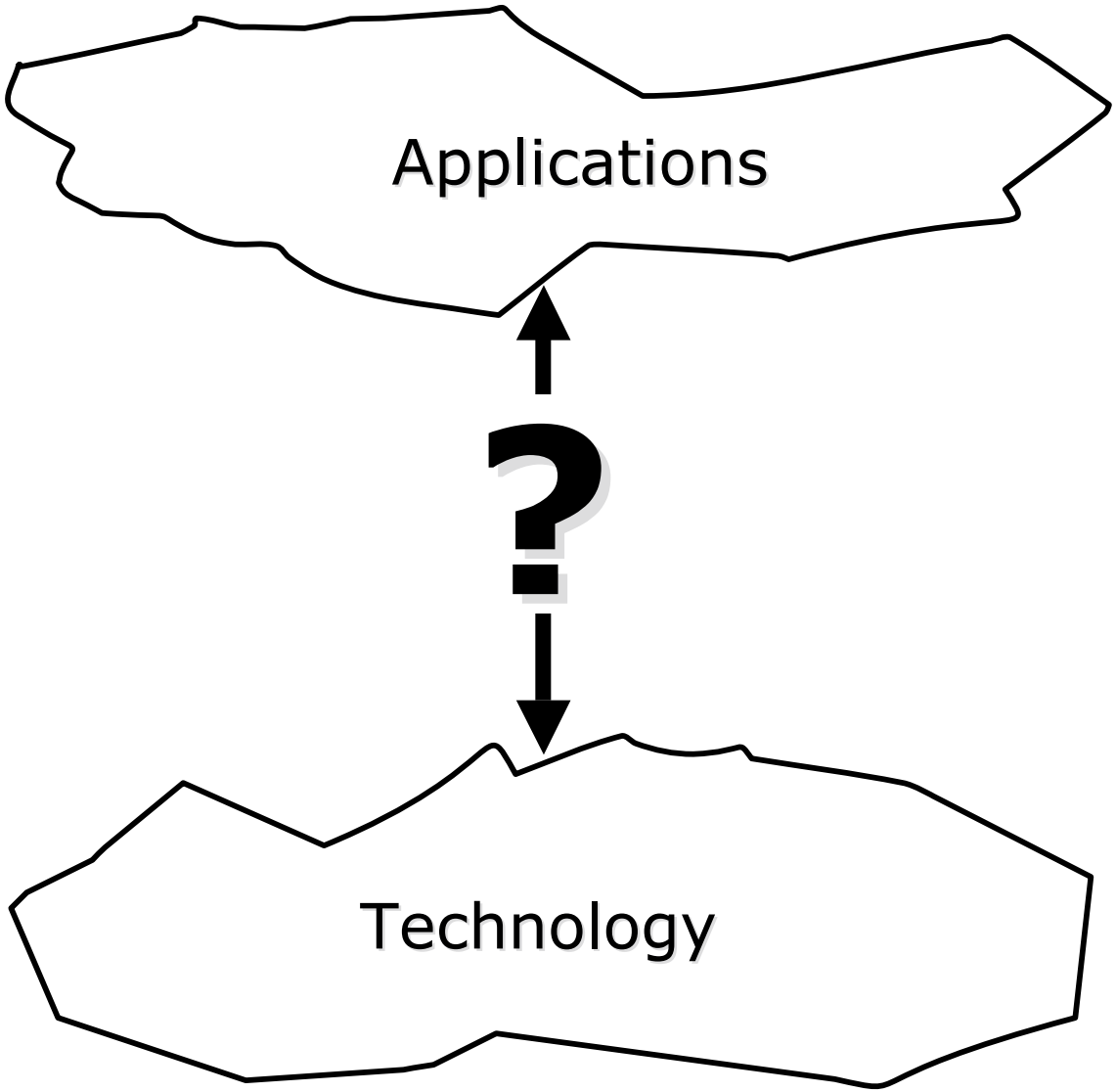
The First Twenty Years, The First Twenty Chips

Krste Asanovic
ICSI Architecture Group,
EECS Dept., UC Berkeley,
& Lawrence Berkeley National Laboratory

ICSI 20th Anniversary Celebration
October 17, 2008

The more things change,
the more things stay the same…

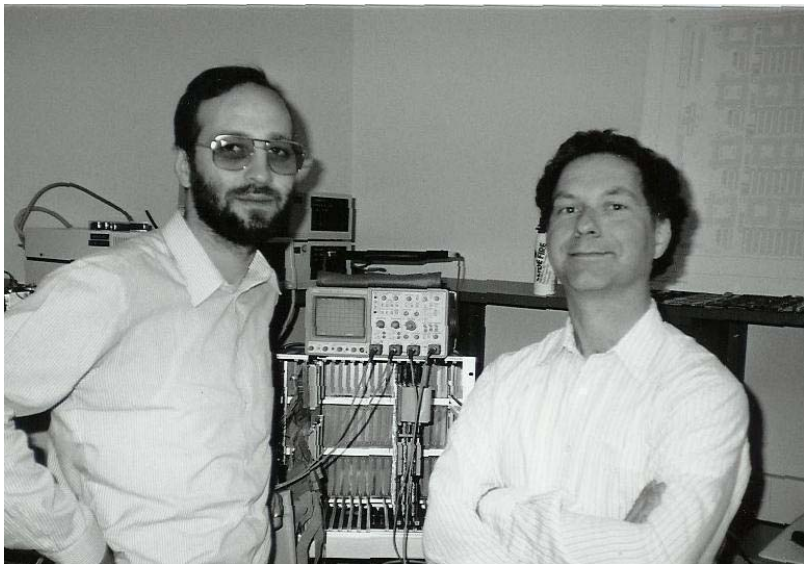
Computer Architecture: 30,000ft view



Ring Array Processor, 1989

(Nelson Morgan, Jim Beck, Phil Kohn, Jeff Bilmes)

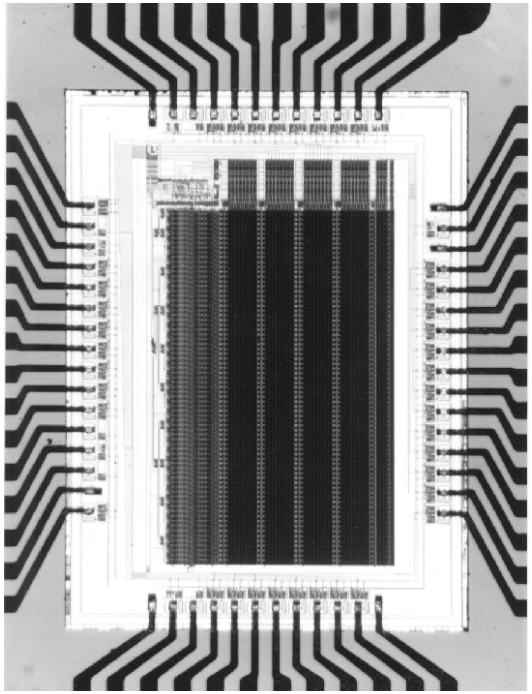
- RAP Machine under development for fast training of neural networks for speech recognition
- Ring of TMS320C30 floating-point DSPs
  - Each DSP providing 32MFLOPS
  - Four DSPs/board, up to 10 boards connected at once (>1GFLOP/s peak, 640MB DRAM)
  - Neural net training rate of >100MCUPS (million connection updates per second) on 10 boards
- Fast, flexible, but expensive
  - ~$100,000 each

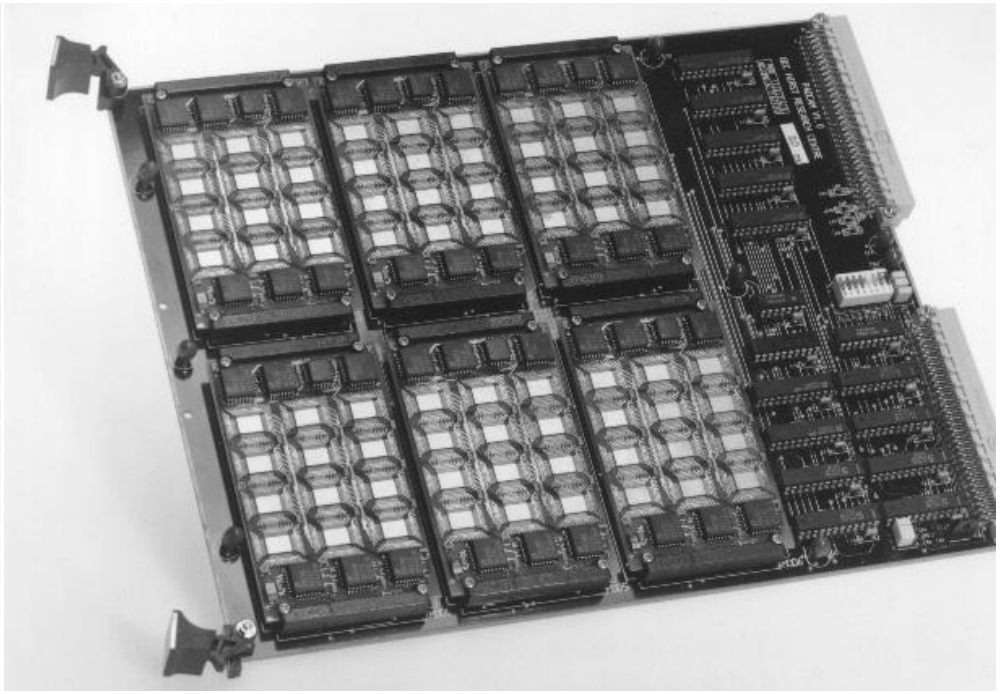


PADMAVATI/SPACE (1987-89)
GEC, UK

- Target Application: Natural Language Processing and Image understanding using Lisp and/or Prolog
- 170,000 36-bit associative processors
  - 148 per chip
- Controlled by 16 transputers

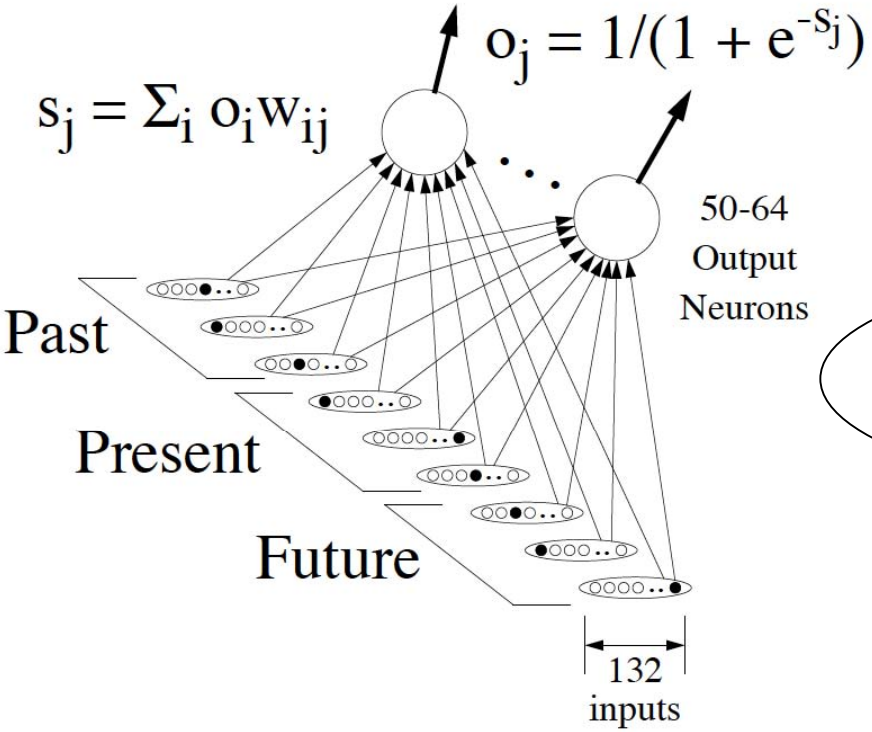
1.2μm CMOS
5.8 x 7.9mm^2
8 MHz





ICSI, January 1990

New naïve grad student joins Morgan's group to build custom VLSI for speech training

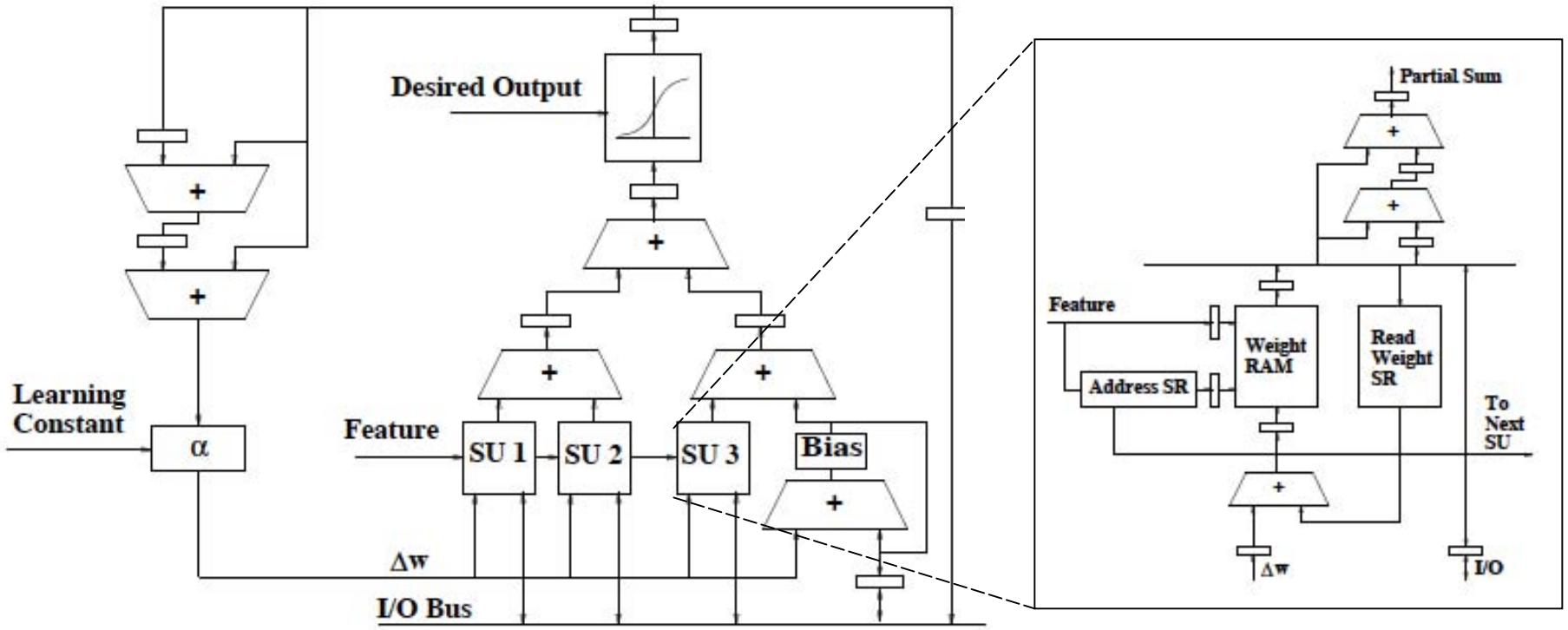


This is a cool ANN architecture for which we need custom silicon!

Training rule: $\Delta w_{ij} = -\alpha(o_j - d_j)\, o_i$

HiPNeT-1: (Highly Pipelined Network Trainer)

Krste Asanovic, Brian Kingsbury, Nelson Morgan, John Wawrzynek

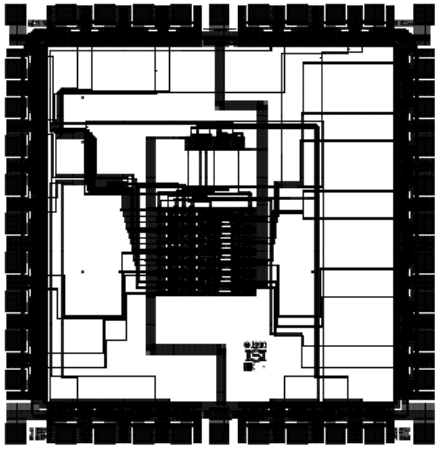


- Custom architecture for neural algorithm
- Predicted 200MCUPS in 16mm^2 of 2µm CMOS running at 20MHz

The first few chips…

- MOSIS had a "TinyChip" program
  - $500 to fab a 2.2mmx2.2mm chip in 2µm CMOS

Sigmoid unit (Pawan Sinha)



8-bit datapath (Krste)

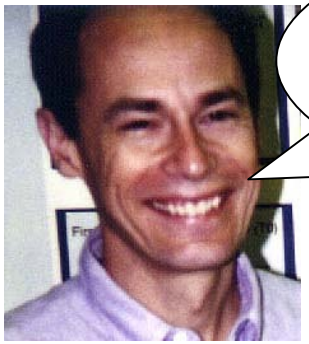
JTAG latches (Krste)

Multiplier (Brian)

24b Adder (Brian)

Regfile (Bertrand)

The infamous static RAM…



I know 45° lines violate the design rules, but it will be much denser!

SRAM (JohnW)

SRAM v2 (JohnW)

SRAM v3 (JohnW)

SRAM v4 (Brian)

Three strikes!
45° are out

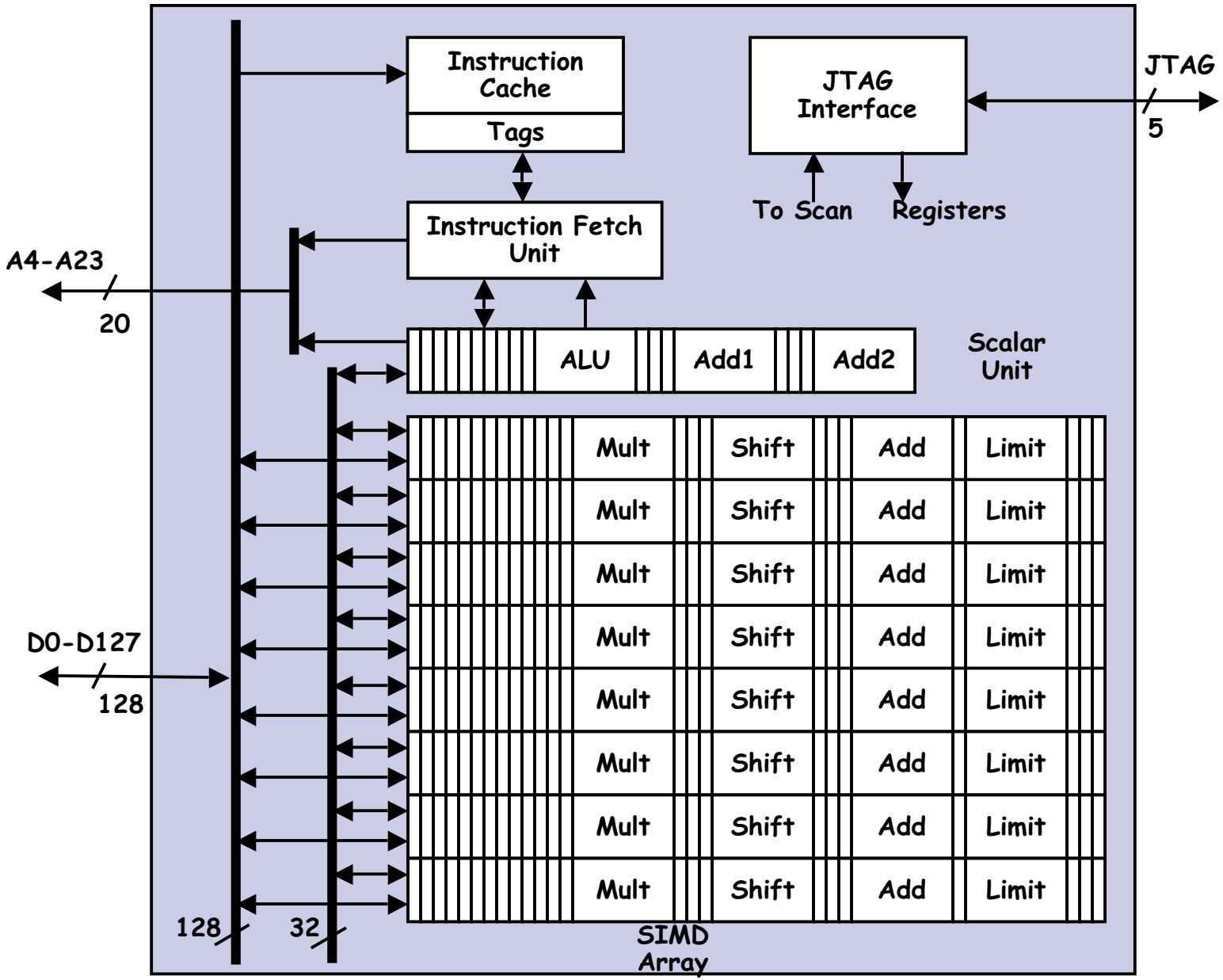
Meanwhile, back at the speech ranch…

There's this even cooler ANN architecture for which we need custom silicon!

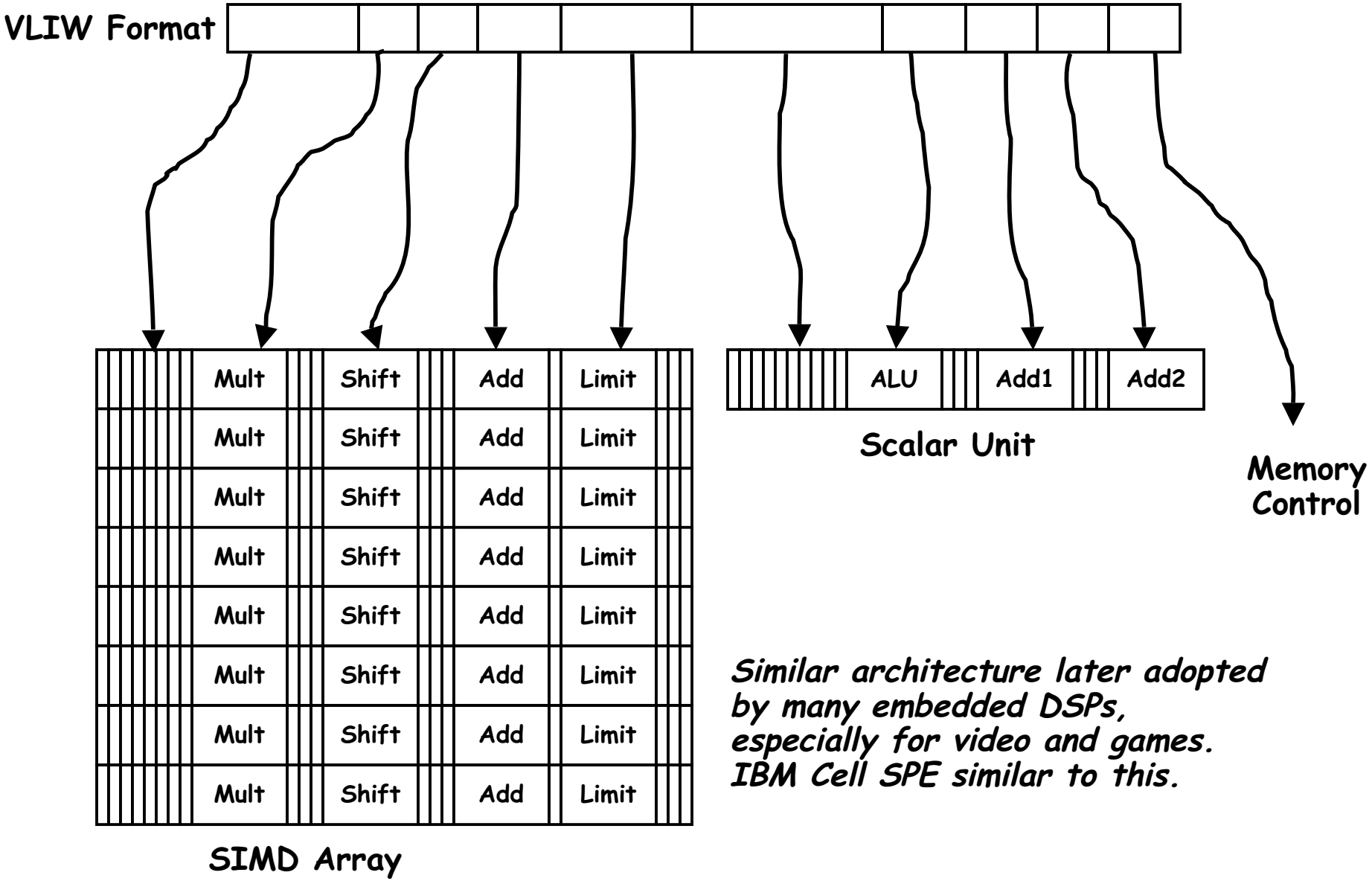
And it doesn't look much like the last one. Can you build a different chip?

Time for a programmable architecture…

"Old" SPERT Architecture

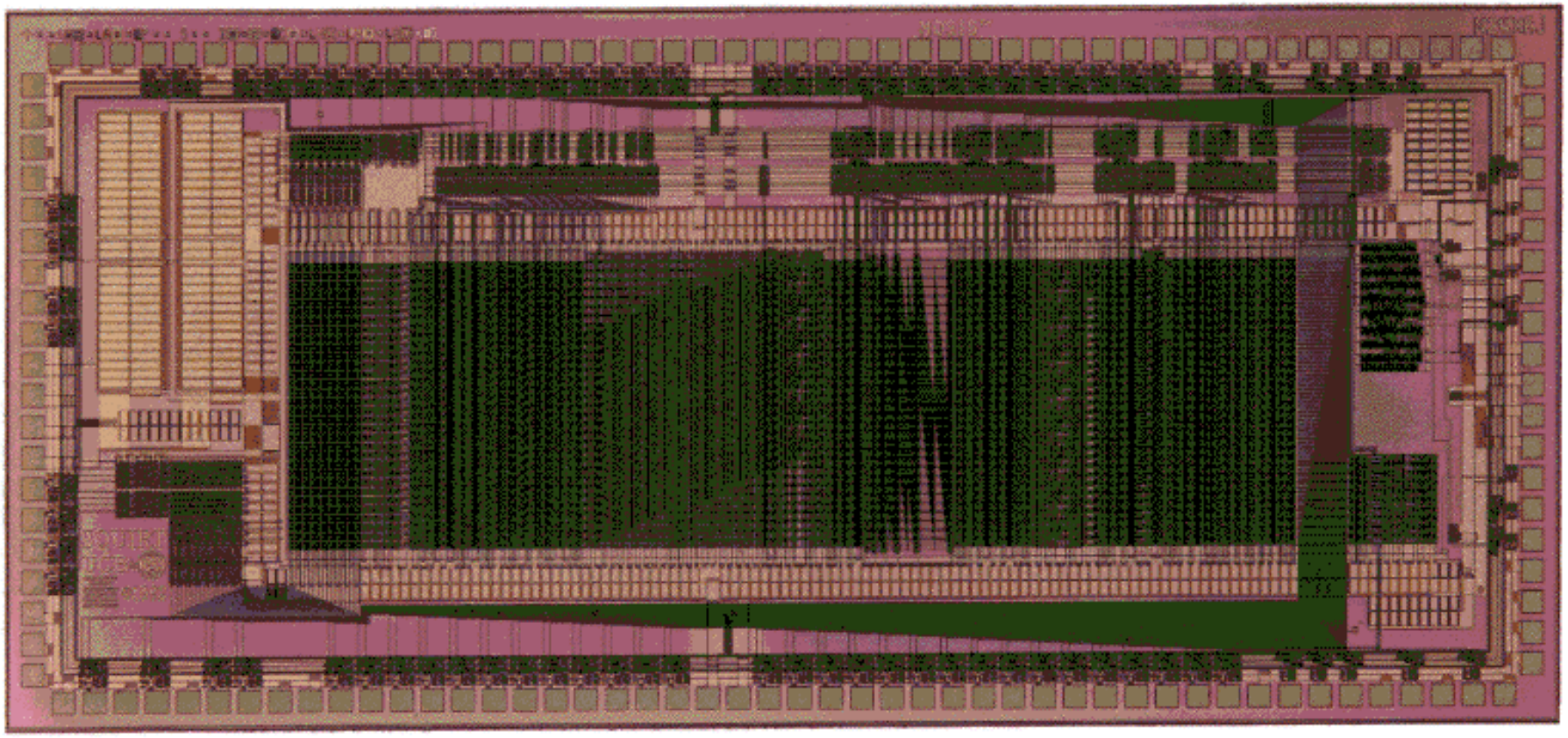


"Old" SPERT VLIW Instruction



Similar architecture later adopted by many embedded DSPs, especially for video and games. IBM Cell SPE similar to this.

SQUIRT Test Chip, 1992



- 1.2μm CMOS, 2 metal layers
- 61,521 transistors, 8x4 mm^2, 400mW@5V, 50MHz
- 72-bit VLIW instruction word
- 16x32b register file, 24bx8b->32b multiplier, 32b ALU/shifter/clipper

CNS-1: Connectionist Network Supercomputer (ICSI/UCB 1992-95)

- *Faculty*

  Jerry Feldman
  Nelson Morgan
  Carlo Séquin
  John Wawrzynek

- *Staff*

  James Beck
  Phil Kohn

- *Post-doc*

  John Lazzaro

- *Students*

  Krste Asanović
  David Bailey
  Tim Callahan
  Ben Gomes
  Bertrand Irissou
  Brian Kingsbury
  Srini Narayaran
  David Stoutamire

- *Visiting Researcher*

  Thomas Schwair

CNS-1 Target Applications

- Speech Research
  - Current Problem —
    - Large layered neural-networks used to estimate phonetic probabilities trained with back-propagation.
    - 1 Million Parameters, 10^{14} arithmetic operations per training run,
    - plus non-neural computations.
  - Later — Unified approach to
    - Analog "front-end" $\Rightarrow$ recognizer $\Leftrightarrow$ language model
- Other
  - Early Vision
  - High-level Vision
  - Simulation of biological neurons and neural masses
  - Functional simulation of hardware

CNS-1 Benchmark

- Benchmark Problem

  Evaluate a network with a million units and an average of a thousand connections per unit for a total of a billion connections. This should be done 100 times per second.

A connectionist accelerator can at best speed up
an application by a factor of
1/(*fraction of non-connectionist computation*).

Equates to around 200GFLOPS
(new Apple MacBook Pro GPUs are 120GFLOPS peak)

CNS-1 Funding

- Office of Naval Research URI Grant (since May 1992)
- National Science Foundation

  Experimental Systems
  PYI award
  Graduate Fellowships
  Mammoth Infrastructure Grant

- ICSI

  Funds provided by ministries of research of
  Germany, Italy, and Switzerland, and cooperating companies.

- ARPA/ONR Grant
- Total approximately $2M per year.

CNS-1 Physical Design (Jim Beck)

First CNS Design review, October 1992

Another Processor for CNS-1

- Started a new architecture, vaguely similar to old-SPERT VLIW-SIMD design
- Then realized vector instruction set would be better

Hold it! This is crazy!!!
We haven't finished SPERT and we're doing another processor?
Who's going to write all the software?

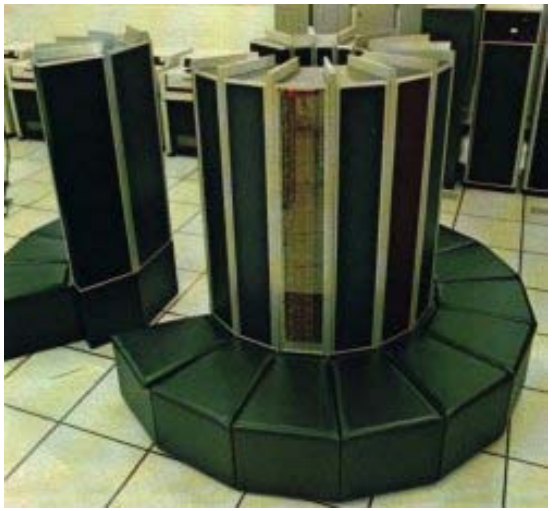
We abandoned old SPERT VLIW

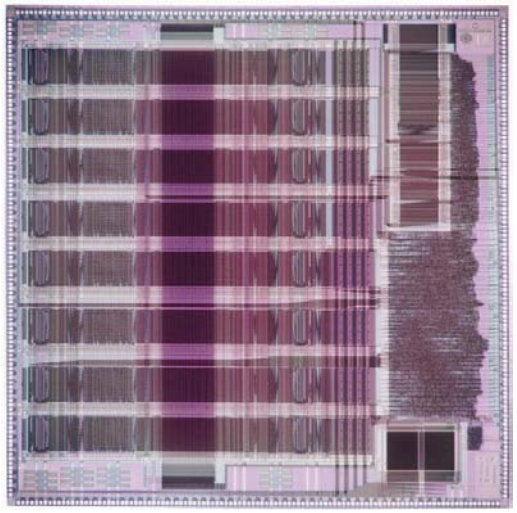
- VLIW means no upward compatibility
  - we wanted same ISA for CNS-1 to reuse software effort
- VLIW scalar compiler was tough
  - Simple VLIW hardware + complex VLIW compiler more work than more complex RISC architecture + standard compiler
- Assembly code was tough to write
  - soon discovered this when writing test code and key loops
- VLIW format too rigid
  - hard to fit some operations into statically scheduled instruction slots (misaligned loads/stores, scatter/gathers)
- VLIW had too large an instruction cache footprint
  - loop prologue/epilogue code plus unrolled loop body

Software, software, software,….

Torrent-0 (T0): A Vector Microprocessor

Vector supercomputers (like Crays) are very successful in scientific computing and have a clean programming model





T0 idea: Add a vector coprocessor to a standard RISC scalar processor, all on one chip

- Primary motivation was software support effort

(Interesting coincidence, T0 and Cray-1 have identical memory bandwidth, 640MB/s)

System Design Choices

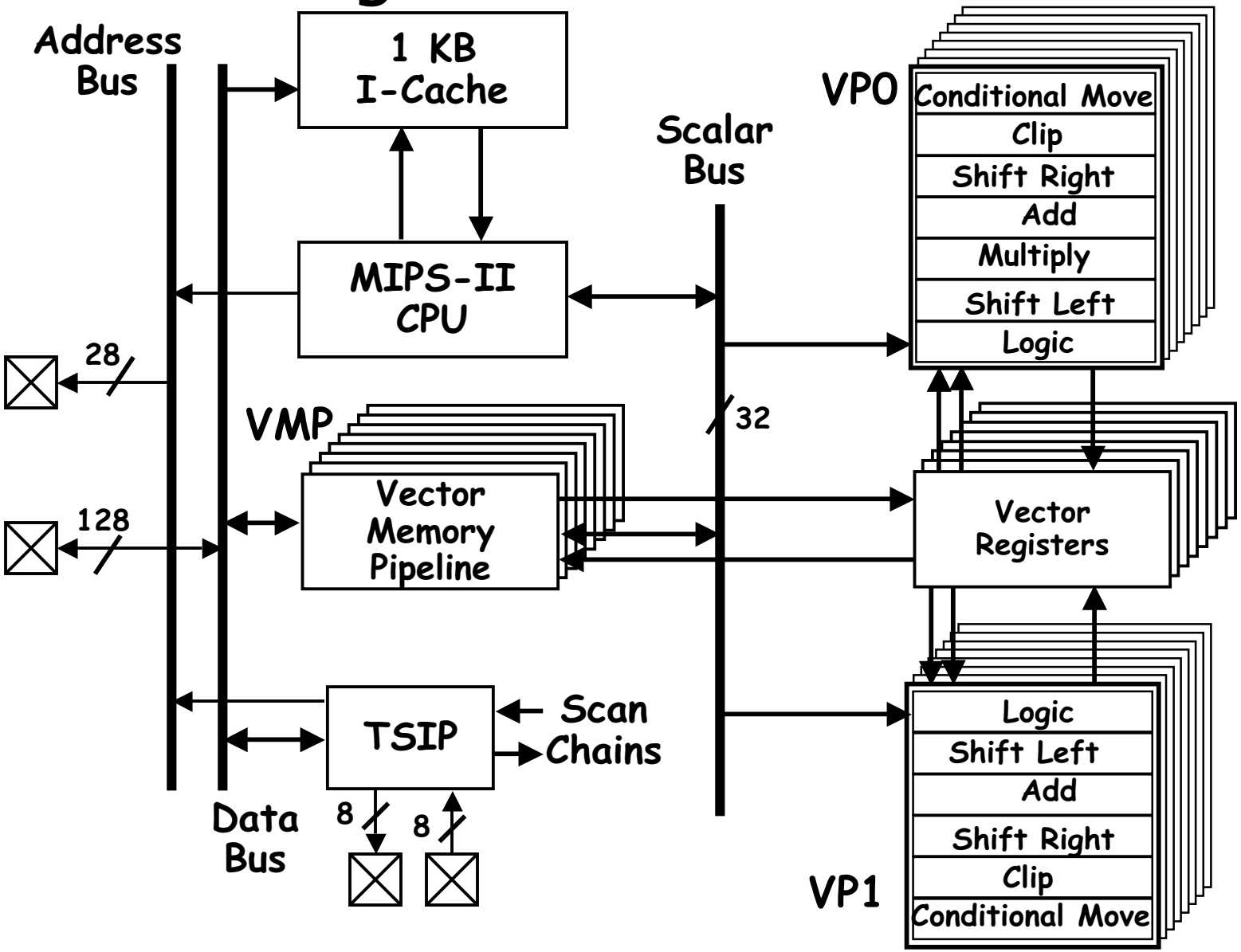
Which standard RISC?

- Considered SPARC, HP PA, PowerPC, and Alpha
- Chose MIPS because it was the simplest and had good software tools and Unix desktop workstations for development, and also had a 64-bit extension path

Buy or build a MIPS core?

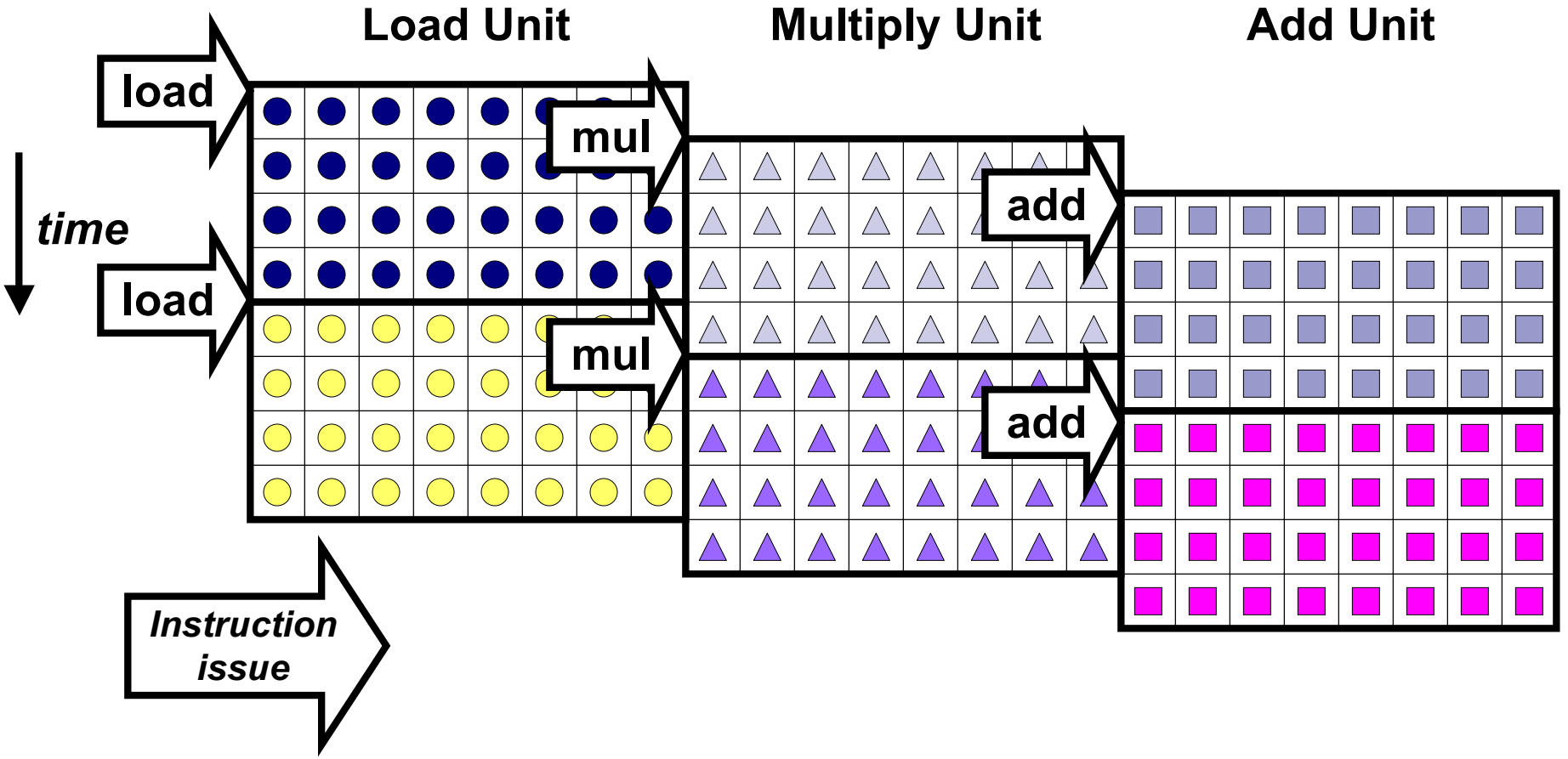
- Commercial MIPS R3000 chips had coprocessor interface
- Decided to roll our own
  - vector coprocessor would have played havoc with caches
  - coprocessor interface too inefficient
  - commercial chip plus glue logic would blow our size and power budgets (to fit inside workstation)
  - couldn't simulate whole system in our environment

T0 Block Diagram



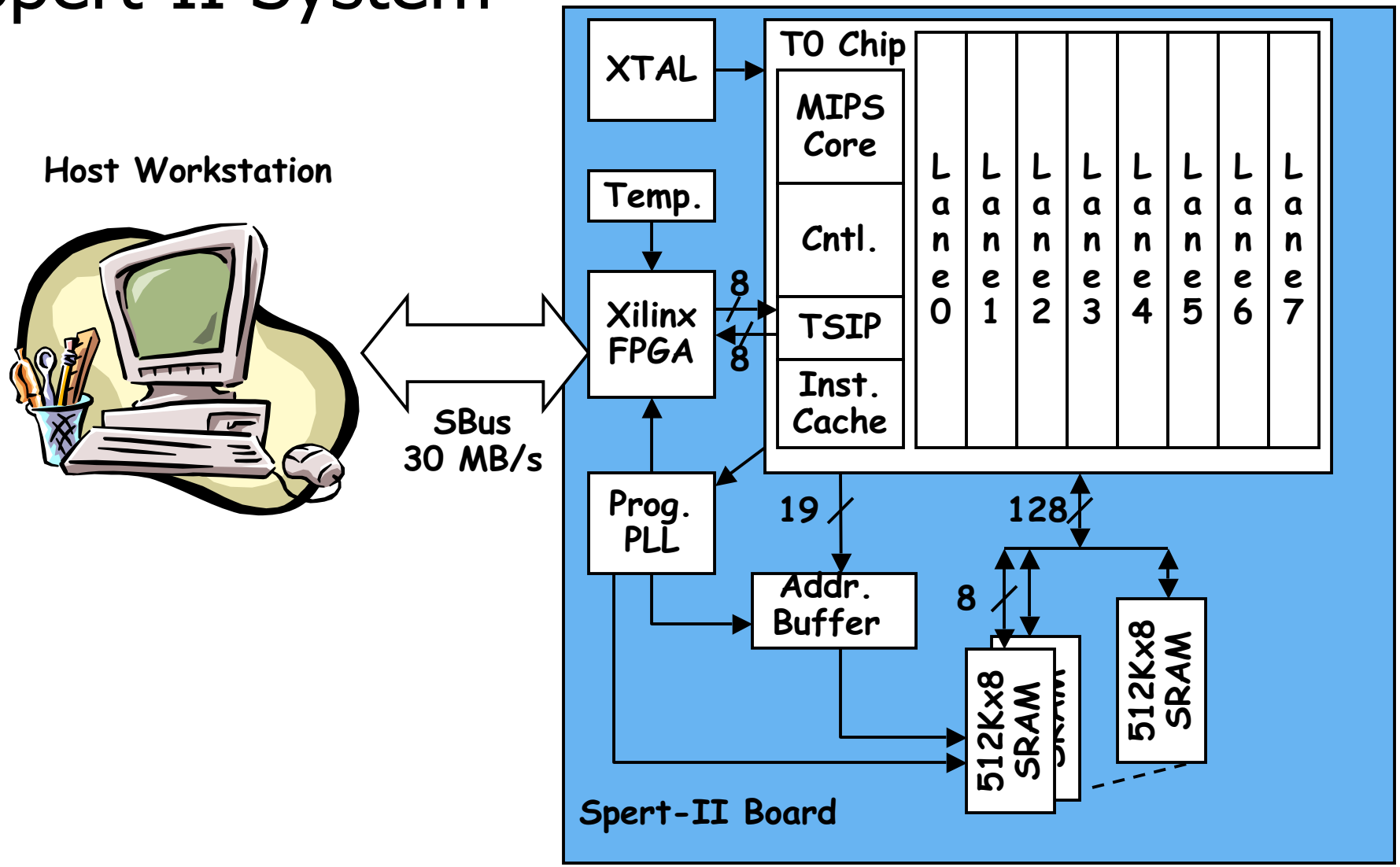
Vector Instruction Parallelism

Can overlap execution of multiple vector instructions



Complete 24 operations/cycle while issuing 1 short instruction/cycle

Spert-II System



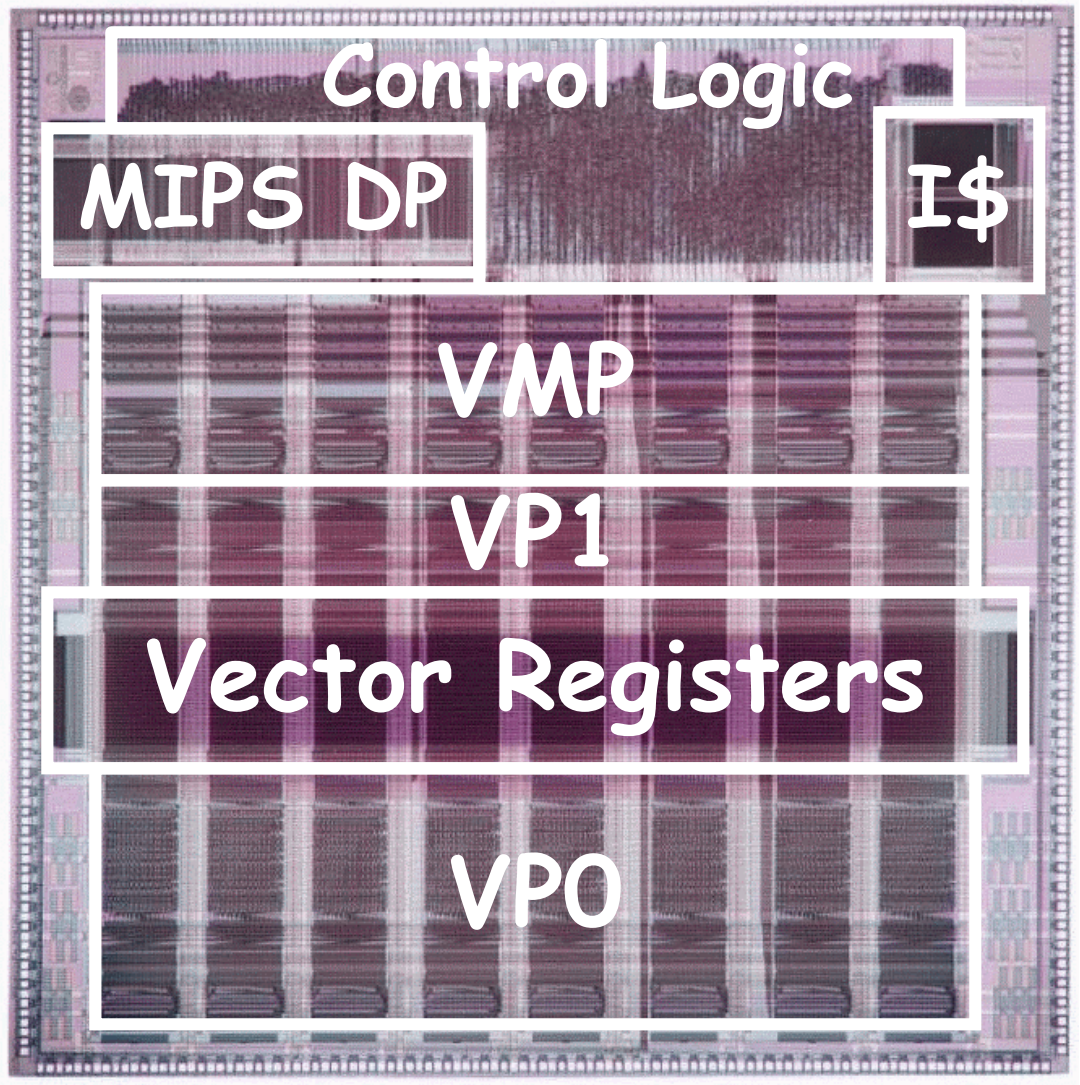
Start again…

- T0 design started in November 1992
- Design was exotic for a small team
  - Custom design (I.e.,many transistors drawn by hand)
  - Our own clocking scheme, pads, power and ground
  - Our own packaging technology
  - Double-pumped 8-port vector register files (Bertrand)
  - Had to resize datapath, redo all cells, three times…
- First prediction of tapeout was May 1993
  - Very wishful thinking…
- VLSI team banned management (Morgan, JohnW) from meetings
  - Asking "Are we there yet?" isn't particularly helpful

CAD Tools Suck!

- We resolved not to write our own CAD tools
- This meant we only spent 50% of our time writing/fixing CAD tools
- At end of project, we had everything except the automatically synthesized, placed and routed section complete
- Took another 3 months to get this to finish - each run would take one week
- Finally taped out on Valentine's Day 1995
  - (3 grad students, 2+ years)

T0 Die Breakdown



Switched to HP CMOS 26G process late in design

- used 1.0µm rules in 0.8µm process
- only used 2 out of 3 metal layers

16.75x16.75mm^2

730,701 transistors

4W typical @ 5V, 40MHz

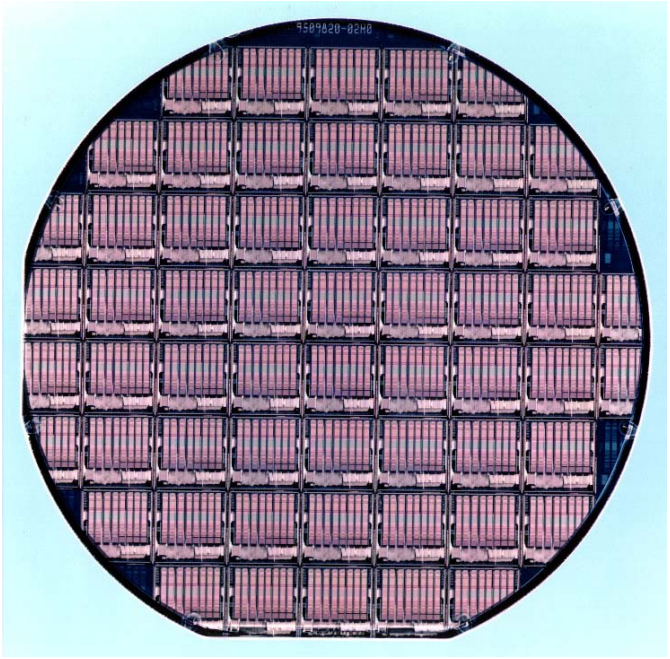
12W maximum

Performance:

320MMAC/s

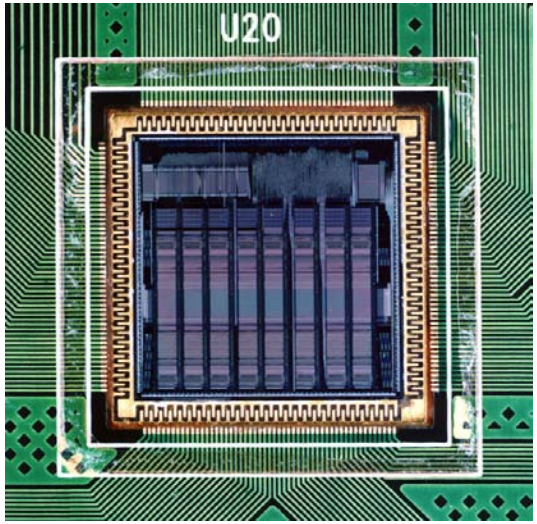
640MB/s

A Long Night at the Test Facility
(Thursday, April 13, 1995)



- After spending several hours not getting wafer tests to work, fixed a simple 1 cycle offset in reset signal
- 40% of chips passed all tests!
- Design was fully functional with no bugs

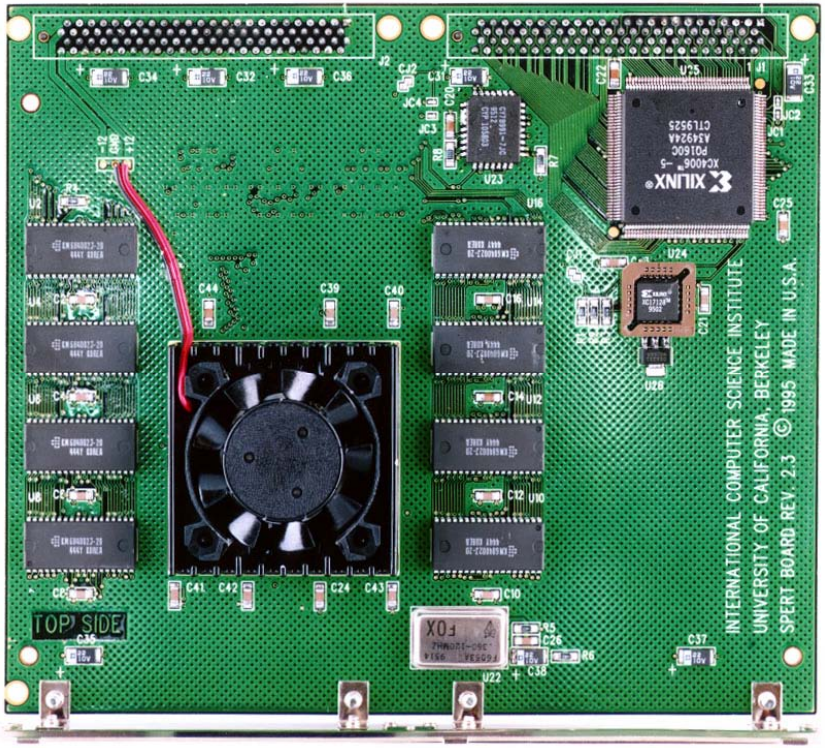
Packaging Adventures, or "Where's Hilda now?"



- To avoid cost of custom package for die, we attached the die directly to the circuit board!
- Chip-on-board used for wristwatches, not processors, previously

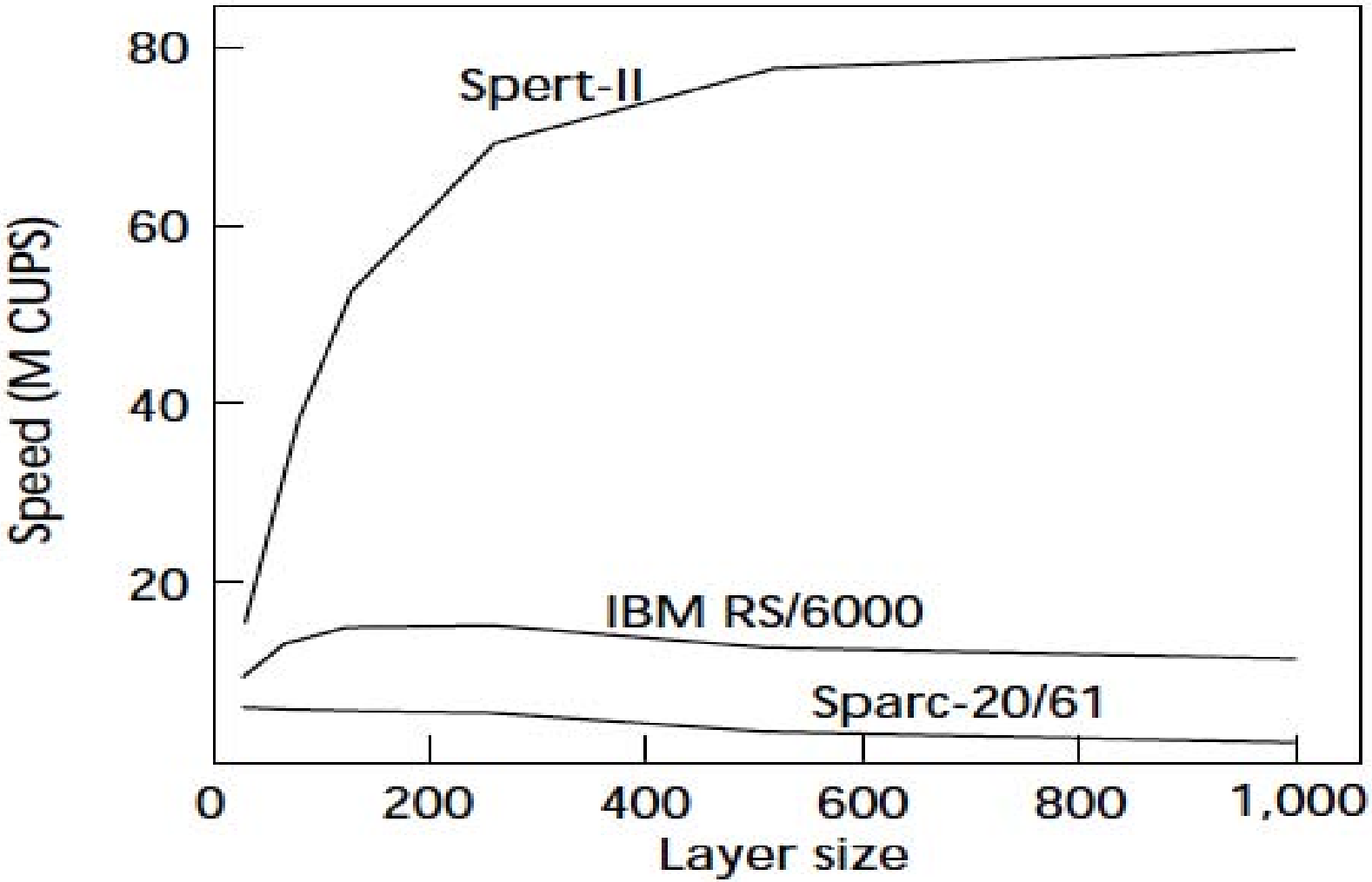
- Had to figure out fabrication recipe to make PCBs
  - Polyamide with low-flow prepeg
- Then get die bonded successfully
  - First 9 out of 10 boards worked fine
  - Next batch of 20 all failed (the only woman who knew how to do this well had left company - "Hilda")

SPERT-II Worked!



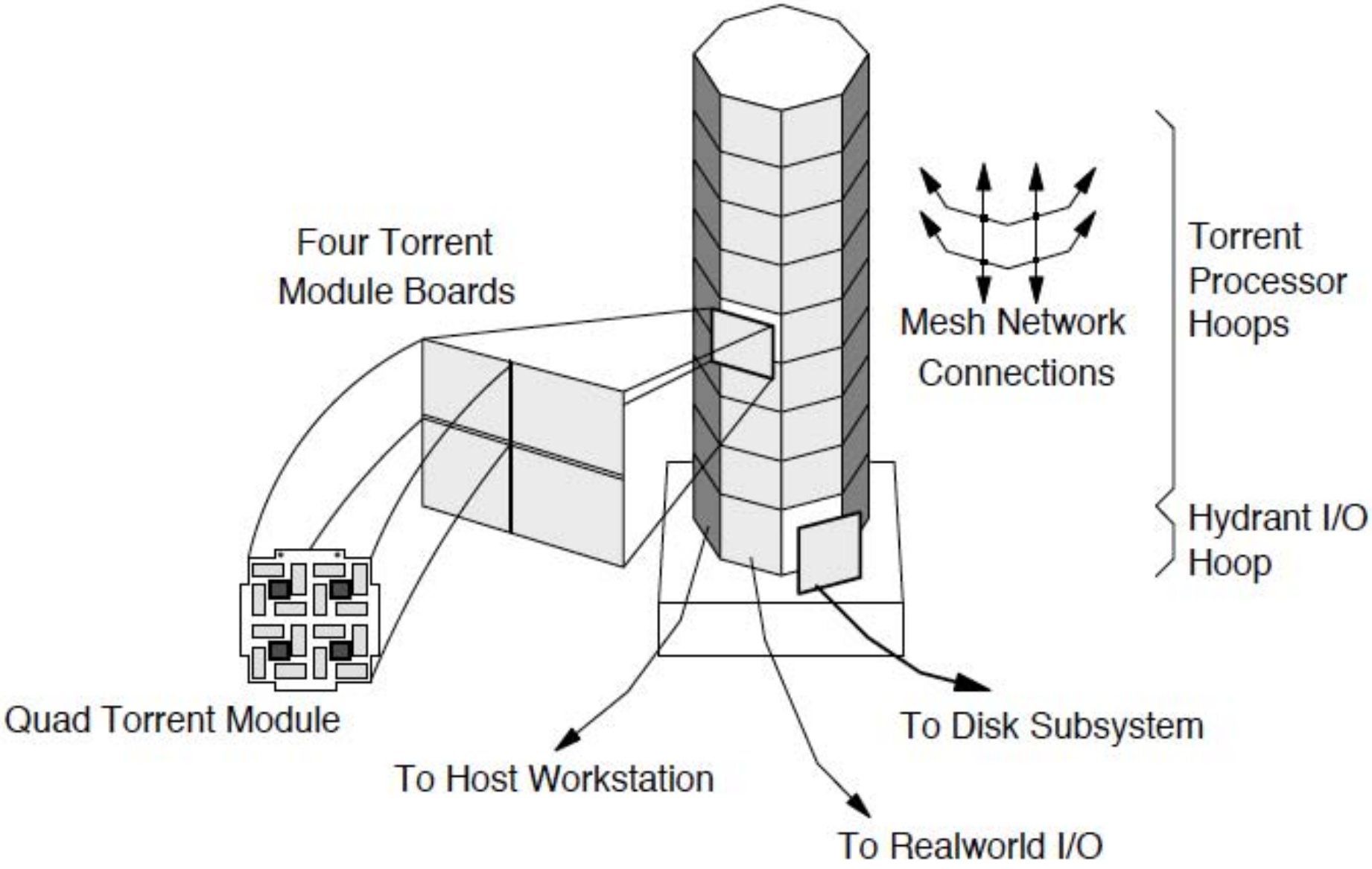
- 35 boards shipped to 9 international sites
- Success due to great board design (Jim Beck) and great software (David Johnson)

Spert-II Performance on Backpropagation

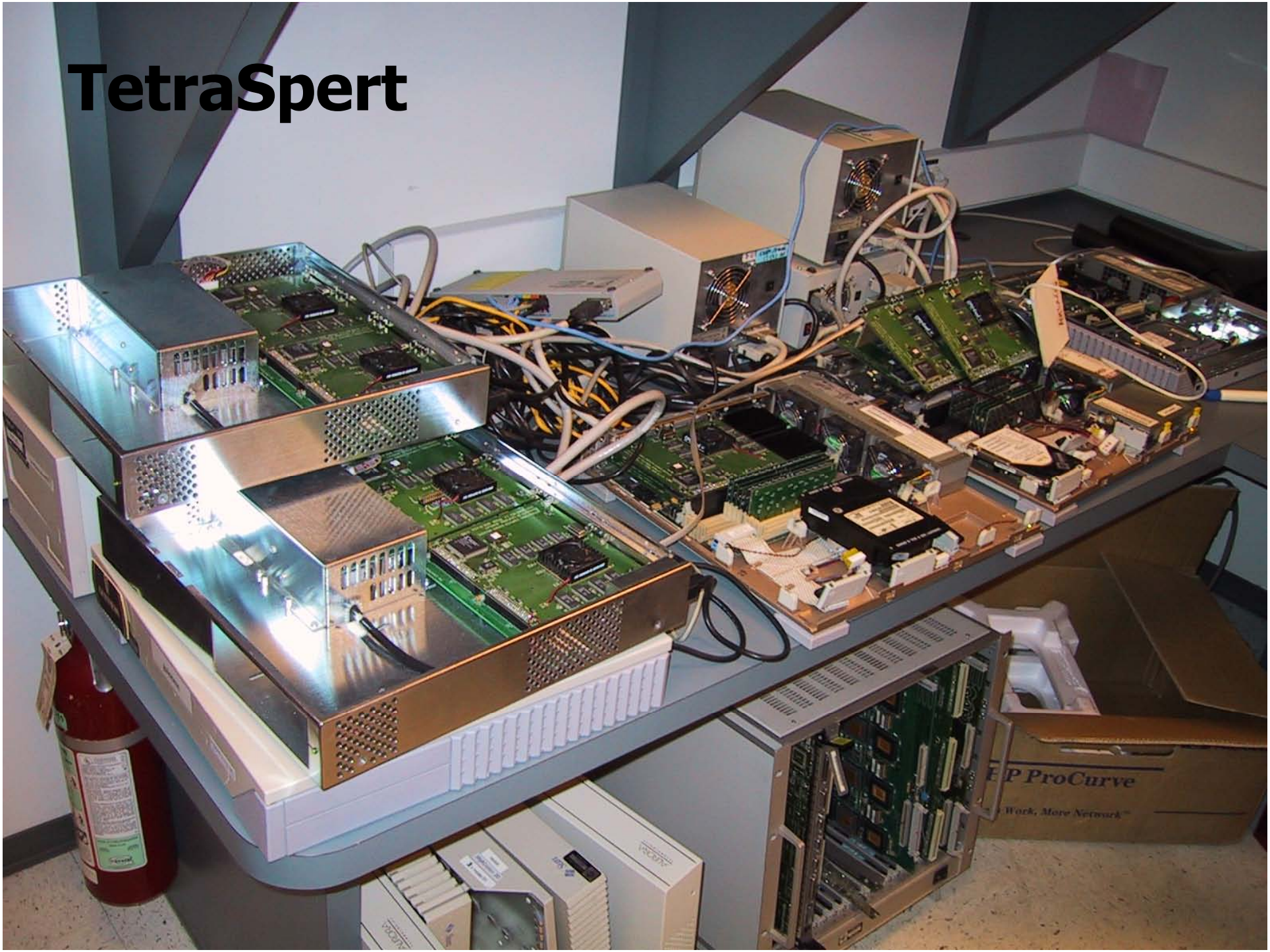


- Used as production research platform for seven years (last one powered down in 2002!)

What about CNS-1?



TetraSpert



TetraSpert: Compact Edition (Dan Ellis)

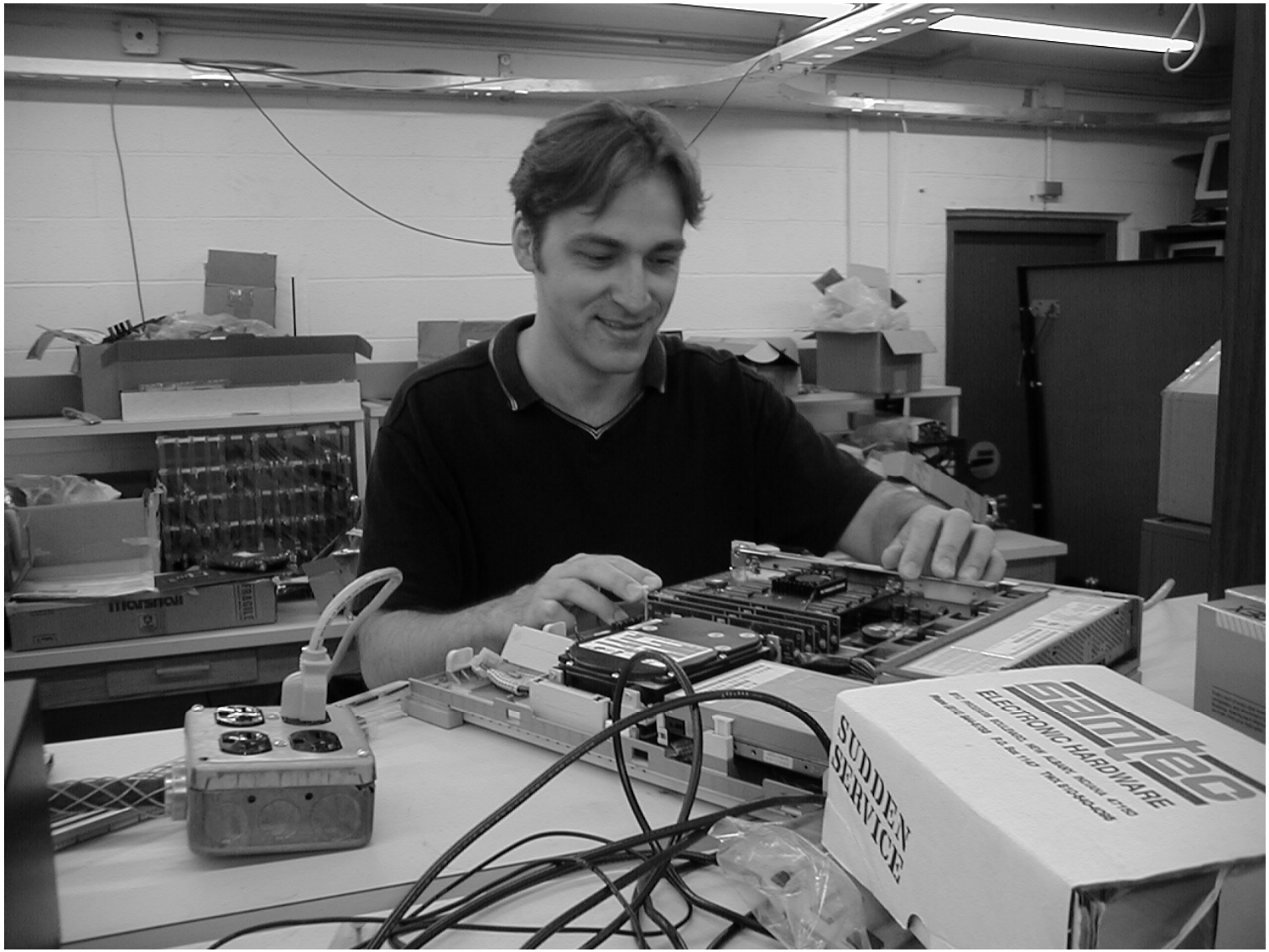
Participating Visitors

- Karlheinz Hafner
- Paul Mehring
- Silvia Mueller
- Heinz Schmidt
- Stephan Murer
- Thomas Schwair
- Arno Formella
- Paola Moretto
- Phillip Pfaerber

Some Project Spin-Offs

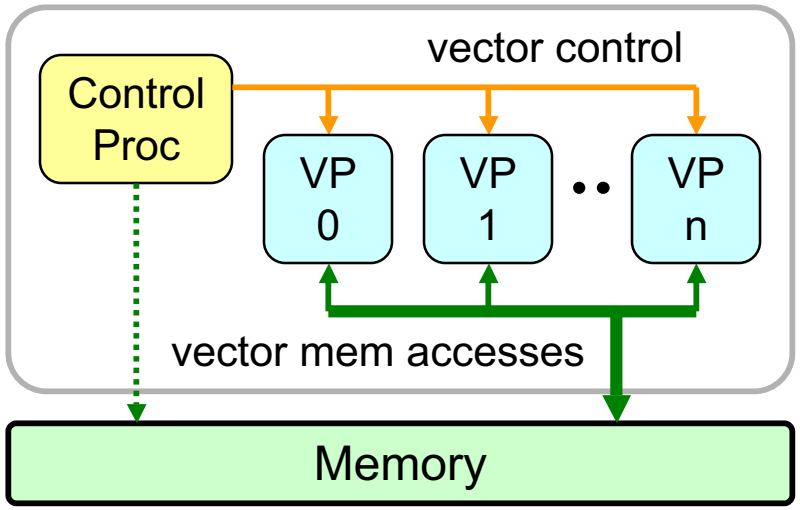
- Vector-IRAM project on UCB campus
  - Led by David Patterson, and grad student Christos Kozyrakis
- SoftFloat and TestFloat libraries
  - IEEE FP emulation libraries written by John Hauser, now widely used
- PHiPAC (Portable, High-Performance ANSI C)
  - High-performance libraries generated by machine (autotuning), with Jeff Bilmes and James Demmel
  - First autotuning effort, now a very popular field (FFTW, ATLAS, Spiral, OSKI)

A Brief Sojourn at MIT (9 years)



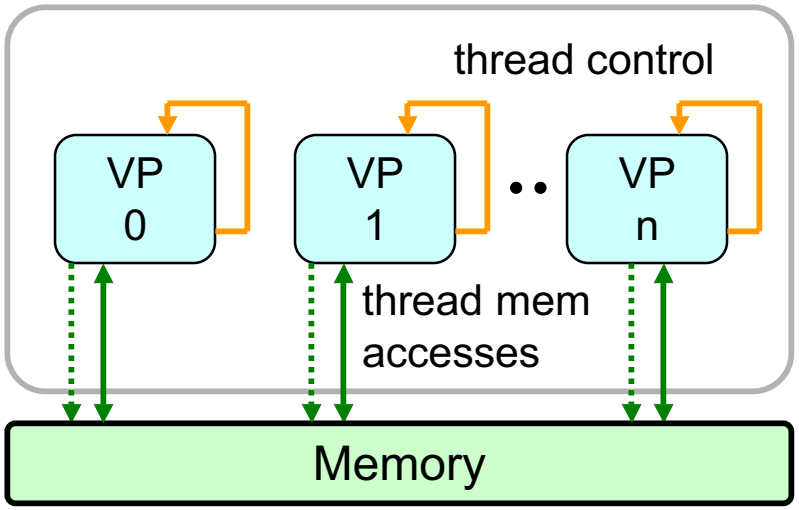
Vector and multithreaded architectures have very different strengths and weaknesses

Vector Architecture



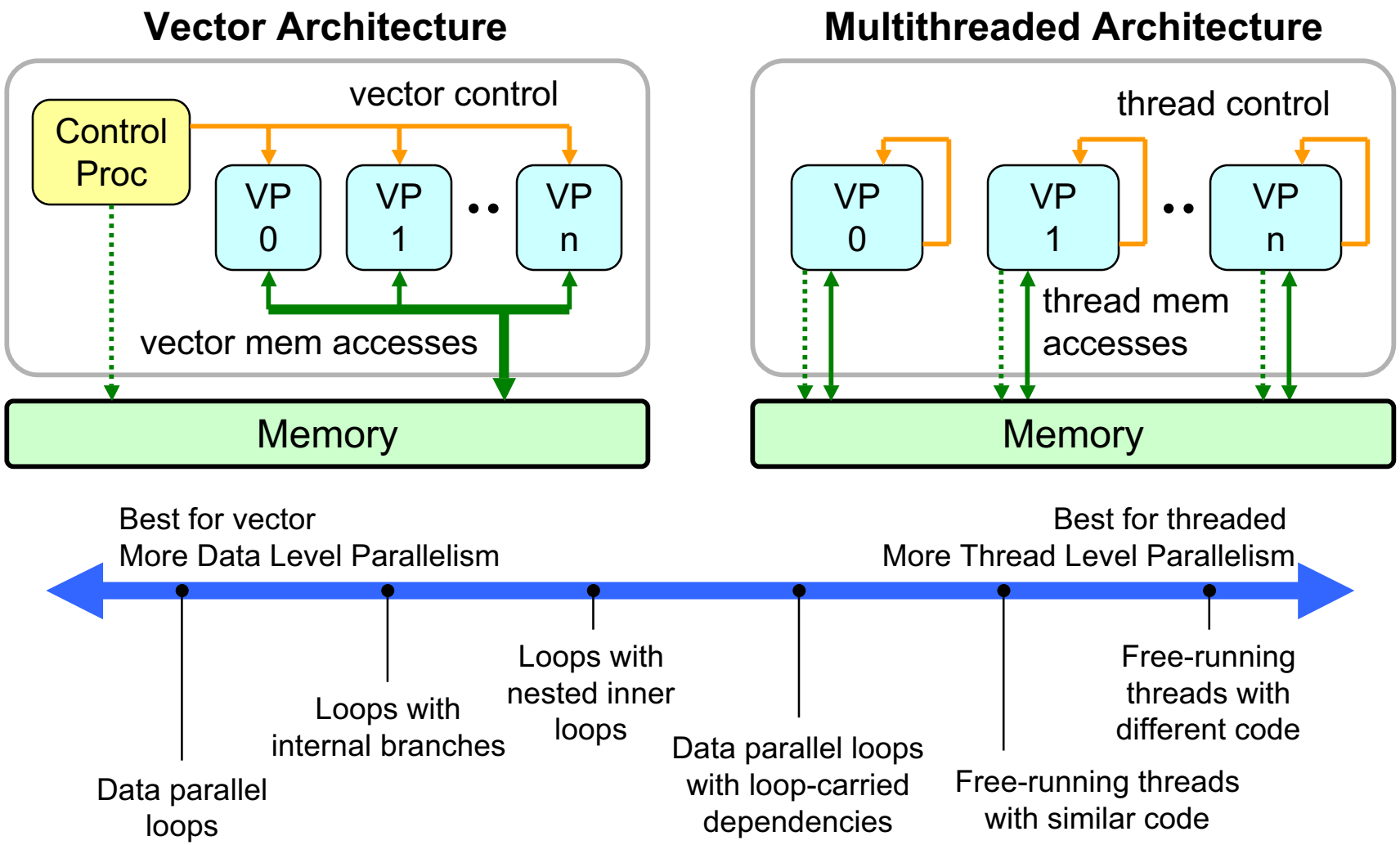
- ⊕ Amortize control and loop bookkeeping overhead
- ⊕ Exploit structured memory accesses across VPs
- ⊖ Unable to execute loops with loop-carried dependencies or complex internal control flow

Multithreaded Architecture

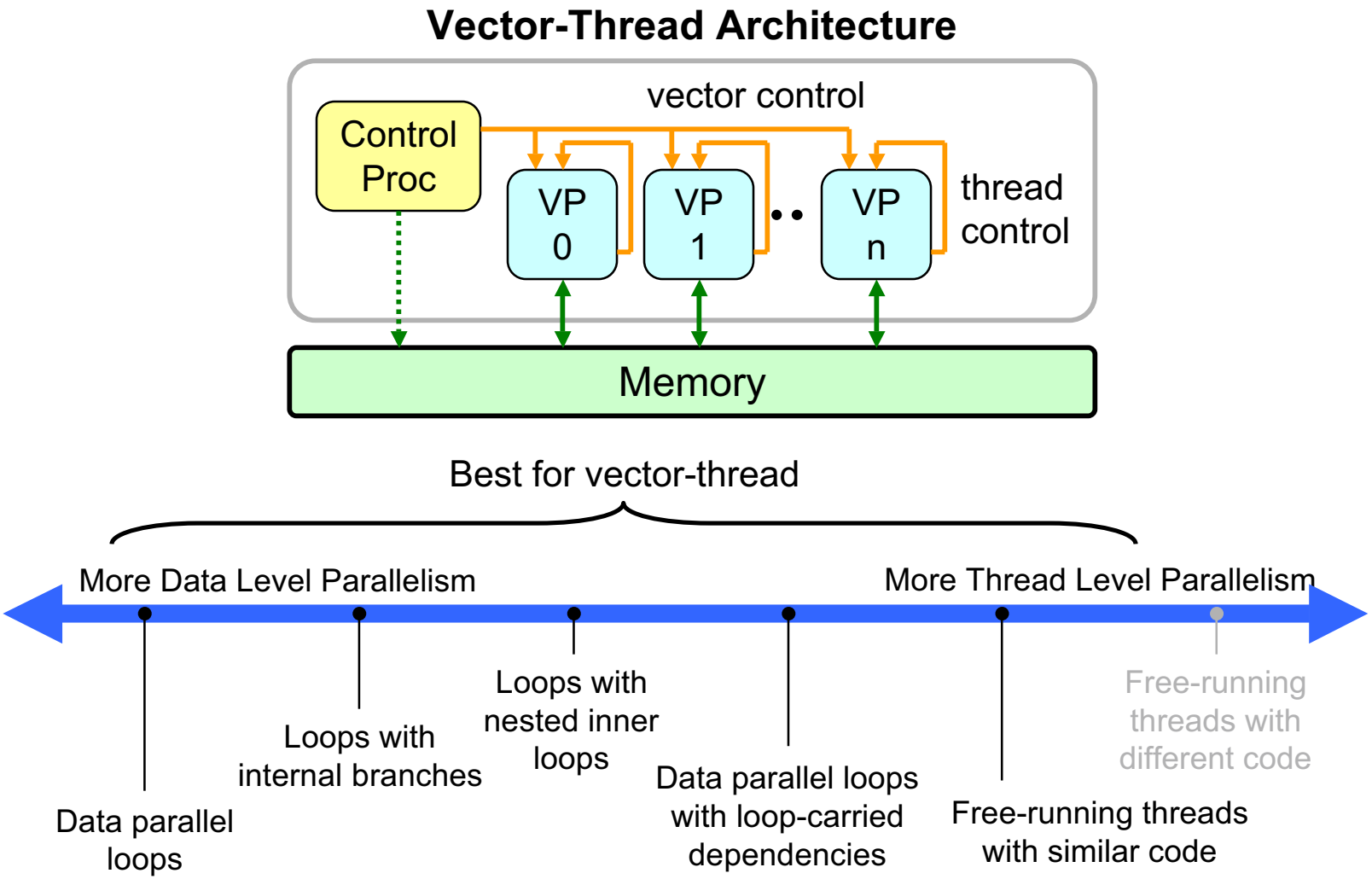


- ⊕ Very flexible model
- ⊖ Unable to amortize common control overhead
- ⊖ Unable to exploit structured memory accesses across threads
- ⊖ Costly memory-based synchronization and communication

Vector and multithreaded architectures have very different strengths and weaknesses



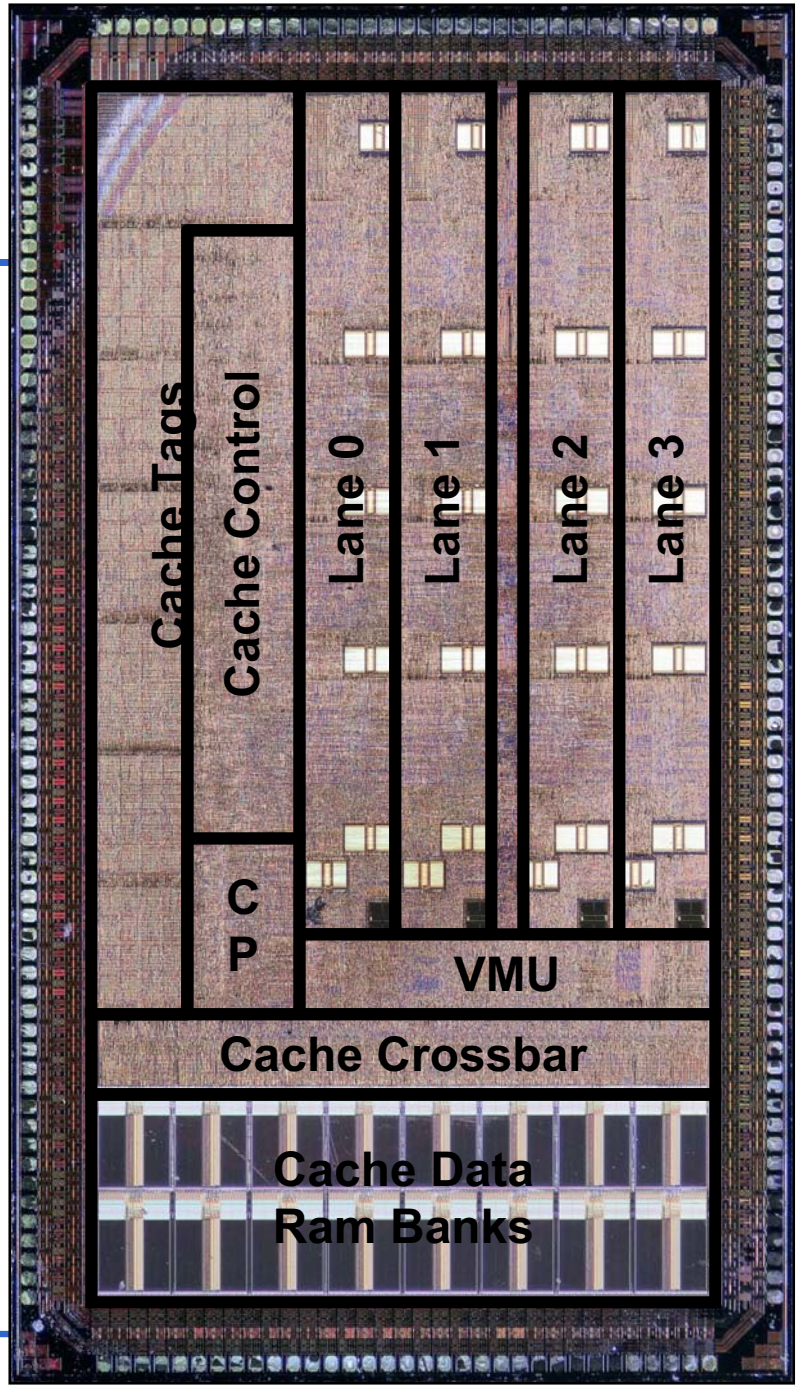
Vector-thread architectural paradigm unifies the vector and threaded compute models



The Scale VT Processor

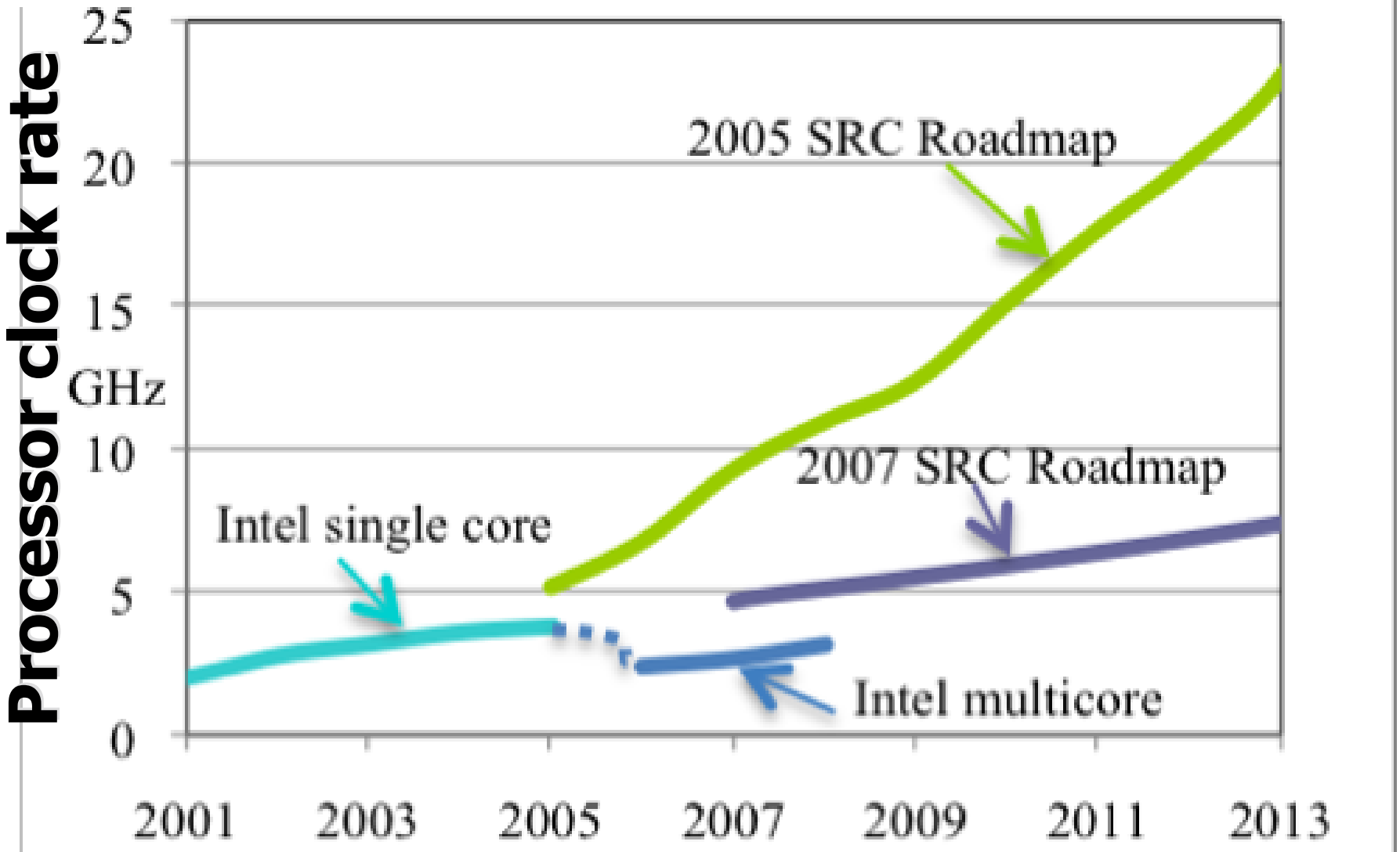
Ronny Krashinsky, Chris Batten

Process Technology	TSMC 0.18µm
Metal Layers	6 Aluminum
Transistors	7.14 Million
Gates	1.41 Million
Standard Cells	397,000
Flip-Flops + Latches	94,000
Core Dimensions	5.7 x 2.9 mm
Core Area	16.6 mm^2
Chip Area	23.1 mm^2
Design Time	19 months
Design Effort	24 person-months



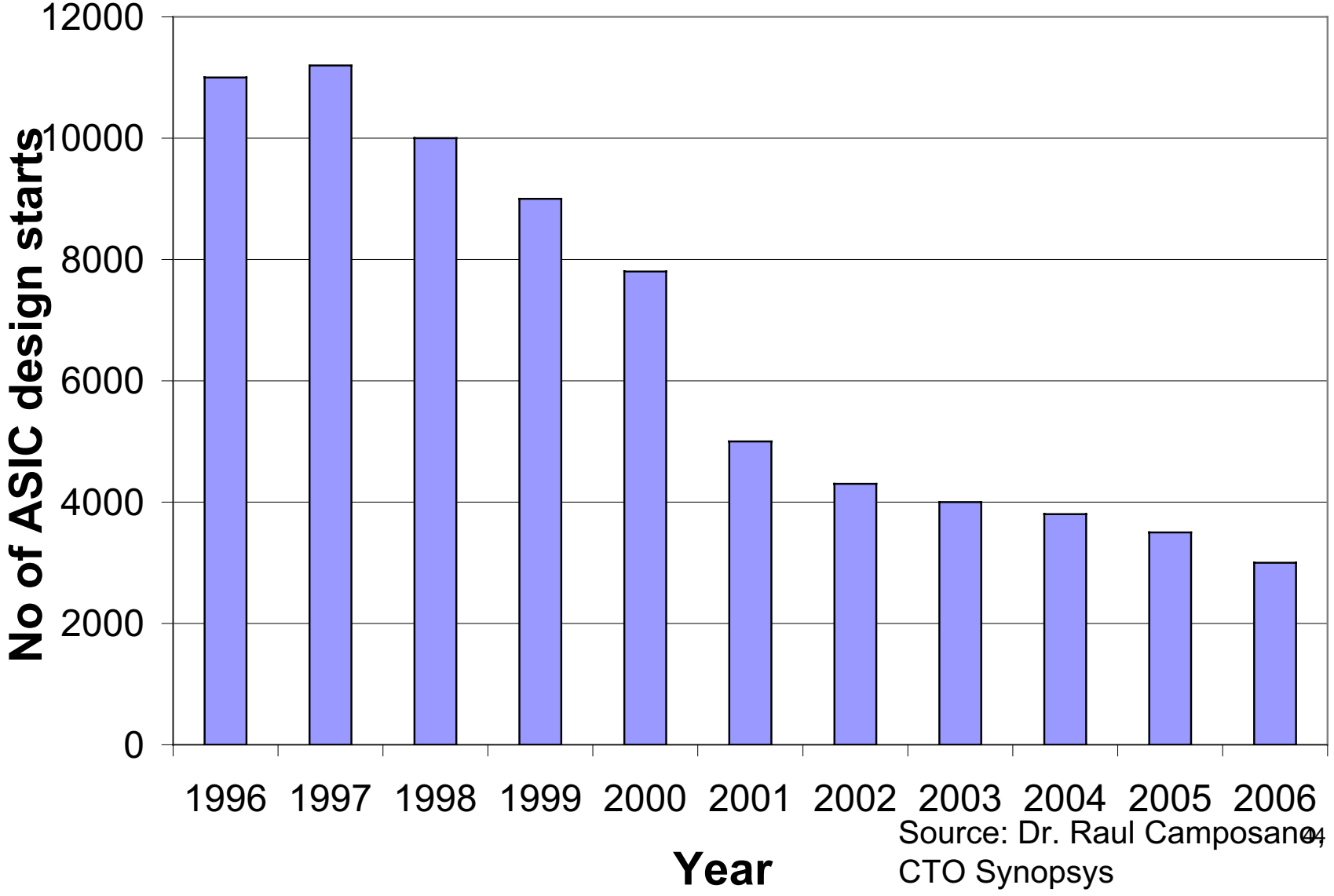
Winner, ISSCC/DAC Student Design Contest, 2007

The End of the Uniprocessor



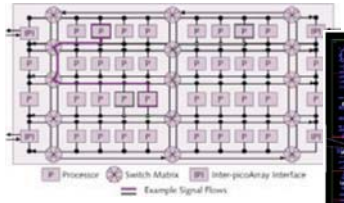
[From "The Parallel Computing Lab at UC Berkeley", UCB Techreport, 2008]

Increasing Cost of Design: Fewer Custom Chips

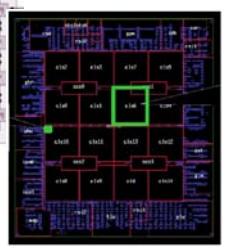


Source: Dr. Raul Camposano, CTO Synopsys

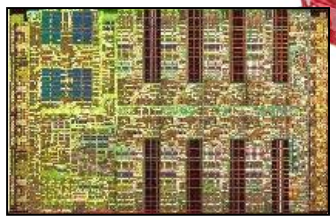
System designers across the board are using processor arrays to meet their design goals



Picochip DSP
1 Ctrl GPP
248 ASPs

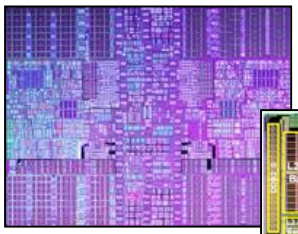


Cisco CSR-1
188 Tensilica ASPs

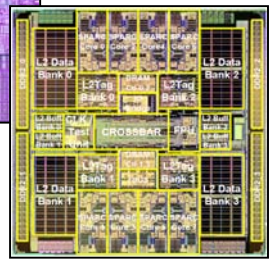


IBM Cell
1 Ctrl GPP (2 threads)
8 ASPs

ATI Unified Shader
GPU Architectures
48 ASPs



IBM Power6
2 GPPs



Sun Niagara
8 GPPs (32 threads)

(GPP: General-Purpose Processor, ASP: Application-Specific Processor)

A Parallel Revolution, Ready or Not

- Embedded: per product ASIC to programmable platforms
  ⇒ Multicore chip most competitive path
  - Amortize design costs + Reduce design risk + Flexible platforms
- PC, Server: Power Wall + Memory Wall = Brick Wall
  ⇒ End of the way we've scaled uniprocessors for last 40 years

⇒ New Moore's Law is 2X processors ("cores") per chip every technology generation, but same clock rate

- "This shift toward increasing parallelism is not a triumphant stride forward based on breakthroughs …; instead, this … **is actually a retreat from even greater challenges that thwart efficient silicon implementation of traditional solutions**."

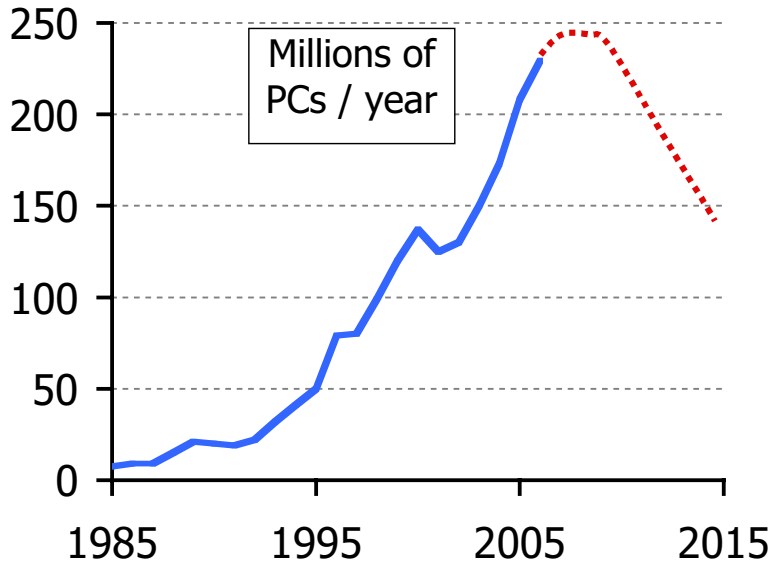
The Parallel Computing Landscape: A Berkeley View

- Sea change for HW & SW industries since changing the model of programming and debugging

P.S. Parallel Revolution May Fail!

- John Hennessy, President, Stanford University, 1/07:
  "…when we start talking about parallelism and ease of use of truly parallel computers, we're talking about a problem that's as hard as any that computer science has faced. …
  I would be panicked if I were in industry."

  "A Conversation with Hennessy & Patterson," *ACM Queue Magazine*, 4:10, 1/07.
- 100% failure rate of Parallel Computer Companies
  - Convex, Encore, MasPar, NCUBE, Kendall Square Research, Sequent, (Silicon Graphics), Transputer, Thinking Machines, …
- What if IT goes from a *growth* industry to a *replacement* industry?
  - If SW can't effectively use 8, 16, 32, ... cores per chip
    ⇒ SW no faster on new computer
    ⇒ Only buy if computer wears out

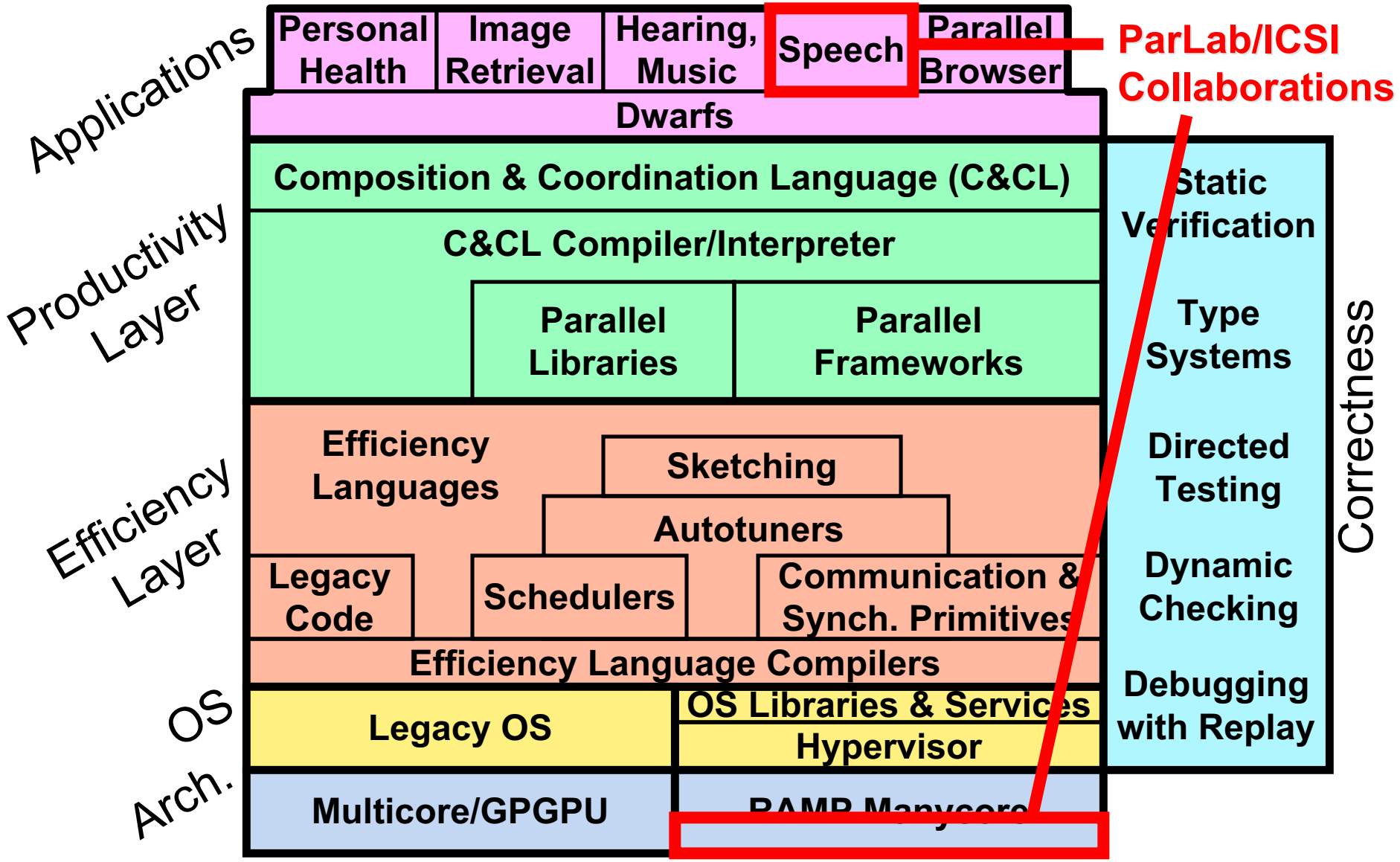


Berkeley View to Par Lab

- Berkeley researchers from many backgrounds meeting since Feb. 2005 to discuss parallelism
  - Krste Asanovic, Ras Bodik, Jim Demmel, Kurt Keutzer, John Kubiatowicz, Edward Lee, Nelson Morgan, George Necula, Dave Patterson, Koushik Sen, John Shalf, John Wawrzynek, Kathy Yelick, …
  - Circuit design, computer architecture, massively parallel computing, computer-aided design, embedded hardware and software, programming languages, compilers, scientific programming, and numerical analysis
- Tried to learn from successes in high performance computing (LBNL) and parallel embedded (BWRC)
- Led to "Berkeley View" Tech. Report and new Parallel Computing Laboratory ("Par Lab")
- Goal: Productive, Efficient, Correct Programming of 100+ cores & scale as double cores every 2 years (!)

Par Lab Research Overview

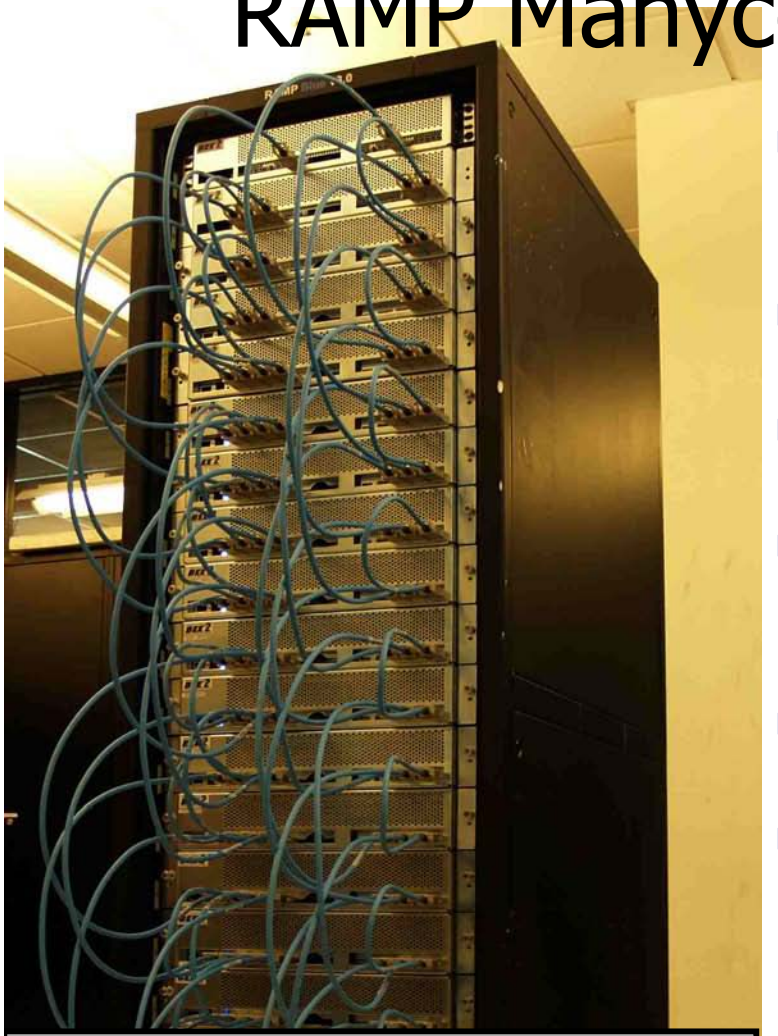
Easy to write correct programs that run efficiently on manycore



Flashback: CNS-1 Software Stack

- Experimenter — machine details are totally hidden
  - Non-programming experiments, some current speech work
  - Connectionist simulators (CNSsim)
- Programmer - (Ph.D. Student)
  - Libraries
    - Distributed memory objects (e.g. matrix and vector)
    - Message passing, synchronization, I/O
    - Simple scheduler, remote function call
  - Compilers - originally serial C++
  - Other tools (debugger, profiler, emulator, etc.)
- Wizard - knows everything about CNS-1
  - Low level libraries
  - Assembler and C++
  - Hardware simulator
  - Diagnostic network

RAMP Manycore Prototype



RAMP Blue, July 2007
- 1000+ RISC cores @90MHz
- Works! Runs UPC version of NAS parallel benchmarks.

- Multi-university RAMP project building FPGA emulation infrastructure
  - BEE3 boards with Chuck Thacker/Microsoft
- Expect to fit hundreds of 64-bit cores with full instrumentation in one rack
- Run at ~100MHz, fast enough for application software development
- Flexible cycle-accurate timing models
  - What if DRAM latency 100 cycles? 200? 1000?
  - What if barrier takes 5 cycles? 20? 50?
- "Tapeout" every day, to incorporate feedback from application and software layers
- Rapidly distribute hardware ideas to larger community

Ultra-Efficient Exascale Scientific Computing

Lenny Oliker, John Shalf, Michael Wehner

And many other folks at LBL and UC Berkeley

1km-Scale Global Climate Model Requirements

Simulate climate 1000x faster than real time

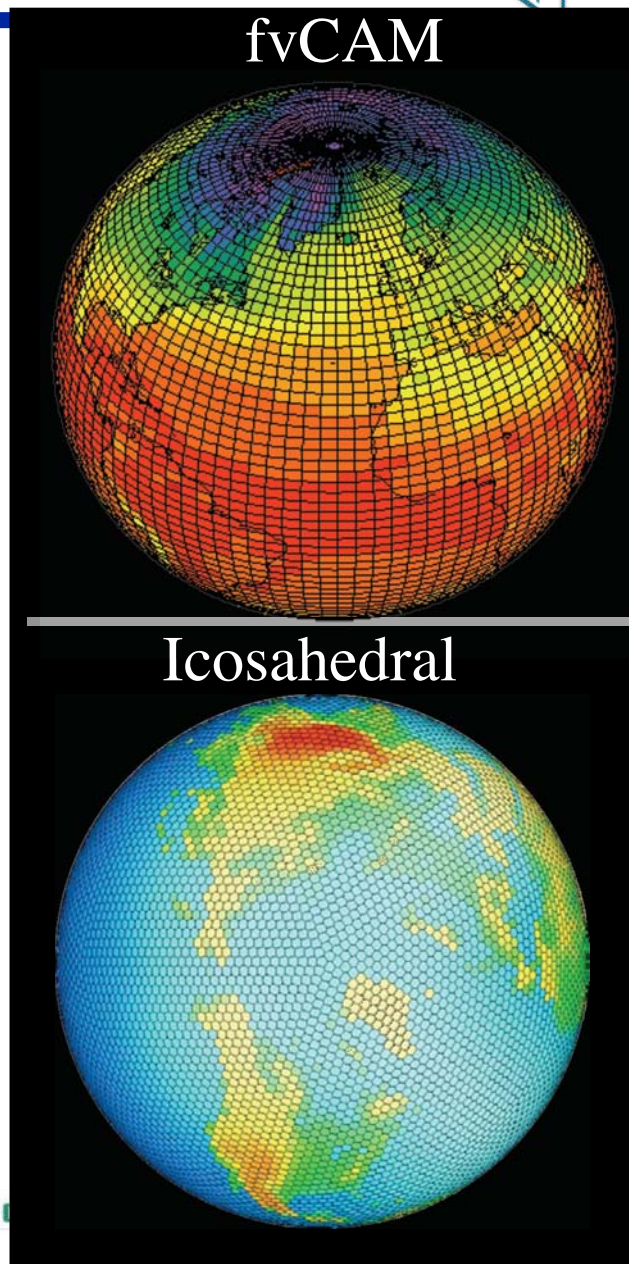
10 Petaflops sustained per simulation (~200 Pflops peak)

10-100 simulations (~20 Exaflops peak)

Truly exascale!

Some specs:

- **Advanced dynamics algorithms: icosahedral, cubed sphere, reduced mesh, etc.**
- **~20 billion cells → Massive parallelism**
- **100 Terabytes of Memory**
- **Can be decomposed into ~20 million total subdomains**

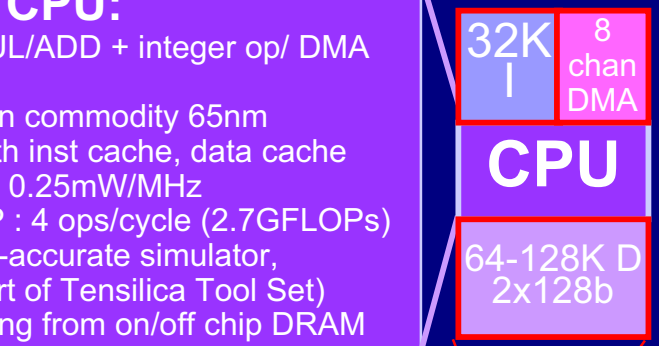


Climate System Design Concept

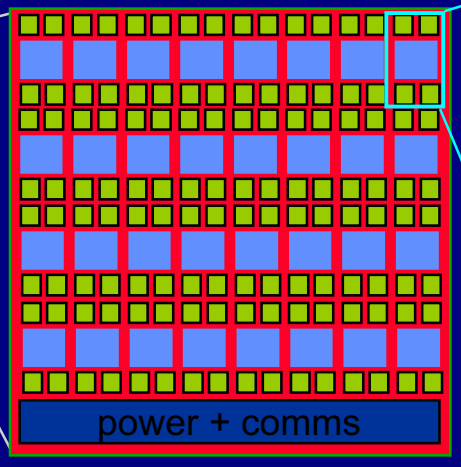
Strawman Design Study

VLIW CPU:

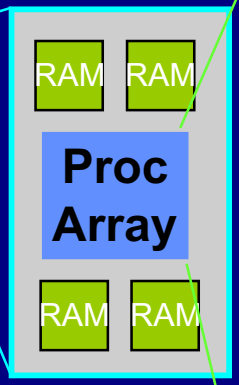
- 128b load-store + 2 DP MUL/ADD + integer op/ DMA per cycle:
- Synthesizable at 650MHz in commodity 65nm
- 1mm² core, 1.8-2.8mm² with inst cache, data cache data RAM, DMA interface, 0.25mW/MHz
- Double precision SIMD FP : 4 ops/cycle (2.7GFLOPs)
- Vectorizing compiler, cycle-accurate simulator, debugger GUI (Existing part of Tensilica Tool Set)
- 8 channel DMA for streaming from on/off chip DRAM
- Nearest neighbor 2D communications grid



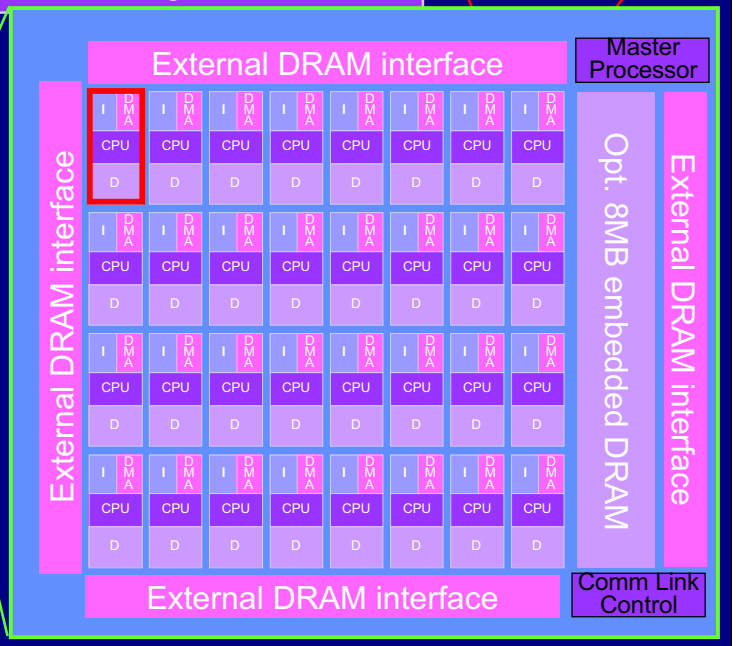
100 racks @ ~25KW



32 chip + memory clusters per board (2.7 TFLOPS @ 700W



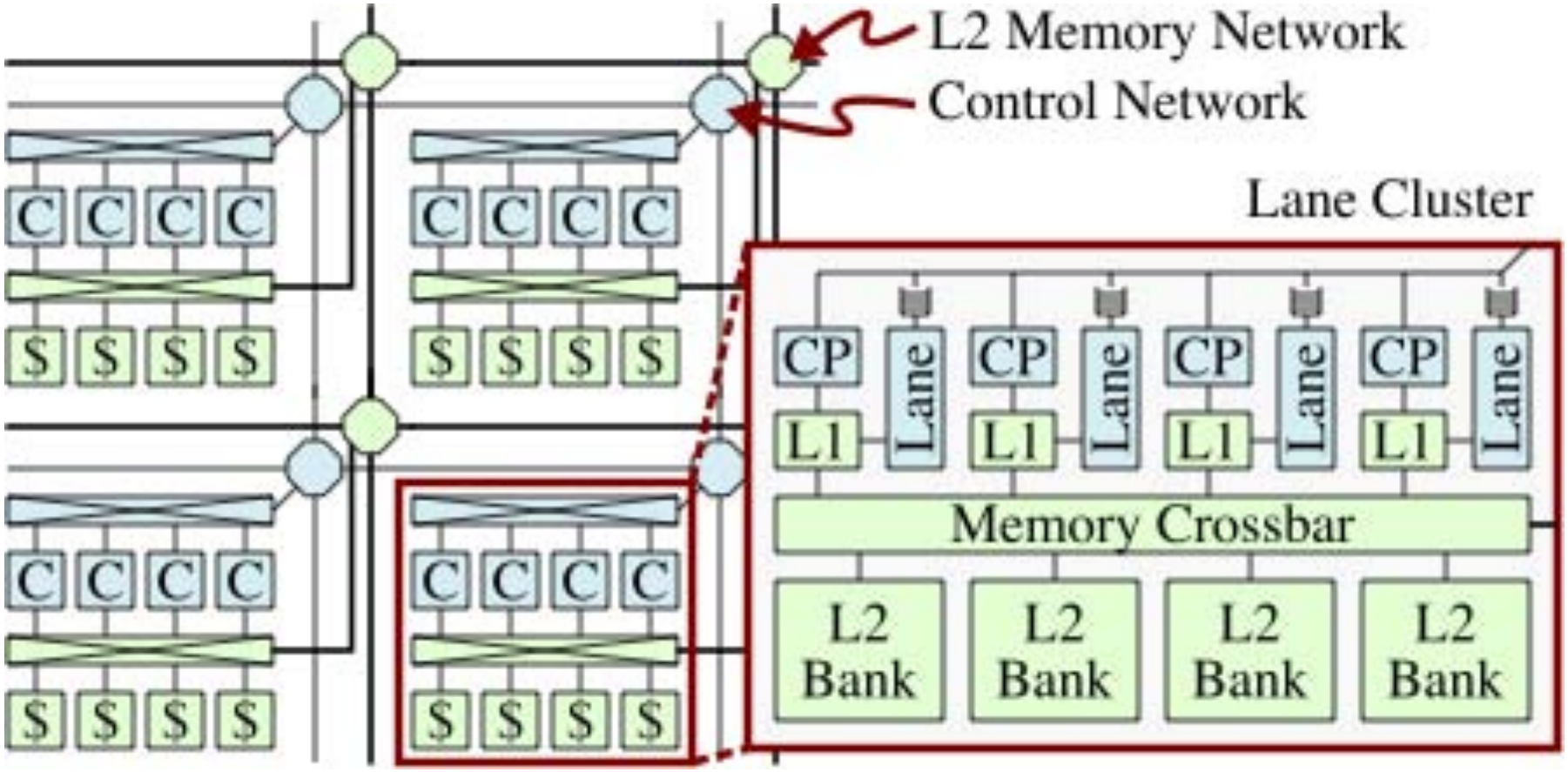
8 DRAM per processor chip: ~50 GB/s



32 processors per 65nm chip
83 GFLOPS @ 7W

Maven: Malleable Vector-Thread Engines

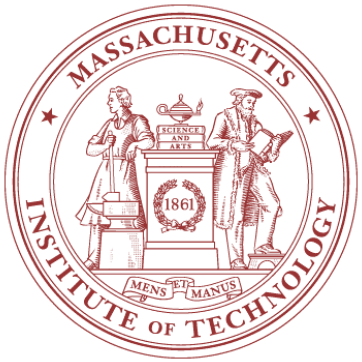
Christopher Batten, Yunsup Lee



Integrated photonic networks

Vladimir Stojanović, Judy Hoyt, Rajeev Ram,
Franz Kaertner, Henry Smith and Erich Ippen

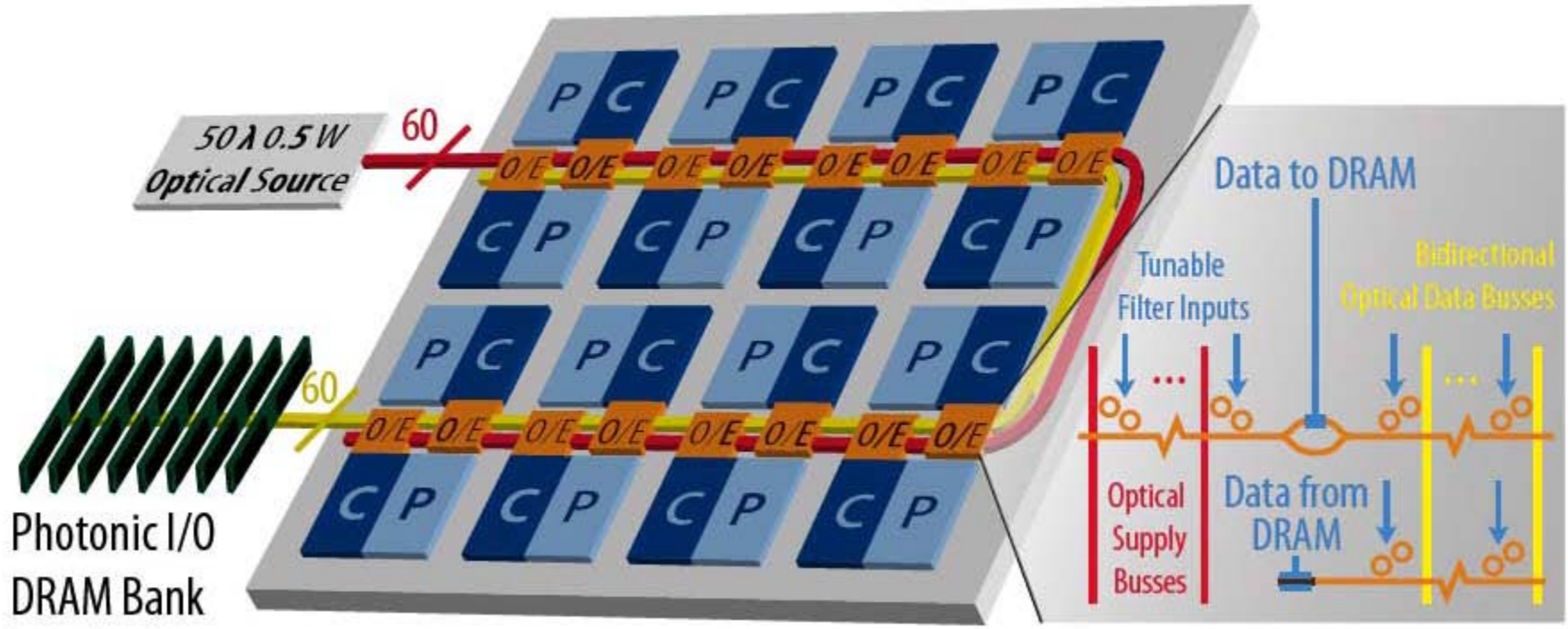
Krste Asanović



Massachusetts Institute
of Technology

University of California
at Berkeley/ICSI

Integrated photonic on/off-chip processor-memory interconnect



- Tile-to-off-chip-DRAM with multiple-access photonic network
  - Network has to resolve multiple access problem
    - Many cores to same DRAM bank (wavelength channel)
- Remove L2 cache (hit rate only 50%)
  - Add more cores
- On-chip and off-chip networks are aggregated into one
- Initial results indicate 20x improvement in bandwidth and energy consumption

Thanks for a great 20 years, here's to twenty more