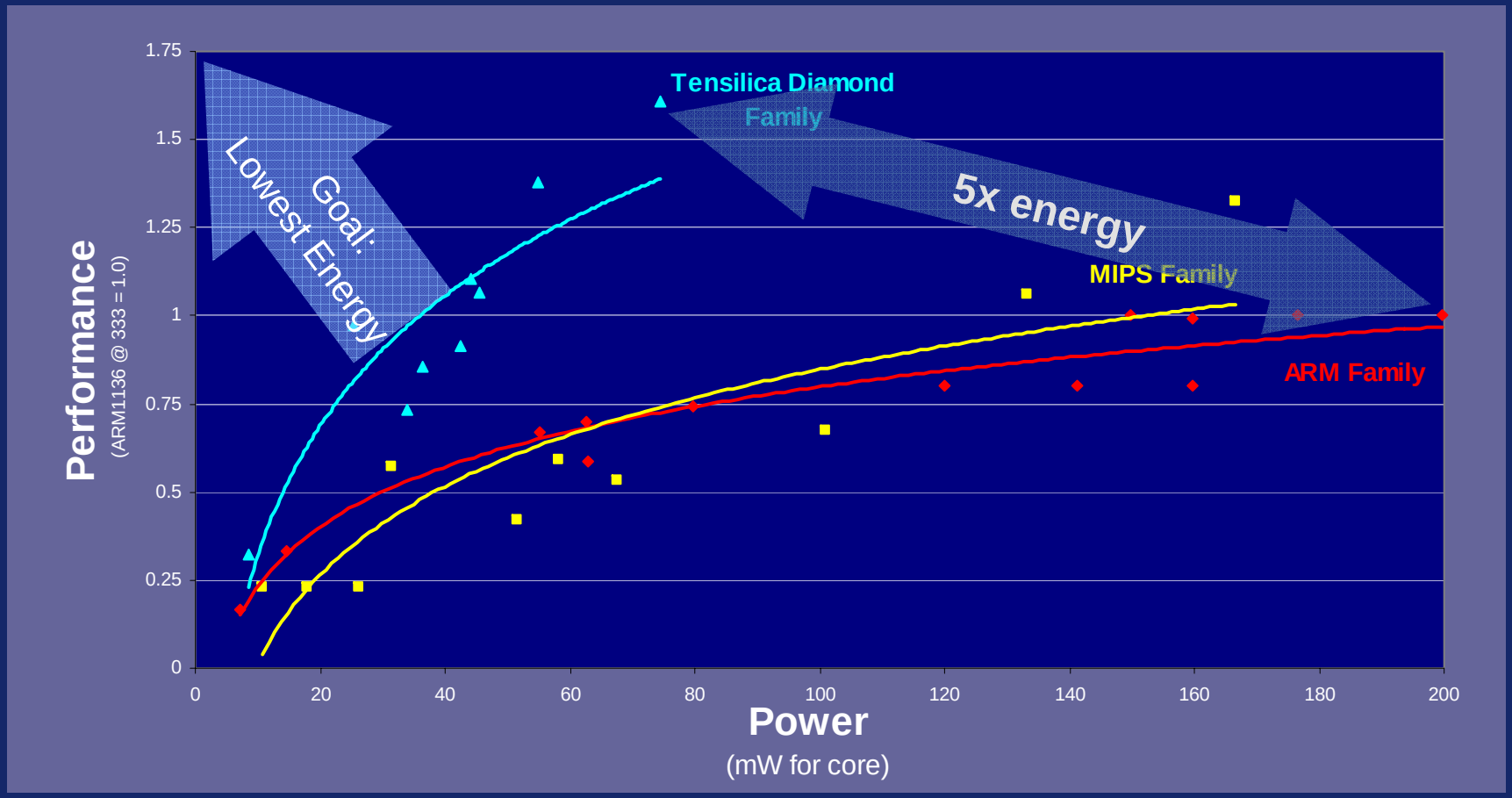
1. **Breakthrough on automatic generation of hardware + software**
   - Same quality and completeness of compilers, debuggers, development GUI, simulators, prototype emulation, OS ports as traditional one-size-fits all processor families
2. **Application-directed processor naturally fits into application-directed system-on-chip**
   - No extra mask cost in using unique processor variant
   - Smaller and lower power than generic processor
   - More flexible and programmable than hardwired accelerator blocks

Covering Breadth of SOC Processor Demands



- Tensilica is the largest supplier of configurable processors
- Diamond Standard cores:
  - Broadest line of controller, CPU and DSP cores in industry
  - Highest performance synthesizable general-purpose CPU
  - Highest performance DSP
  - Most complete low-power audio solution
- Tensilica spans general-purpose and application-specific processors and software

Processor Power and Performance
Standard Core Families



Performance on EEMBC benchmarks aggregate for Consumer, Telecom, Office, Network, based on ARM1136J-S (Freescale i.MX31), ARM1026EJ-S, Tensilica Diamond 570T, T1050 and T1030, MIPS 20K, NECVR5000). MIPS M4K, MIPS 4Ke, MIPS 4Ks, MIPS 24K, ARM 968E-S, ARM 966E-S, ARM926EJ-S, ARM7TDMI-S scaled by ratio of Dhrystone MIPS within architecture family. All power figures from vendor websites, 2/23/2006

Automatic Instruction Set Optimization

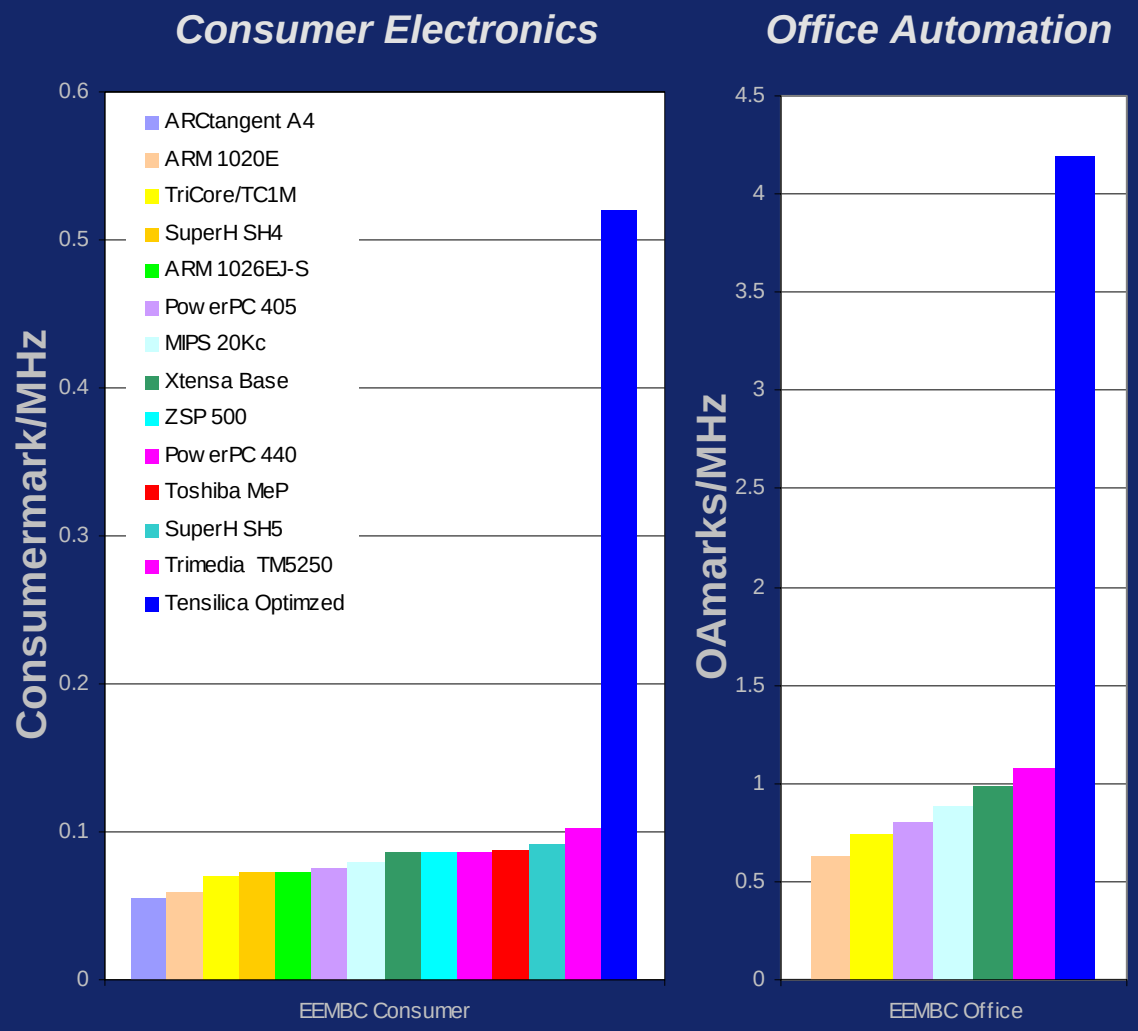
General-purpose processors all look alike (more or less)

Automatic optimization from general-purpose C code with *XPRES Compiler*

- no intrinsic functions
- no assembly code
- no instruction set hardware definition coding

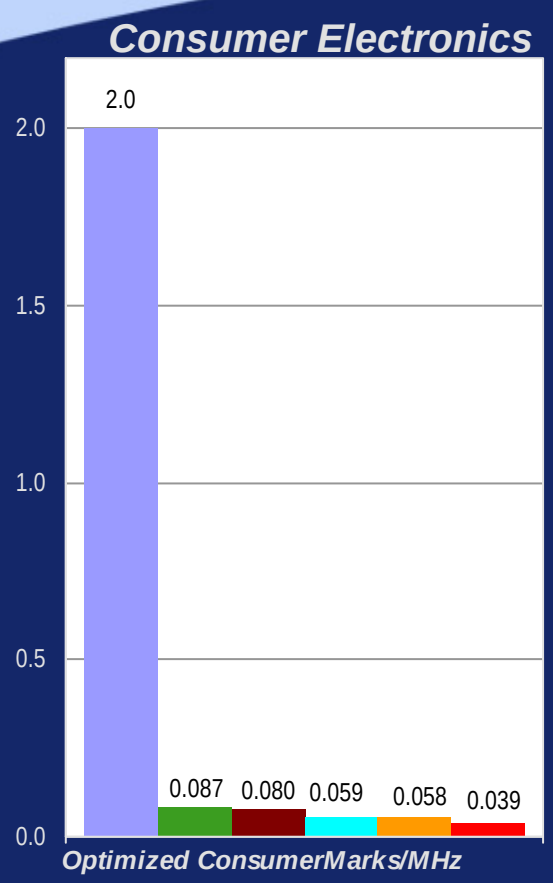
Results for all reported cores:

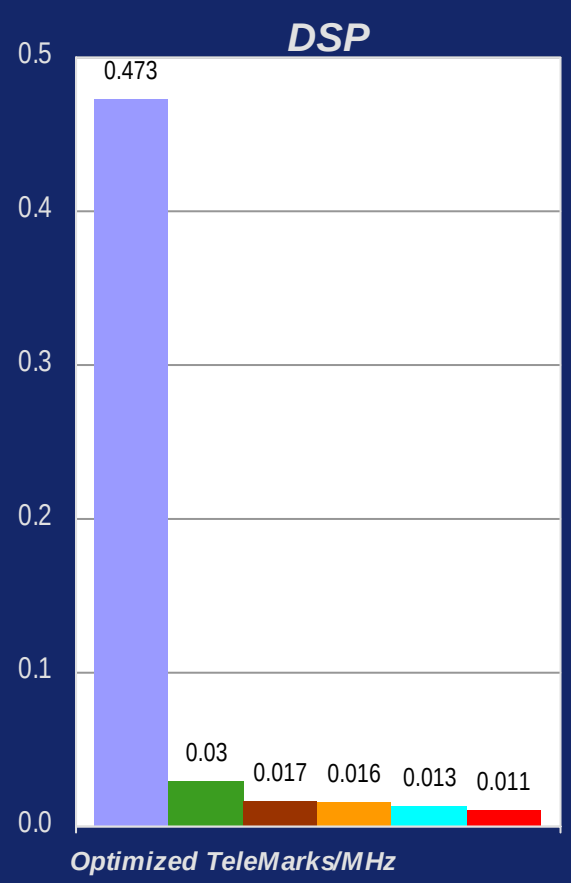
- Instruction set synthesis beats all other processors by 4-8x

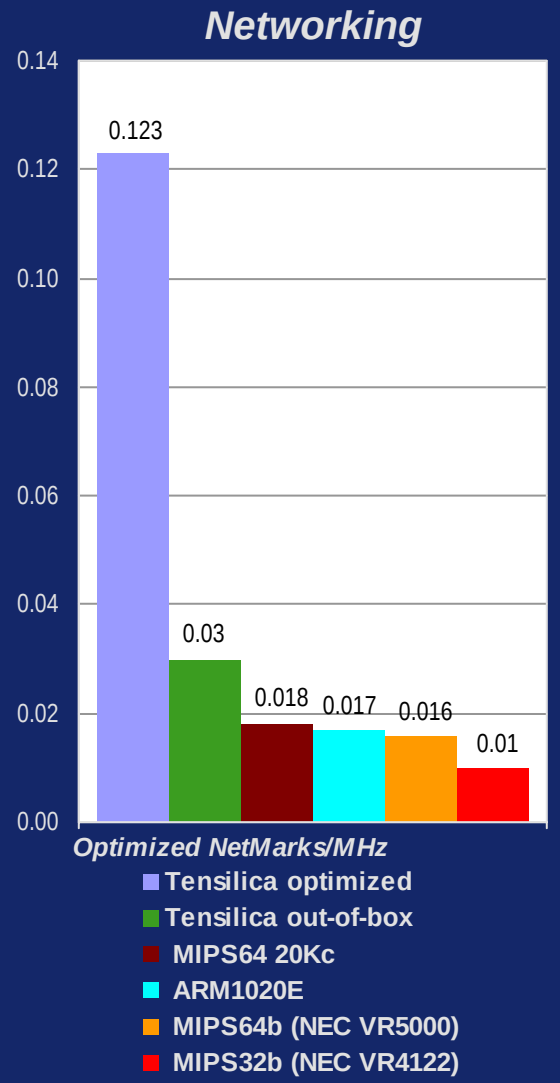


Explicit Instruction Set Optimization

Tensilica Instruction Extension



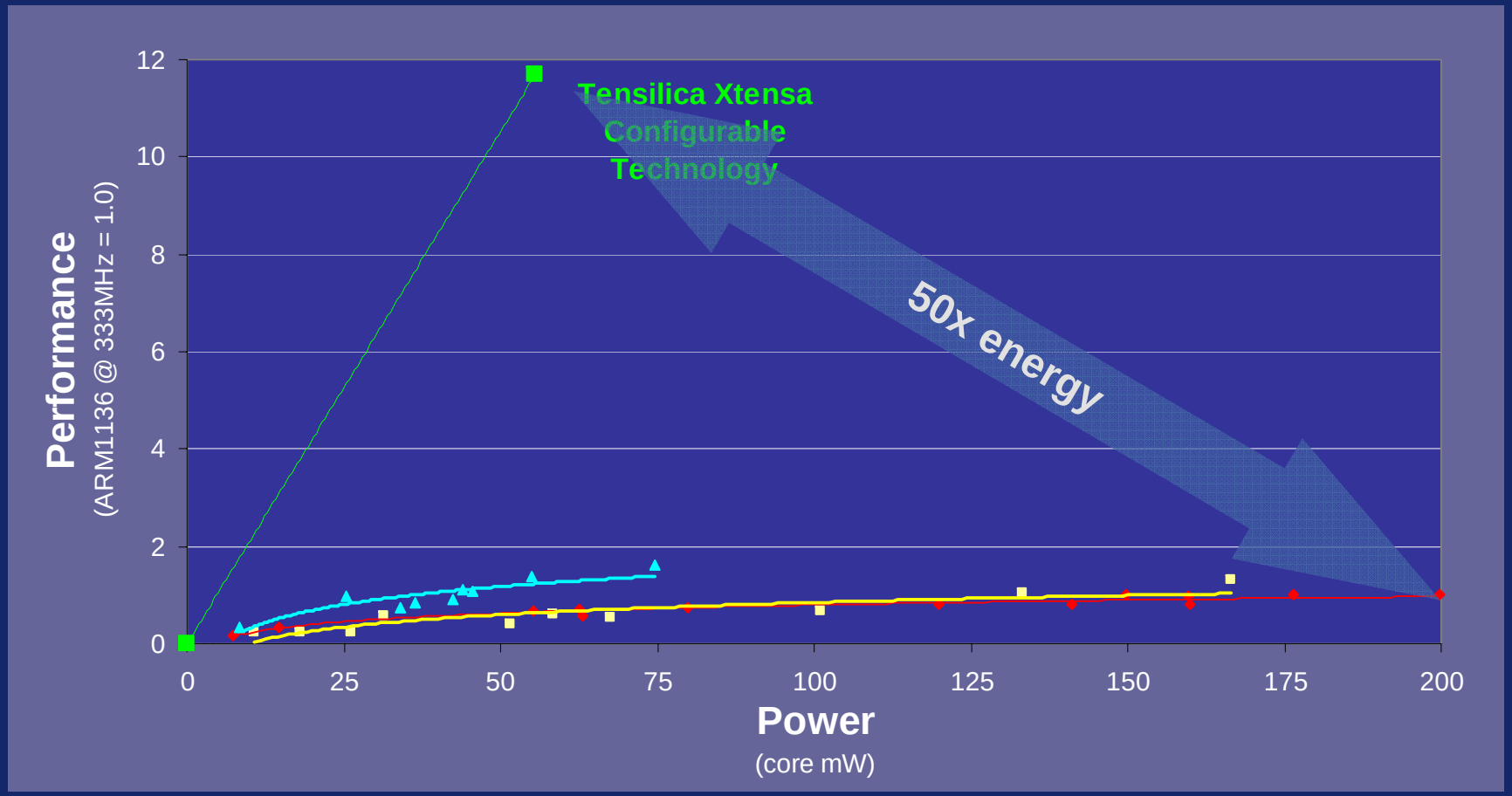




Source: EEMBC Certified Benchmarks

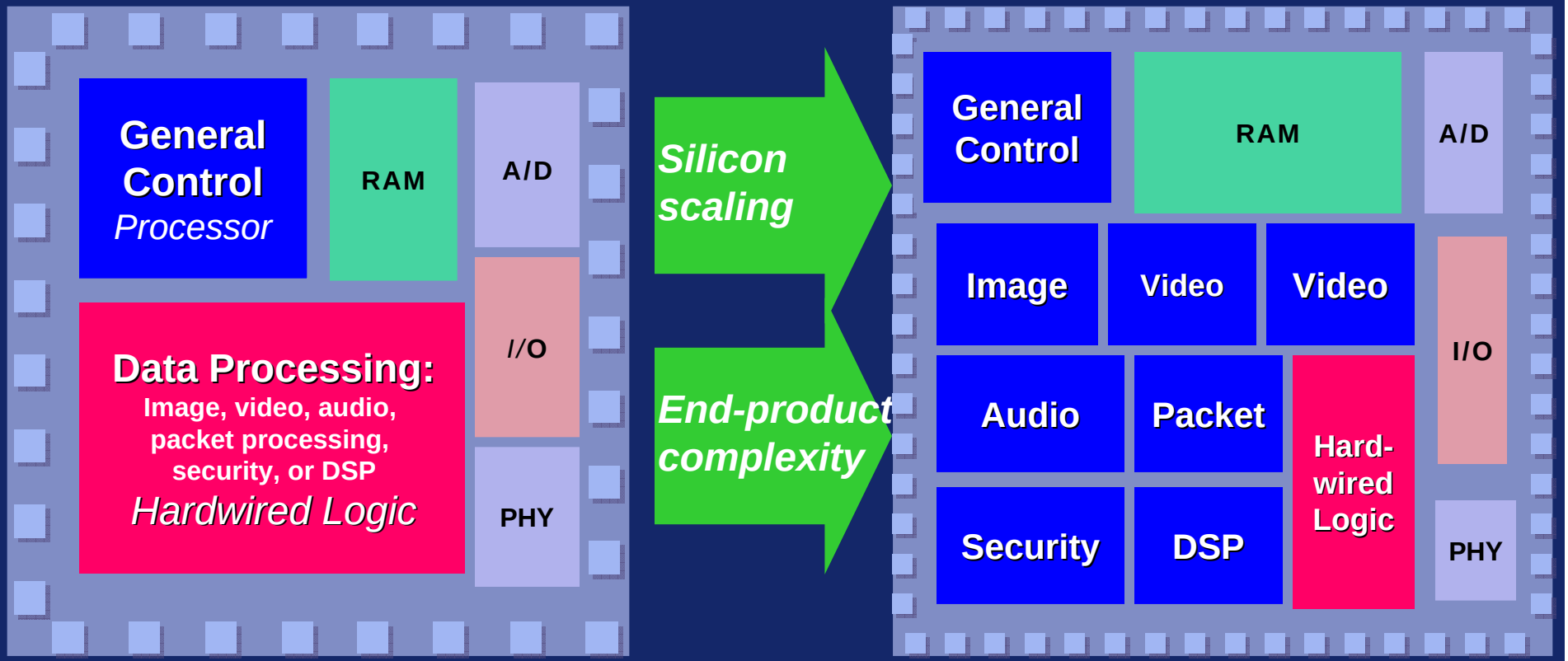
Processor Power and Performance

Xtensa Application-Specific Cores



Performance on EEMBC benchmarks aggregate for Consumer, Telecom, Office, Network, based on ARM1136J-S (Freescale i.MX31), ARM1026EJ-S, Tensilica Diamond 570T, T1050 and T1030, MIPS 20K, NECVR5000). MIPS M4K, MIPS 4Ke, MIPS 4Ks, MIPS 24K, ARM 968E-S, ARM 966E-S, ARM926EJ-S, ARM7TDMI-S scaled by ratio of Dhrystone MIPS within architecture family. All power figures from vendor websites, 2/23/2006

Basic Transition in Chip Design



Traditional View of "System on Chip"

Multiple Processor SOC

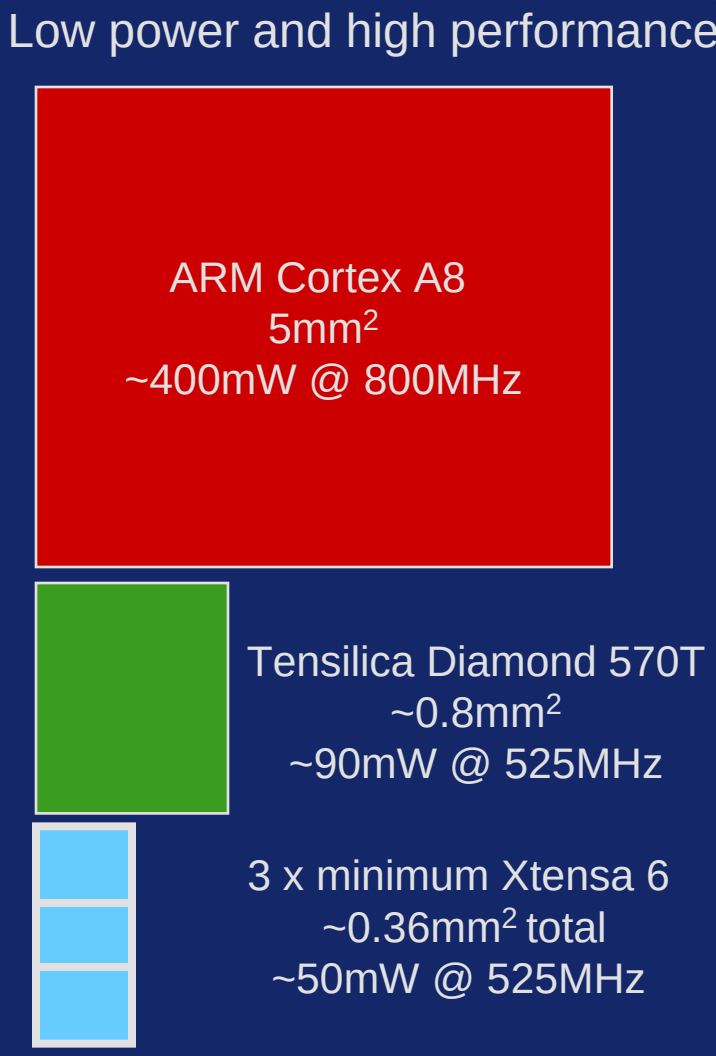
Low Power and Multiple Processors

Why MP?

- Real systems implement multiple concurrent functions
- Demand for flexibility turns multiple hard-wired blocks into multiple processors

MP from the ground up

- MP architecture
- MP SW development tools
- High bandwidth/low latency
- Novel MP interconnects



Optimizing Data Movement

Traditional processor-processor data transfer:

1. Proc 1: application writes local mem
2. Proc 1: OS reads local mem, writes shared mem
3. Proc 2: OS reads shared mem, writes local mem
4. Proc 2: application reads local mem

Result: 2 bus transfers, 3 reads, 3 writes

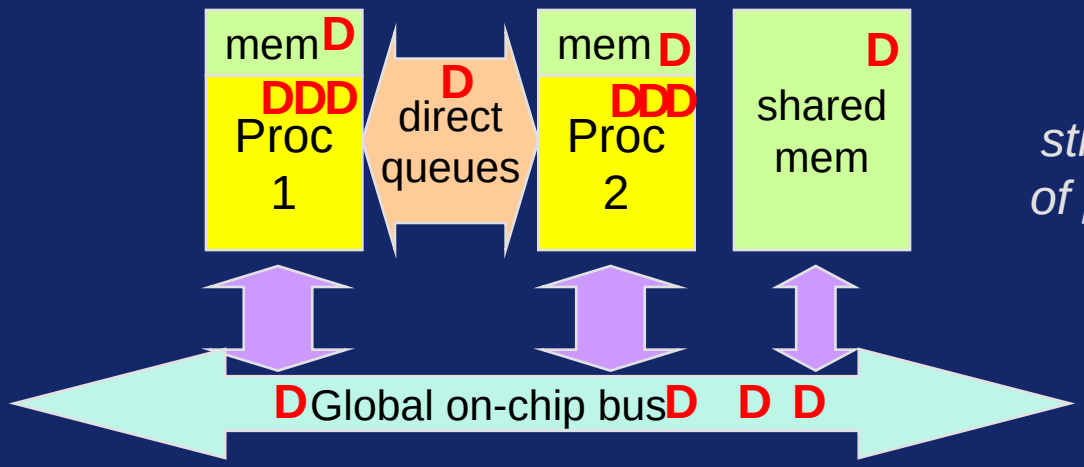
Optimized: 2 bus transfers, 1 read, 1 write

Optimized processor-processor data transfer:

1. Proc 1: application writes to Proc 2 mem
2. Proc 2: application reads local mem

Result: 1 bus transfer, 1 read, 1 write

Optimized with direct queues: 0 bus transfers, 0 reads, 0 writes



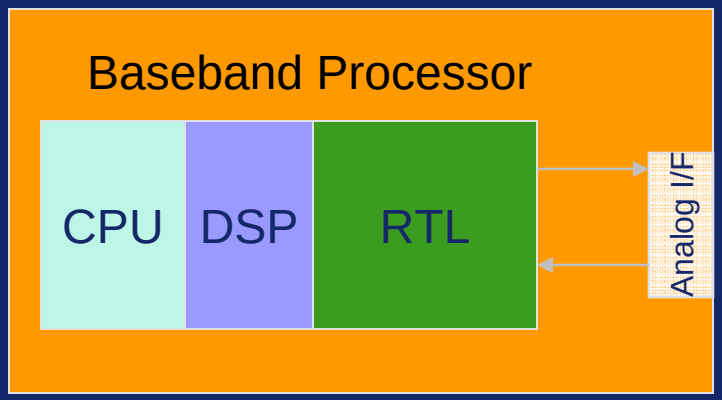
Memory and bus structures consume lots of power when accessed

The Explosion of Wirless Modem Standards:

CDMA, 4G, 802.11*, 802.16*, T-DMB, DVB-H, MediaFlo, HSDPA, HSUPA, UWB, BT

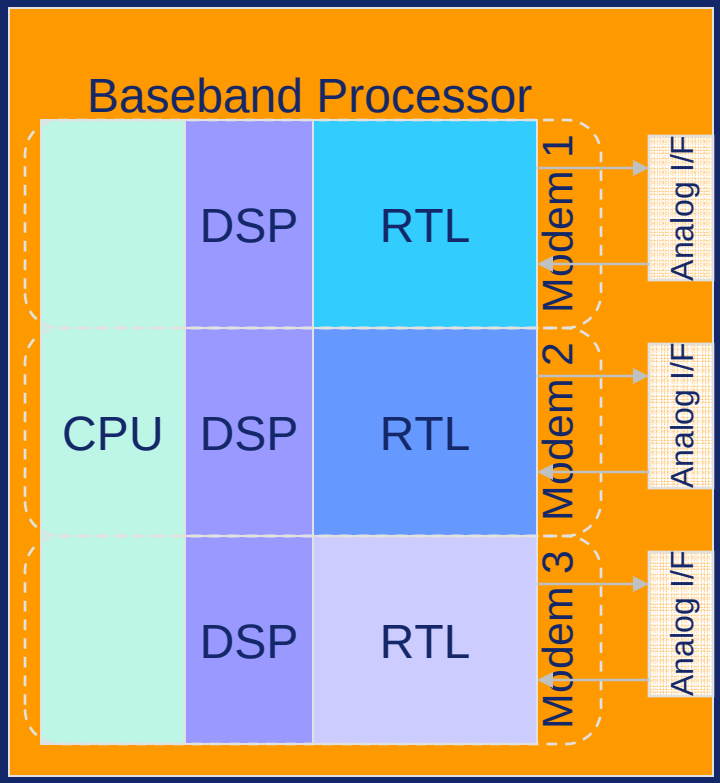
For a single, simple modem:

- DSP+RTL, CPU+RTL or CPU+DSP+RTL
- Complex RTL data-path needed for performance and efficiency in
  - •modulation and filtering
  - •error coding
  - •encryption
- Tasks requiring flexibility run on modest-performance CPU or DSP



As products require multiple (more complex) modems:

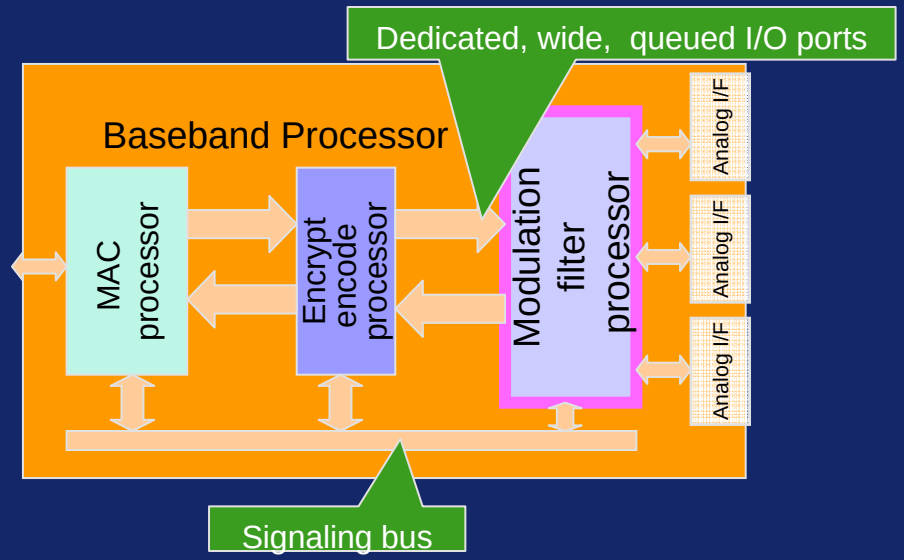
- Duplication of modem-specific RTL
- Difficult design, verification and evolution of RTL for complex functions
- Not all modems used simultaneously



Improved architecture for multiple complex radios

Optimized multi-standard baseband architecture using processor engines

- Common baseline processor and tools
- Switches seamlessly between modem firmware personalities
- Optimized processor engines rival RTL for performance, cost and power



Key questions:

Can processor achieve the necessary performance?

How efficient are processor engines compared to RTL?

- Area
- Power

What is the benefit of common architecture?

- Common tools
- Function migration
- Rapid development

Design Example:
OFDM Modulation Processor

The challenge:

A very fast engine for OFDM FFT and filter operations

Goals (90GT process):

- 1024pt complex FFT in < 4 microseconds (1 radix-4 butterfly/cycle)
- Complex FIR rate: 2B complex taps per second (4 taps per cycle)

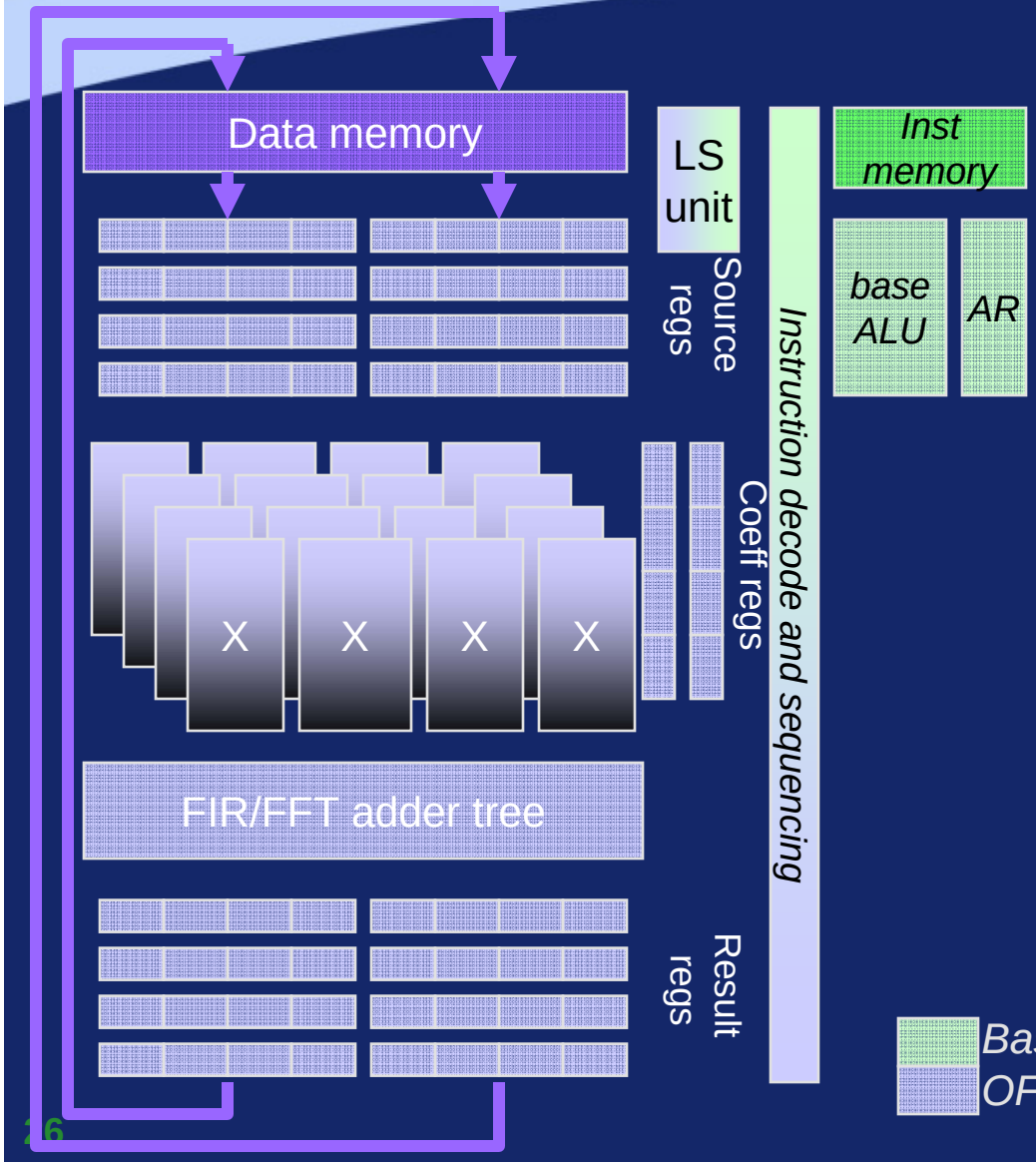
Approach: Specialized DSP architecture with wide-deep computation pipelines

- 2 x 128b XY memory system
- 16 16b partial product multipliers
- 52 32b adders
- Built as extension to 32b Xtensa baseline engine

How does Specialized DSP compare to RTL-equivalent?

- Structure
- Area
- Power

OFDM Modulation Processor



- Simple configuration of Xtensa processor written in Tensilica Instruction Extension (TIE) format
- Dual-port 128b data memory
- 16/24/32b instruction formats
- 32b instructions have 3 slots: X memory op, Y memory op, FFT/FIR op
- 7 stage processor pipeline with 4 stage computation data-path
- 40 source registers (16b+16b)
- 32 result registers
- JTAG-based source-level debug
- Maximum performance FFT code written in ANSI C with generated extensions (<1800 cycles, 371 bytes of code)
- Optional 128b bus interface with in-bound DMA support (+48K gates)
- Optional 128b input and output queues for direct data-streaming (+15K gates)

Base processor engine logic
OFDM Modulation processor extension

Specialized Processor vs. RTL

- Assert processor extension hardware closely equivalent to hardwired RTL design
- Processor overhead, compared to RTL data-path only
  - +22% in gates
  - +24% in area (die cost impact: ~$0.025)
  - +7% in power (for same clock gating)
- Modest overhead for faster design, programmability and full debuggability

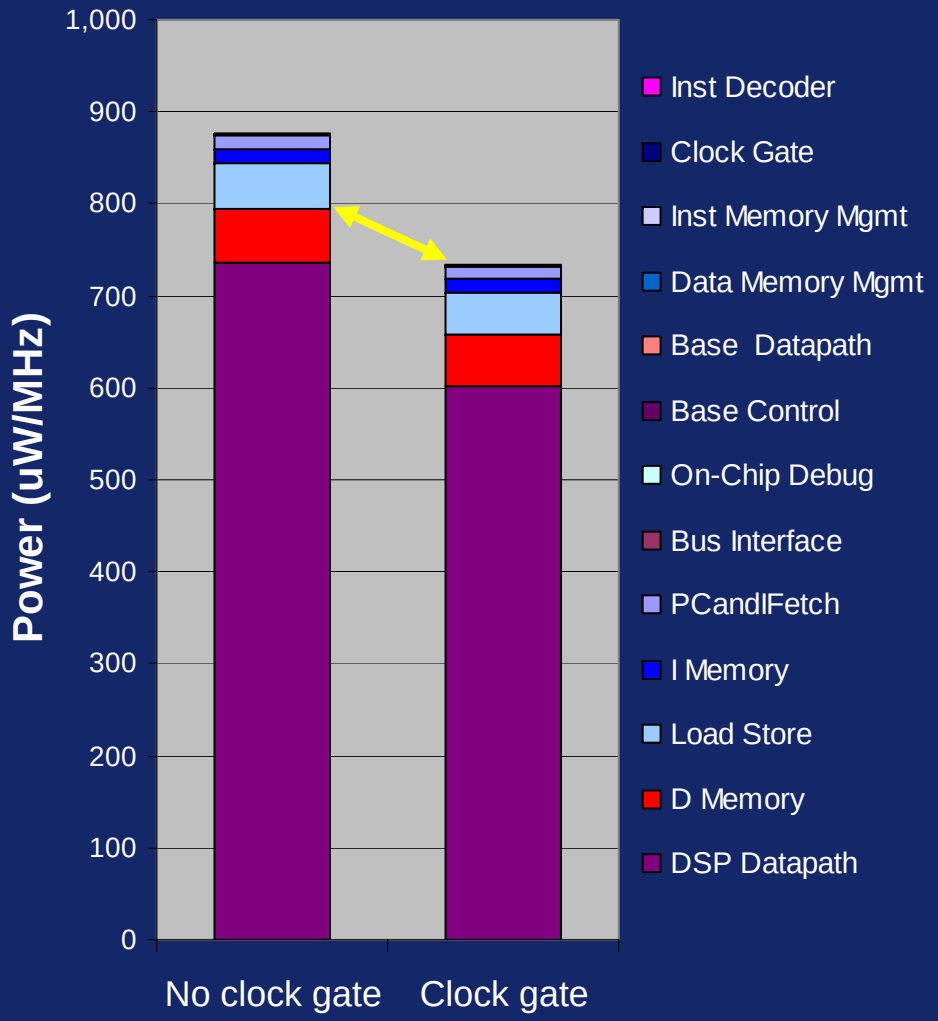
90GT process	Base processor	Instruction memory	Data Memory	DSP Extensions	Total processor	RTL Equivalent
Gates (or bytes)	31Kg	4KB	8KB	159Kg	194Kg + 12KB	159Kg + 8KB
Layout Area (75% util, mm^2)	0.17	0.09	0.24	0.85	1.35	1.09
Power (μW/MHz)	30	16	58	630	734	688

Preliminary data

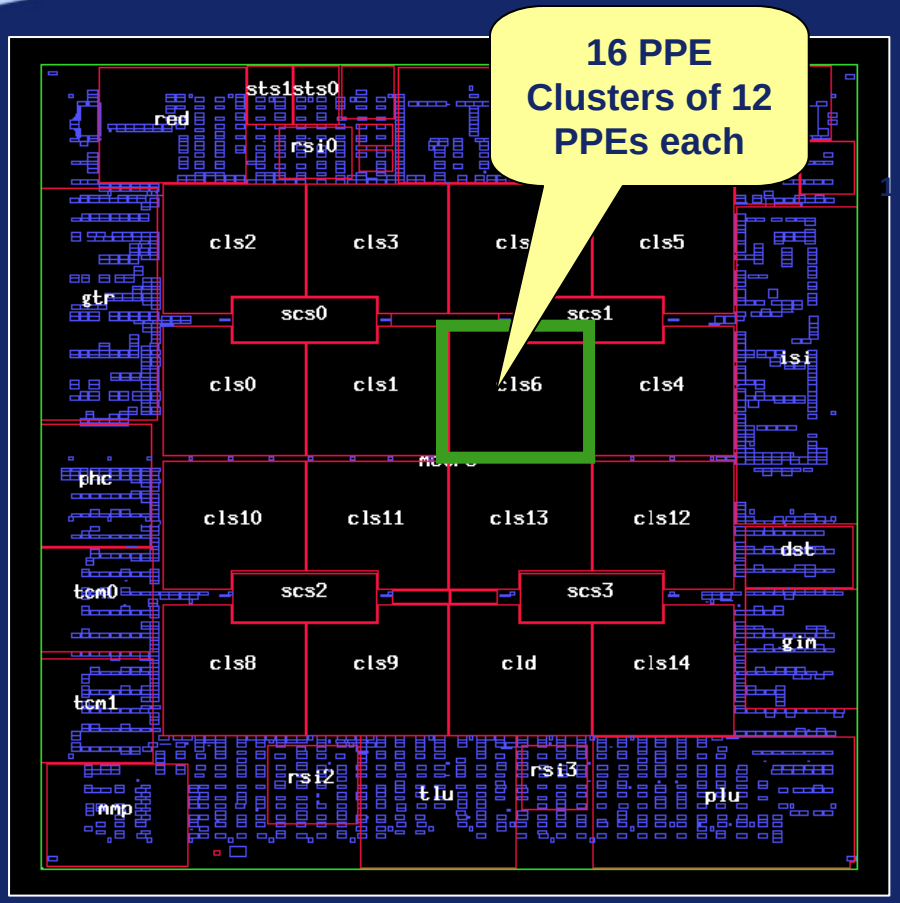
Power in 90GT at 500MHz: ~330mW (1200nJ)
Power in 90G at 350MHz: ~130mW (~700nJ)
Compare to 90nm TI C6416 DSP core @ 1GHz: 6.0 μs x >800mW = ~5000nJ
Note: energy for lower performance FFT can be even better

Processor is Lower Power than RTL

- Processor generator does automatic pipeline activity analysis and gates clocks cycles by cycle.
- Clock gating in OFDM Modulation Processor:
  - **Base processor: 48 gated domains**
  - **DSP Extensions: 199 gated domains**
- Designers can implement clock gating by hand, but often don't have time or skill
- Benefit of auto-clock-gating outweighs overhead of processor logic + instruction memory
- Compare power (μW/MHz):
  - **No clock gate DSP Datapath + DataRAM: 795**
  - **Clock gate full processor: 734**



Example Parallel Architectures
Cisco CRS-1 Terabit Router



CISCO SYSTEMS

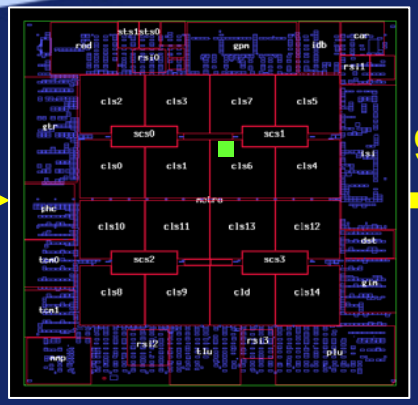
188 Xtensa network processing cores per Silicon Packet Processor

Up to 400,000 processors per system

Example Parallel Architectures

CRS-1: Massive general-purpose throughput

96Gb/s



96Gb/s

18mm x 18mm IBM 0.13μm
18M gates
8Mbit SRAMs

50,000 general purpose MIPS
175 Gb/s memory bandwidth

Programmability also means

- Ability to juggle feature ordering
- Support for heterogeneous mixes of feature chains
- Rapid introduction of new features

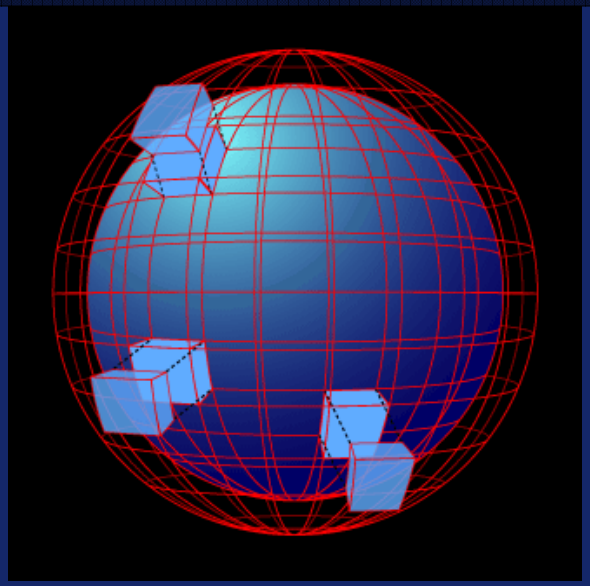
Routing task runs to completion on each processor:

- IPv4 Unicast
- MPLS–3 Labels
- Link Bundling (v4)
- Load Balancing L3 (v4)
- 1 Policier Check
- Marking
- TE/FRR
- Sampled Netflow
- WRED
- ACL
- IPv4 Multicast
- IPv6 Unicast
- Per prefix accounting
- GRE/L2TPv3 Tunneling
- RPF check (loose/strict) v4
- Load Balancing V3 (v6)
- Link Bundling (v6)
- Congestion Control

Example Parallel Architectures
Petascale Climate Modeling System

Technical/ Economic Challenges

1. Variance in data-reference/ communication patterns for codes
2. Potential for extreme scalability via large-scale processing arrays
3. Size, cost and maintenance strongly correlated to system power dissipation
4. General-purpose CPUs optimized for integer applications – unimpressive performance per $, per watt



Parallel Climate Modeling

Lenny Oliker and Michael Wehner of Lawrence Berkeley Lab speculation: a much more parallel climate model

- 1.5km grid for Earth
- 20,000,000 domains
- 500 MFLOPS/domain
- 500 MB/s per domain
- Complex algorithms require general-purpose programmability in double precision floating point
- 2D communications mesh @~20MB/s per domain

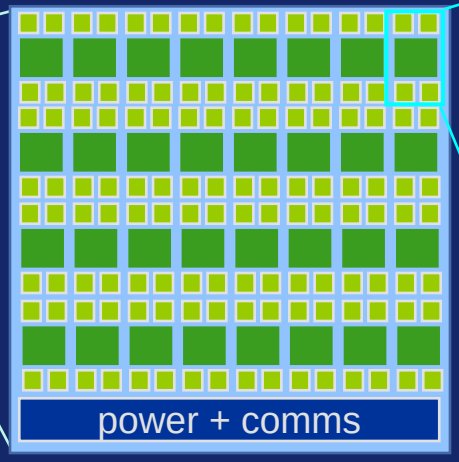
System Architecture Approach

- Highly suitable for distributed array computation
- Two design challenges:
  - Total memory bandwidth: 5-10 peta-bytes/s – many parallel local DRAM channels
  - Power: GFLOPS/W best predictor of system cost, size
- Best off-the-shelf processor (IBM Cell) is about 0.5 DP GFLOPS/W
- Domain-specific processor approach offers significant potential advantage (>10 DP GFLOPS/W)

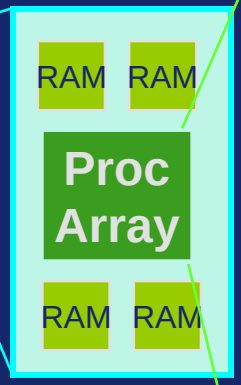
Example Parallel Architectures

10 PetaFLOPS System Concept: 3.8M processors

120 racks @ ~25KW



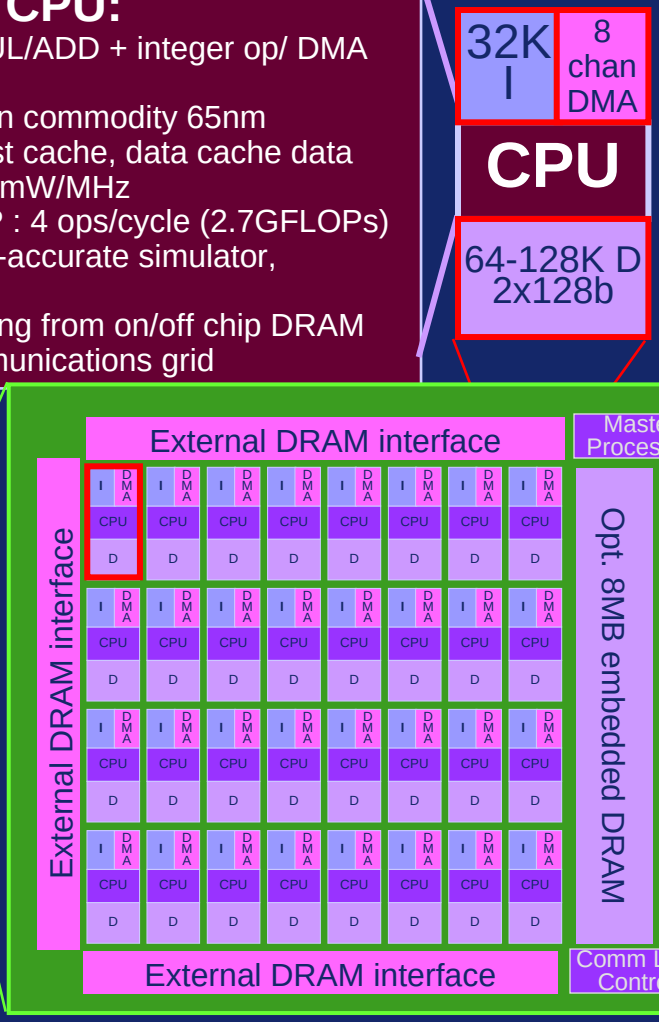
32 chip + memory clusters per board (2.7 TFLOPS @ 1000W



8 DRAM per processor chip: 50 GB/s

VLIW CPU:

- 128b load-store + 2 DP MUL/ADD + integer op/ DMA per cycle:
- Synthesizable at 650MHz in commodity 65nm
- 1mm² core, ~3mm² with inst cache, data cache data RAM, DMA interface, 0.25mW/MHz
- Double precision SIMD FP : 4 ops/cycle (2.7GFLOPs)
- Vectorizing compiler, cycle-accurate simulator, debugger GUI
- 8 channel DMA for streaming from on/off chip DRAM
- Nearest neighbor 2D communications grid



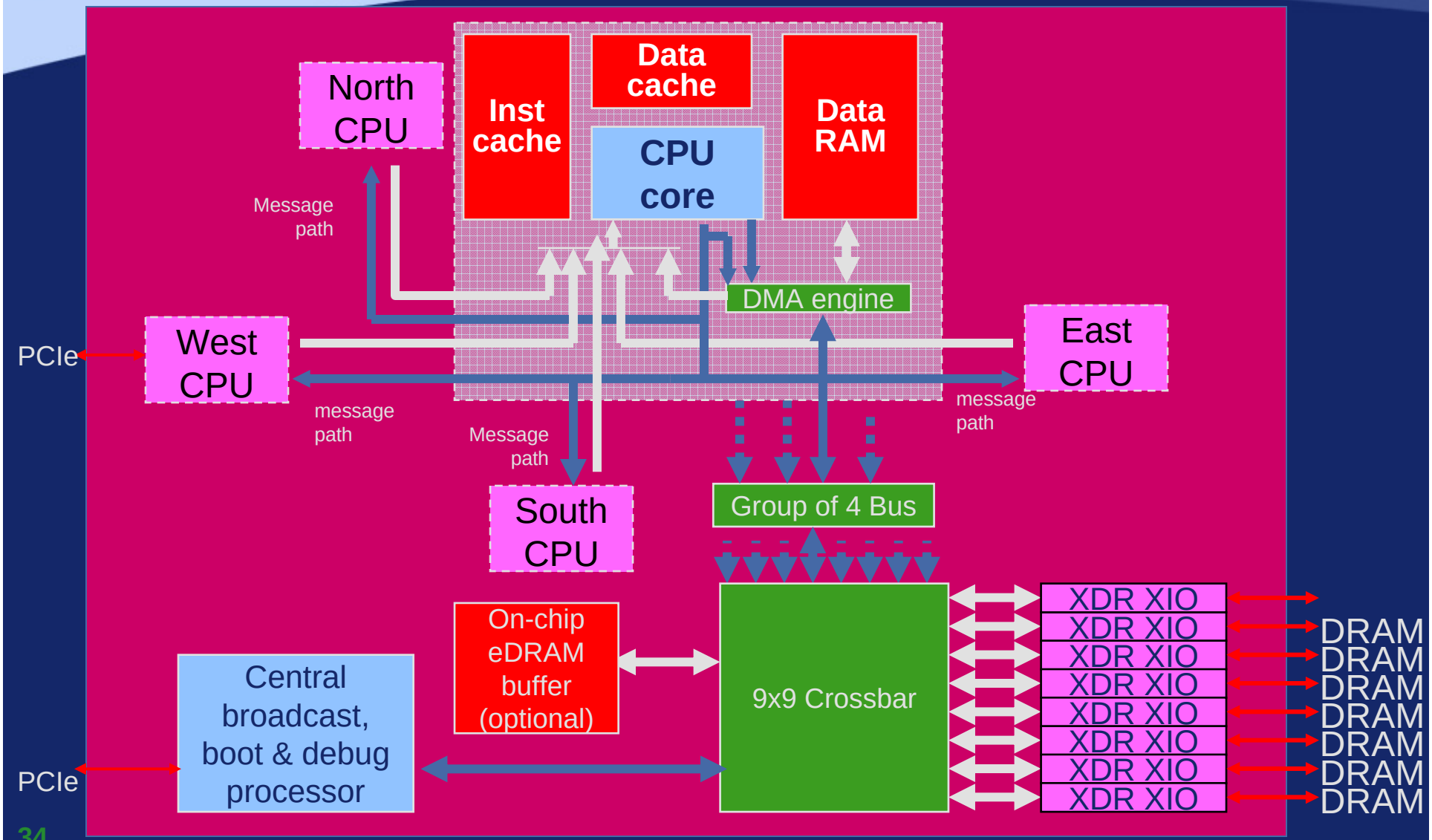
32 processors per 65nm chip
83 GFLOPS @ 12W

Example Parallel Architectures
Optimized Scientific Compute Processor

- Optimized for general-purpose double-precision computation on well-structured local data.
- Three-issue "FLIX" VLIW:
  1. 128b load-store (with update), FP convert, general integer ops
  2. general integer op
  3. 2-way SIMD FPop (IEEE add, sub, mul, mul-add, mul-sub)
- Free intermixing of 16b, 24b and 64b FLIX instructions
- 8-stage pipe with zero-overhead loops
- 32 entry windowed address register file
- 16 entry SIMD FP register file (16 x 2 x 64b)
- 32KB I cache + 8KB D cache + 64-128KB data RAM
- Local instruction and data cache plus large local data RAM (64K-128KB), dual-ported between processor and closely-coupled DMA engine
- Core: 220K gates = 1mm^2 <100mW@650MHz

64b instruction word		
LS convert integer	integer ops	SIMD FP ops
LDI[U] LVI[U] LVX[U] SD{H.L}I[U] SVI[U] SVX[U] TRUNC{H,L} CEIL{H,L} ROUND{H,L} FLOOR{{H,L} FLOAT{H,L} UFLOAT{H,L} VMOV VNEG VABS + 45 integer ALU and LS	27 integer ALU	VADD VSUB VMUL VMADD VMSUB RADD SWAP V[compare] VMOV VMOVEQZ VMOVGEZ VMOVLTZ VMOVNEZ VMOVT VMOVF

Example Parallel Architectures
Local Interconnect Structure



The New World
Automated MP SOC Design Flow

Complex system definition

Rapid partitioning into a set of communicating tasks

Optimized HW/SW communication on tuned processors and interconnect

Physical synthesis of hardware + high-accuracy system software integration