# Low-Power Architecture

Bill Dally
Stanford University
Stream Processors, Inc.

# Processors Today are Inefficient

| Processor (90nm) | Energy/Operation (pJ) (32-bit Op) |
|---|---|
| ASIC | 2 |
| DSP | 60 |
| RISC | 200 |
| PC/Workstation | 2,000 |

# Where Does the Energy Go?

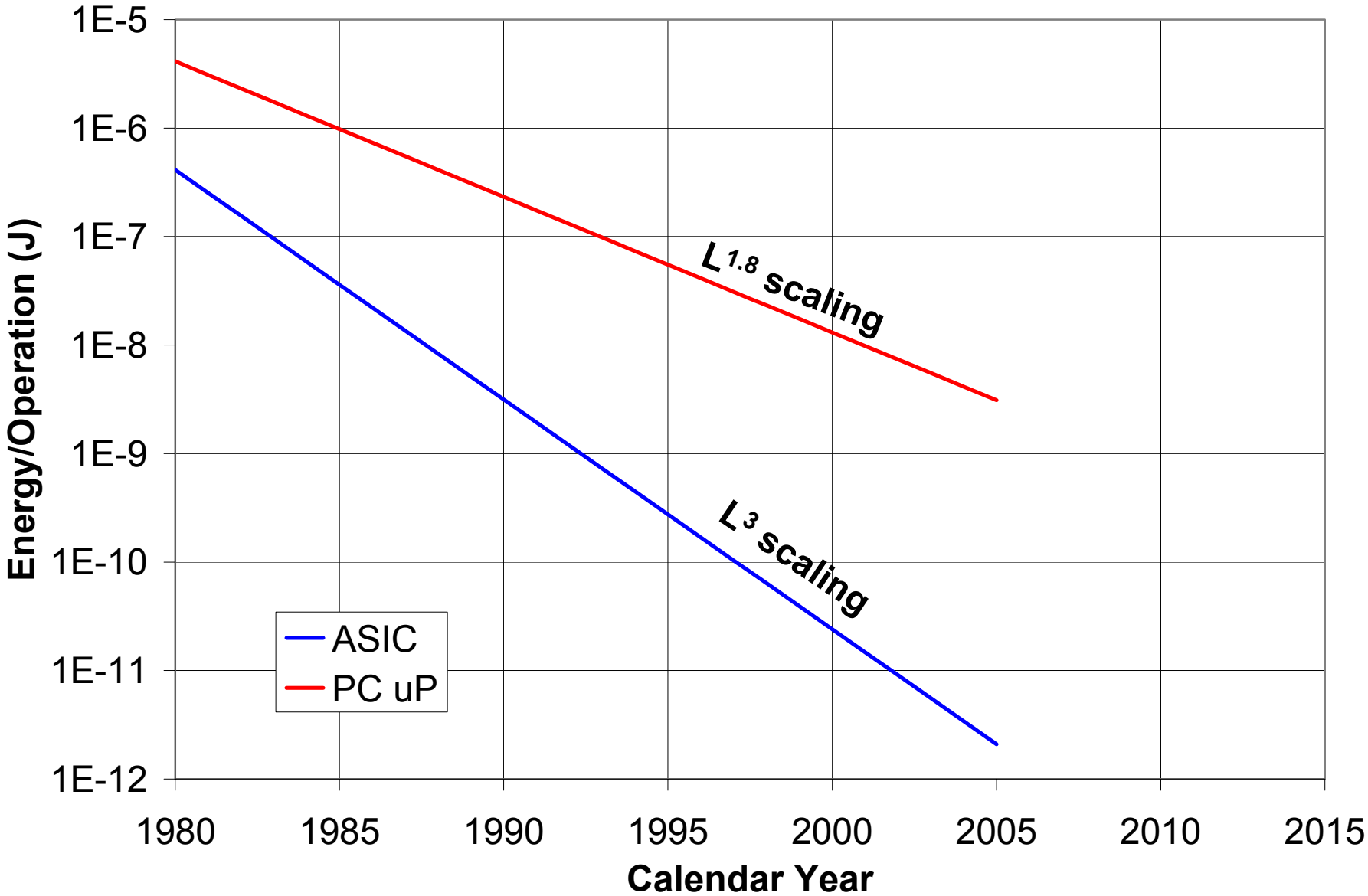
- Driving long wires
- Accessing arrays
- Control overhead
- Off-chip data movement

# Energy Dominated by Bandwidth

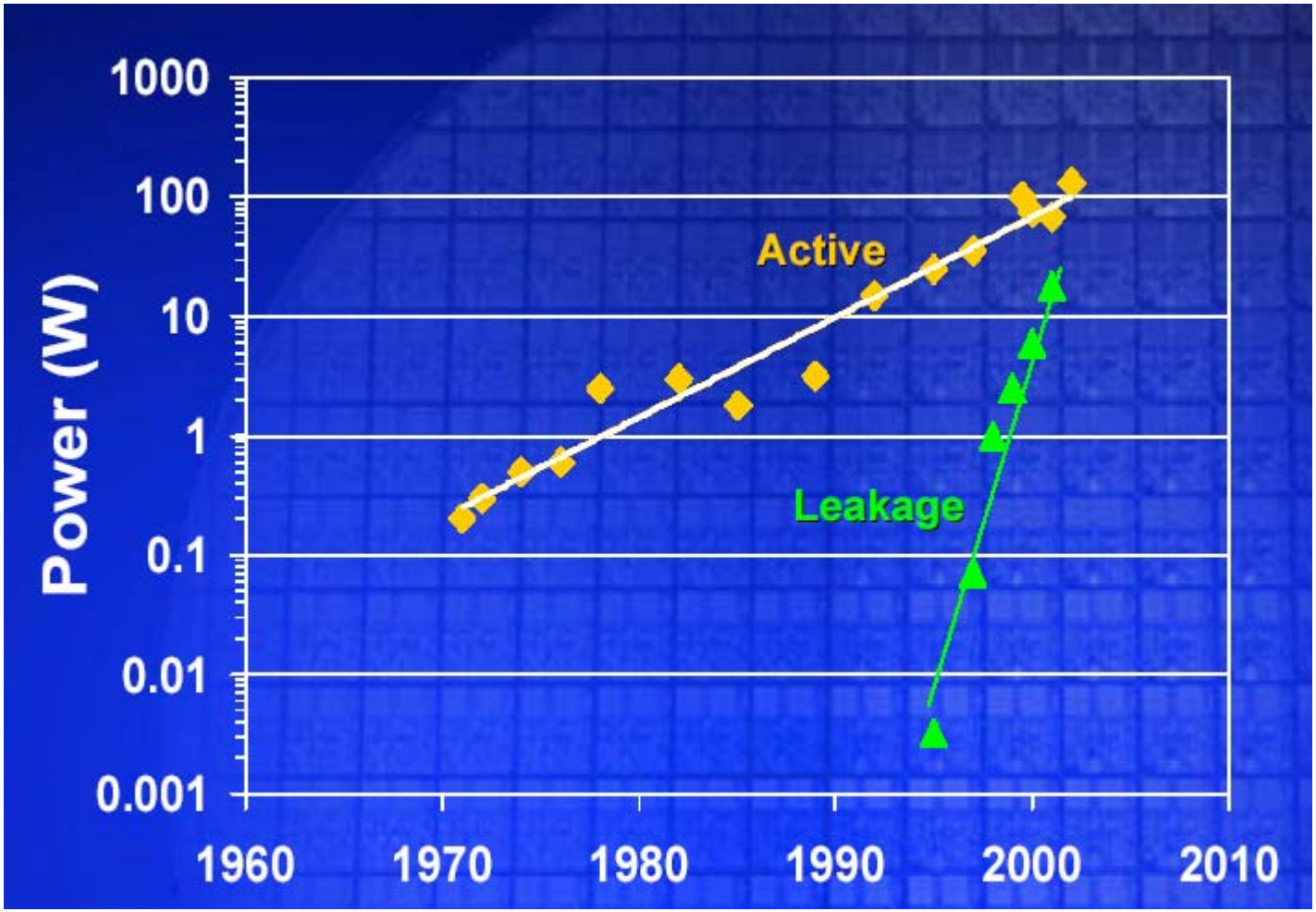
| **Operation** | **Energy** (0.13um) | (0.05um) |
|---|---|---|
| 32b ALU Operation | 5pJ | 0.3pJ |
| 32b Register Read | 10pJ | 0.6pJ |
| Read 32b from 8KB RAM | 50pJ | 3pJ |
| Transfer 32b across chip (10mm) | 100pJ | 17pJ |
| Transfer 32b off chip (CML) | 1.3nJ | 400pJ |
| Transfer 32b off chip (HSTL) | 1.9nJ | 1.9nJ |

Dally, HPCA, 2002

# The Gap is Widening

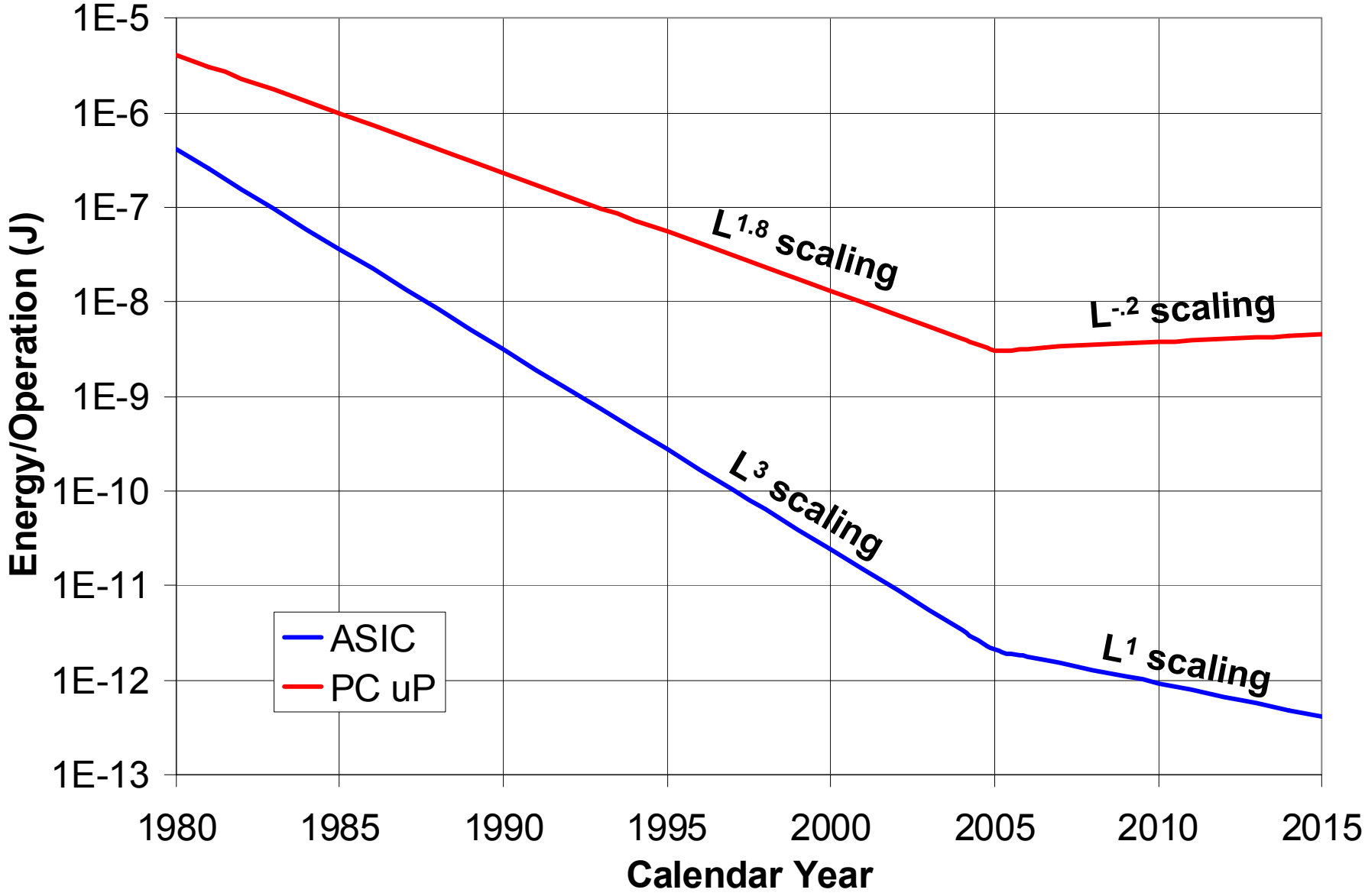


# An Inflection Point Approaches




Gordon Moore, ISSCC 2003

# The Effect of Leakage

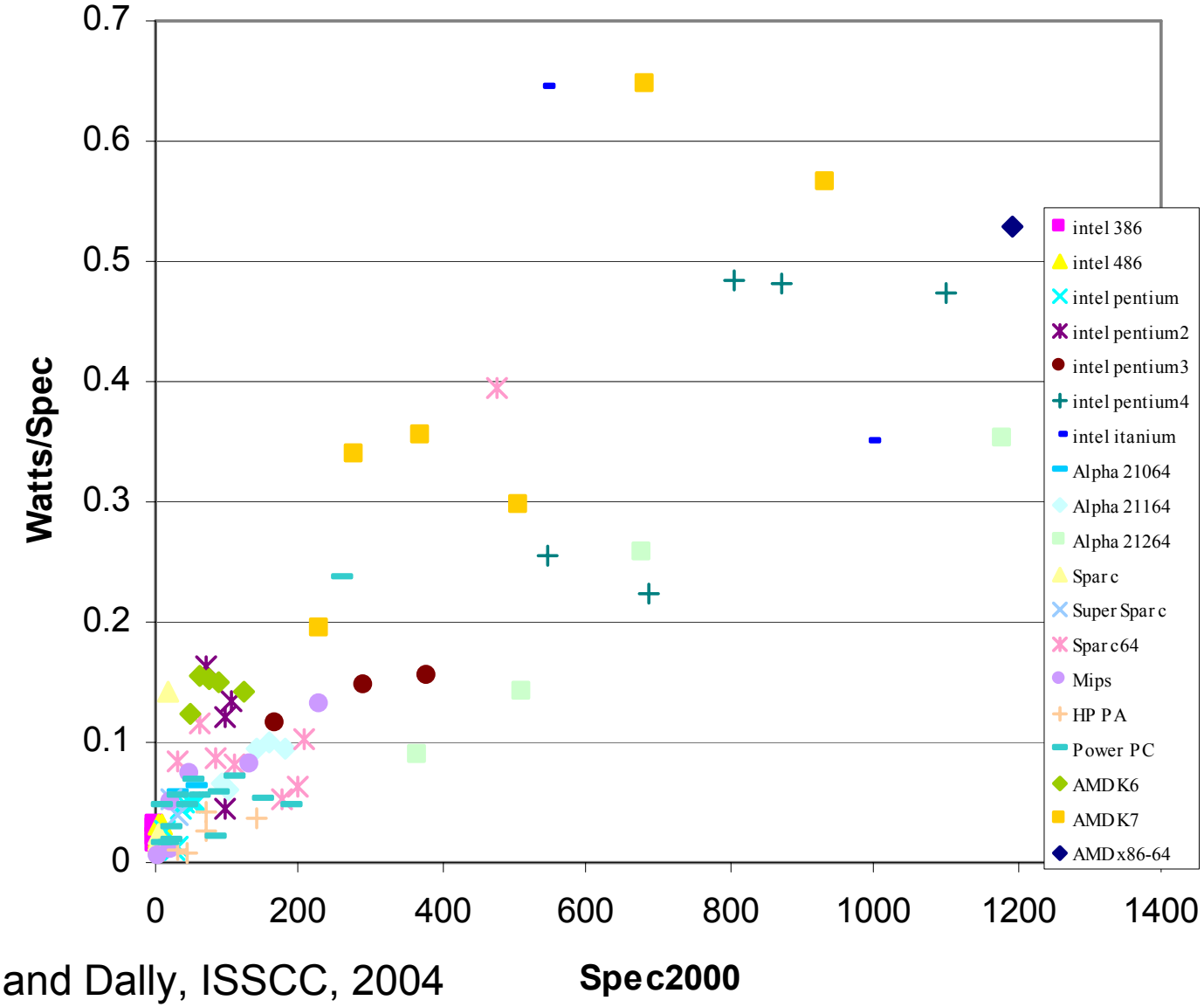


# An Impending Crisis

- ASIC/Processor efficiency gap widening
- Leakage will cause energy/op of high-end single-thread uPs to increase
- ASIC NRE costs increasing

# We need to close the ASIC/Processor Gap

# Solution 1: Parallelism

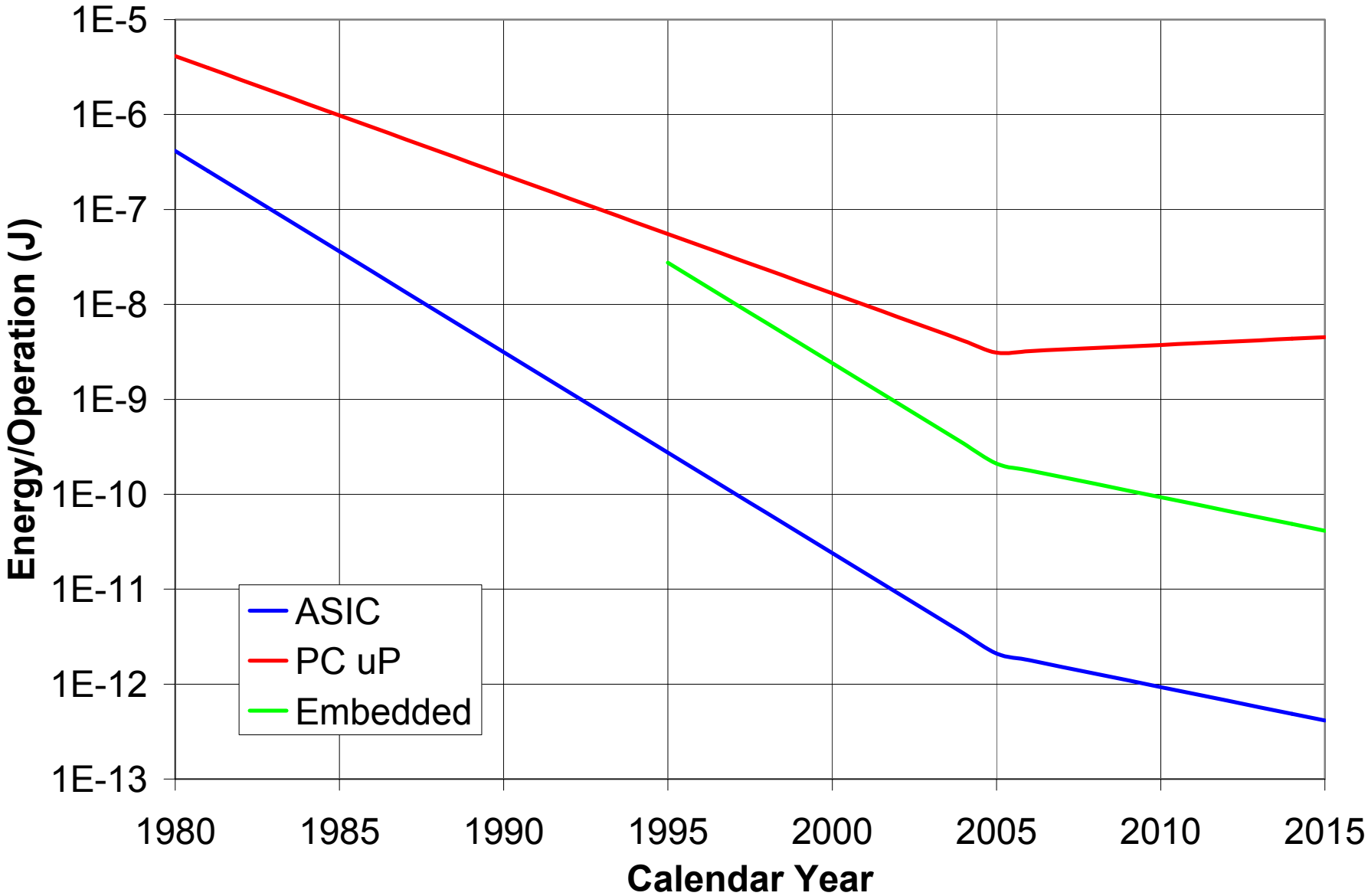



Horowitz and Dally, ISSCC, 2004

# Past point of diminishing returns for single-thread processors

- Multiple, simple processors more efficient than a single fast processor
- Requires parallel software

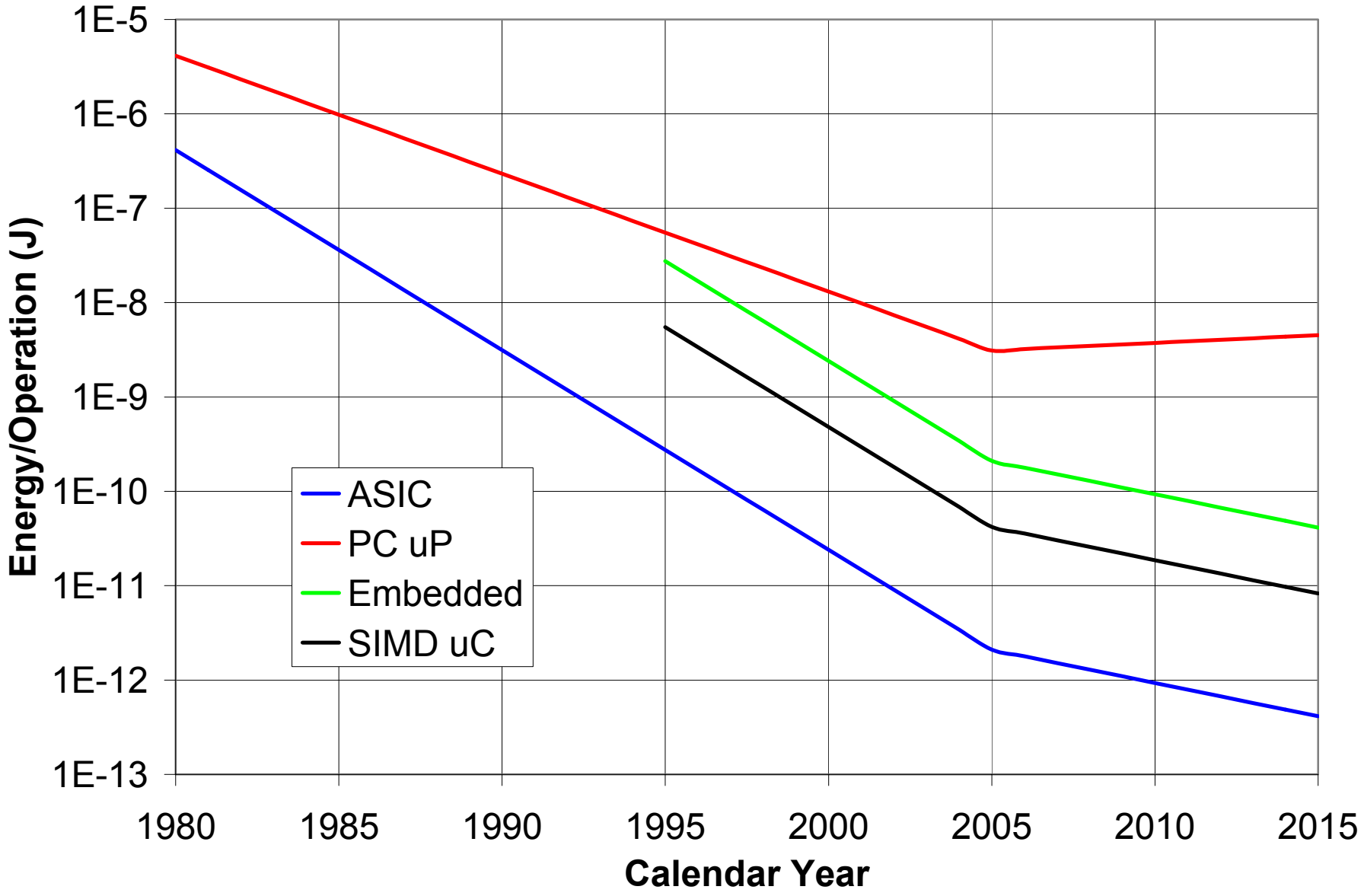
# Parallelism puts us on the embedded line



# Solution 2: Reduce Control Overhead

- Instruction sequencing, fetch, decode, pipeline, etc… consume a large fraction of power
- SIMD amortizes cost of one instruction over many operations
- Can still do conditional code efficiently
- Microcode eliminates I-cache misses and refills – explicit management of code storage

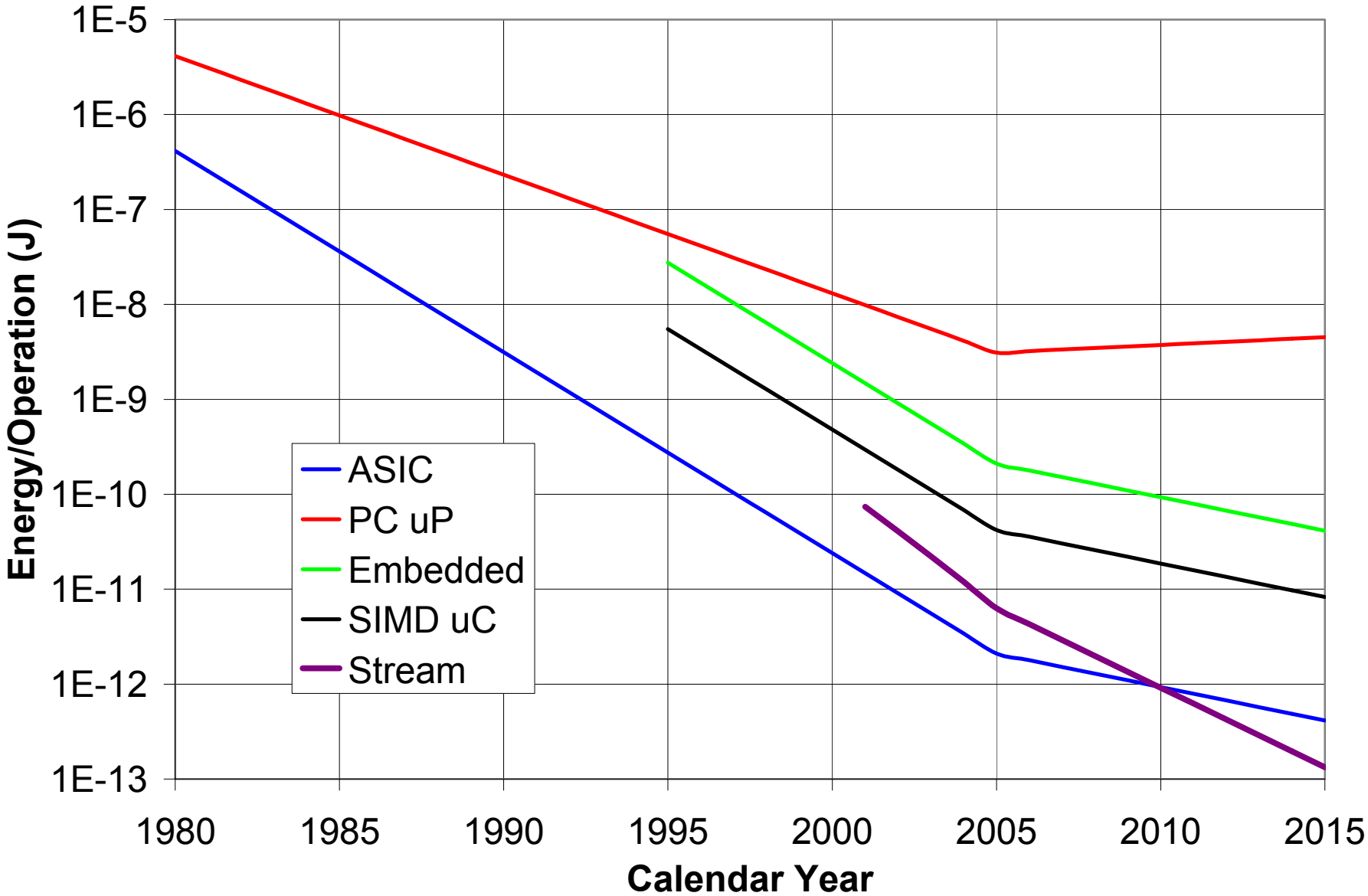
# Explicit SIMD Control

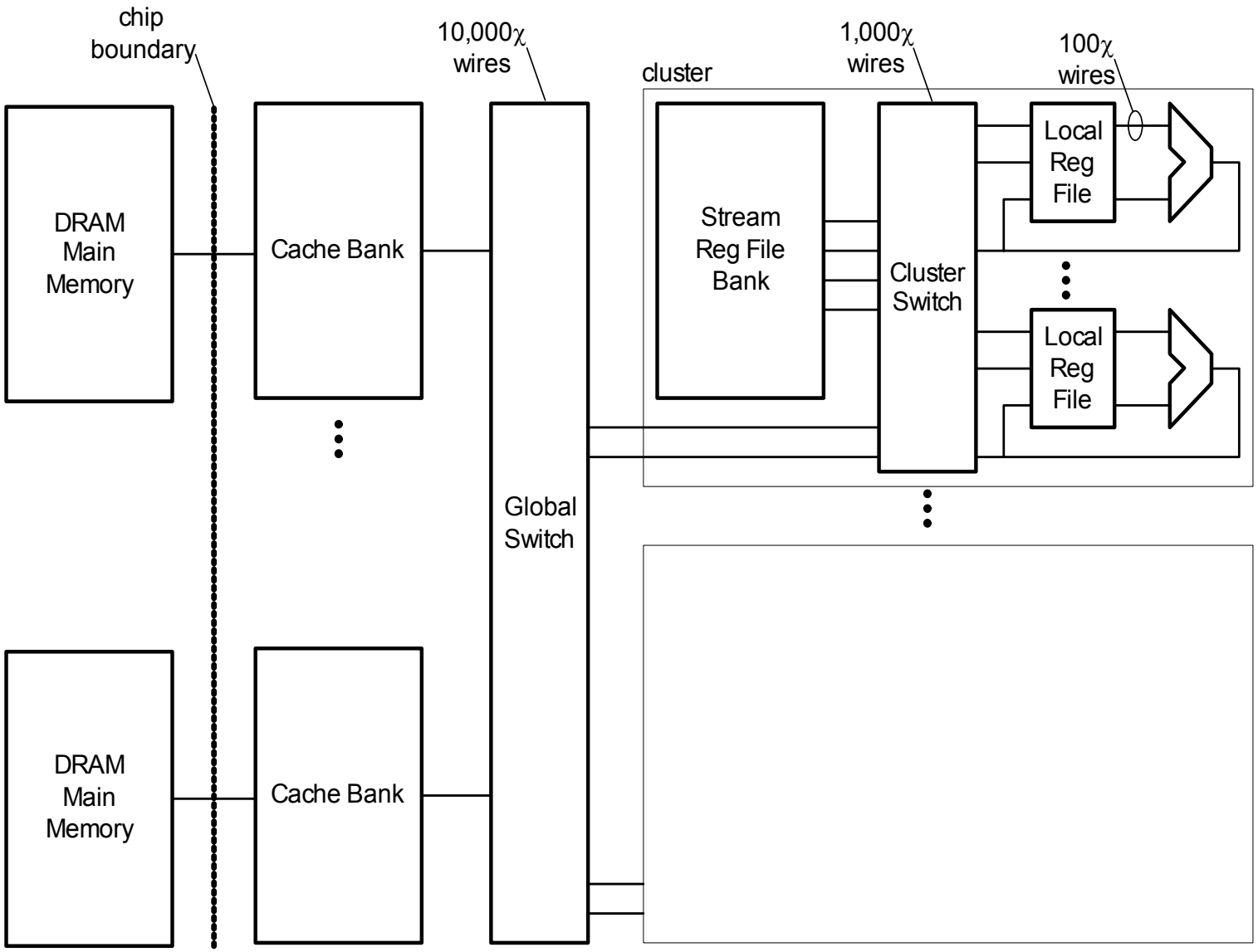


# Solution 3: Reduce Data Movement

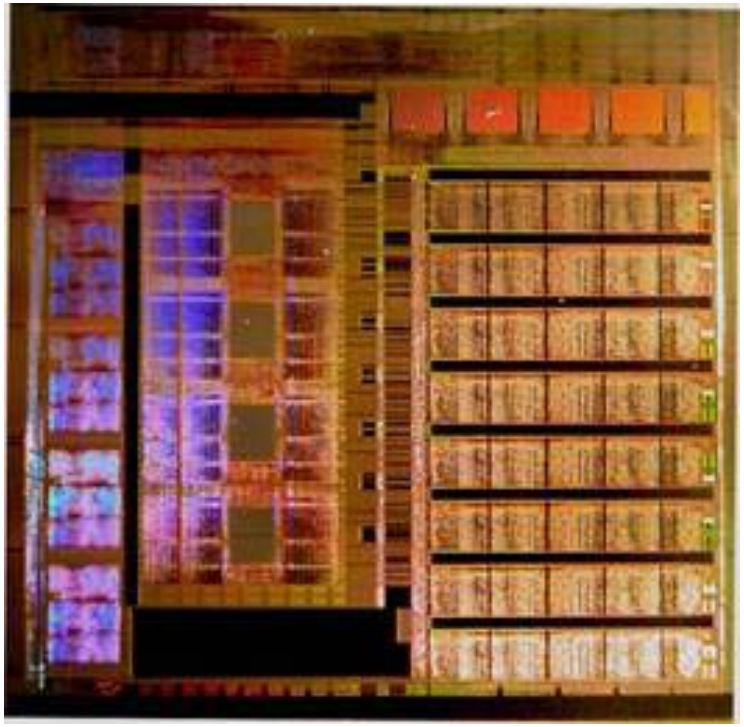
- Much of the remaining power is from data movement
  - Cycling large multi-port register arrays
  - Forwarding structures
  - Accessing data caches
    - N-way, tag arrays.
- Reduce by making data movement explicit
  - Distributed register file
  - Explicitly managed local memory (Stream Reg File)

# Stream Processors

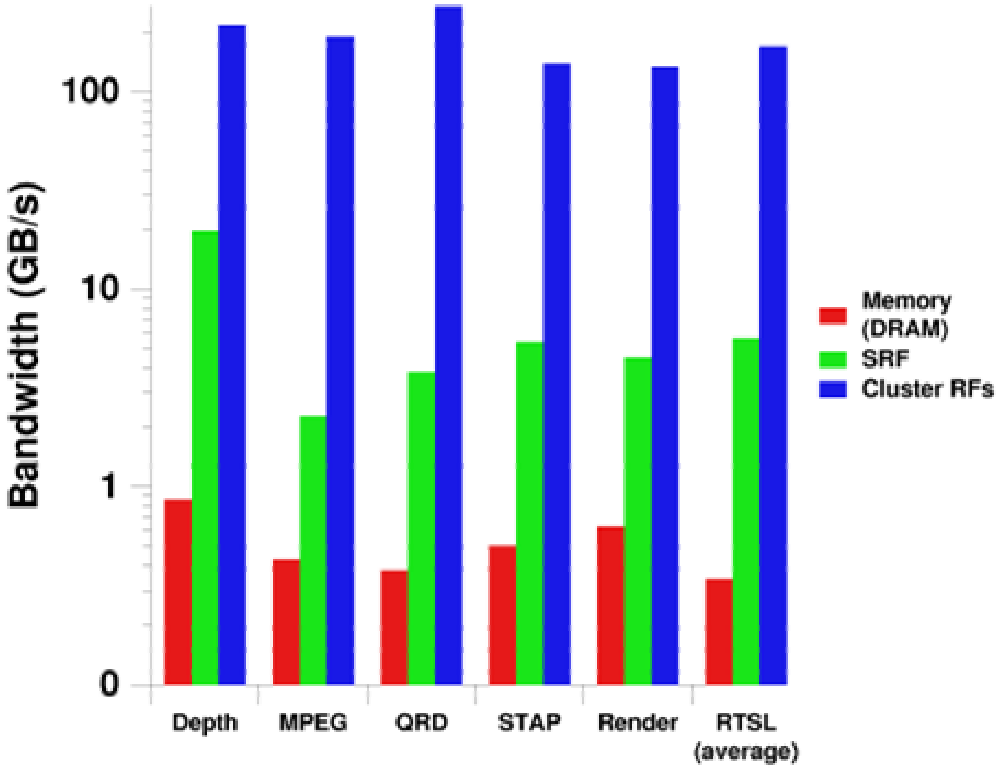





Horowitz and Dally, ISSCC 2004

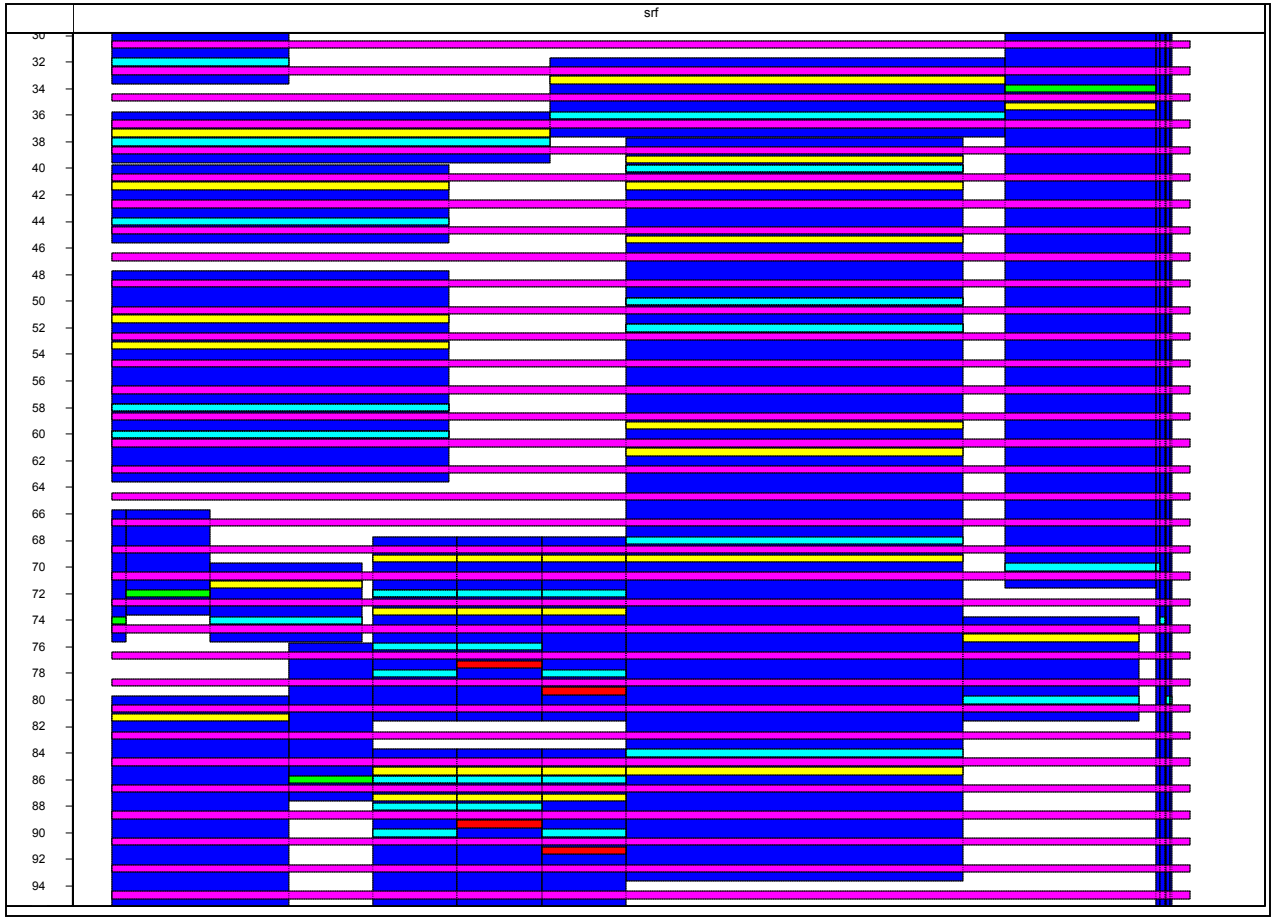


**Imagine Stream Processor**



# Stream Compilation Technology

- SRF allocation



# Summary

- Modern processors are energy inefficient
  - 1,000x vs ASIC
- We are approaching a crisis
  - Leakage, ASIC NRE
- Solution
  - Parallelism – many small processors
  - Control – SIMD with explicit code management (uC)
  - Data – Distributed and hierarchical registers
- Stream Processors
  - Very little overhead
  - Perfect load balance
  - Regular design