Vector IRAM: ISA and Micro-architecture

Christoforos E. Kozyrakis

Computer Science Division
University of California, Berkeley

kozyraki@cs.berkeley.edu
http://iram.cs.berkeley.edu/

Outline

- Project motivation, goals and approach
- Vector IRAM ISA
- VIRAM-1 micro-architecture
- Project status

Project Motivation

- Processor-memory gap is growing exponentially
- Applications shifting from engineering/desktop to multimedia
  - importance of performance of media functions
    importance of real-time predictable performance
- Embedded/ portable systems gain popularity
  - importance of energy consumption
  - importance system size
- Focus on processors for portable, multimedia systems

The Vector IRAM Approach

Vector processing

- multimedia ready
- predictable, high performance
- simple
- energy savings
- high code density
- well understood programming model

Embedded DRAM

- high memory bandwidth
- low memory latency
- energy savings
- system size benefits

Serial I/O

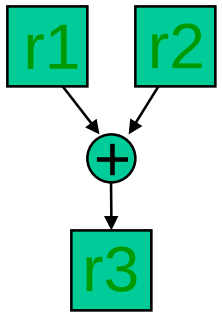
- Gbit/sec I/O bandwidth
- low pin count
- low power

Outline

- Project motivation and goals
- Vector IRAM ISA
  - Overview of VIRAM ISA extensions
  - Fixed-point and DSP support
  - Conditional and speculative execution
  - Memory model
- VIRAM-1 micro-architecture
- Project status

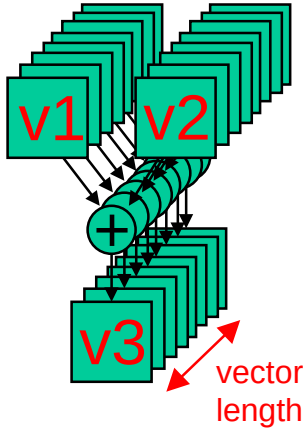
Vector Execution Model

SCALAR
(1 operation)



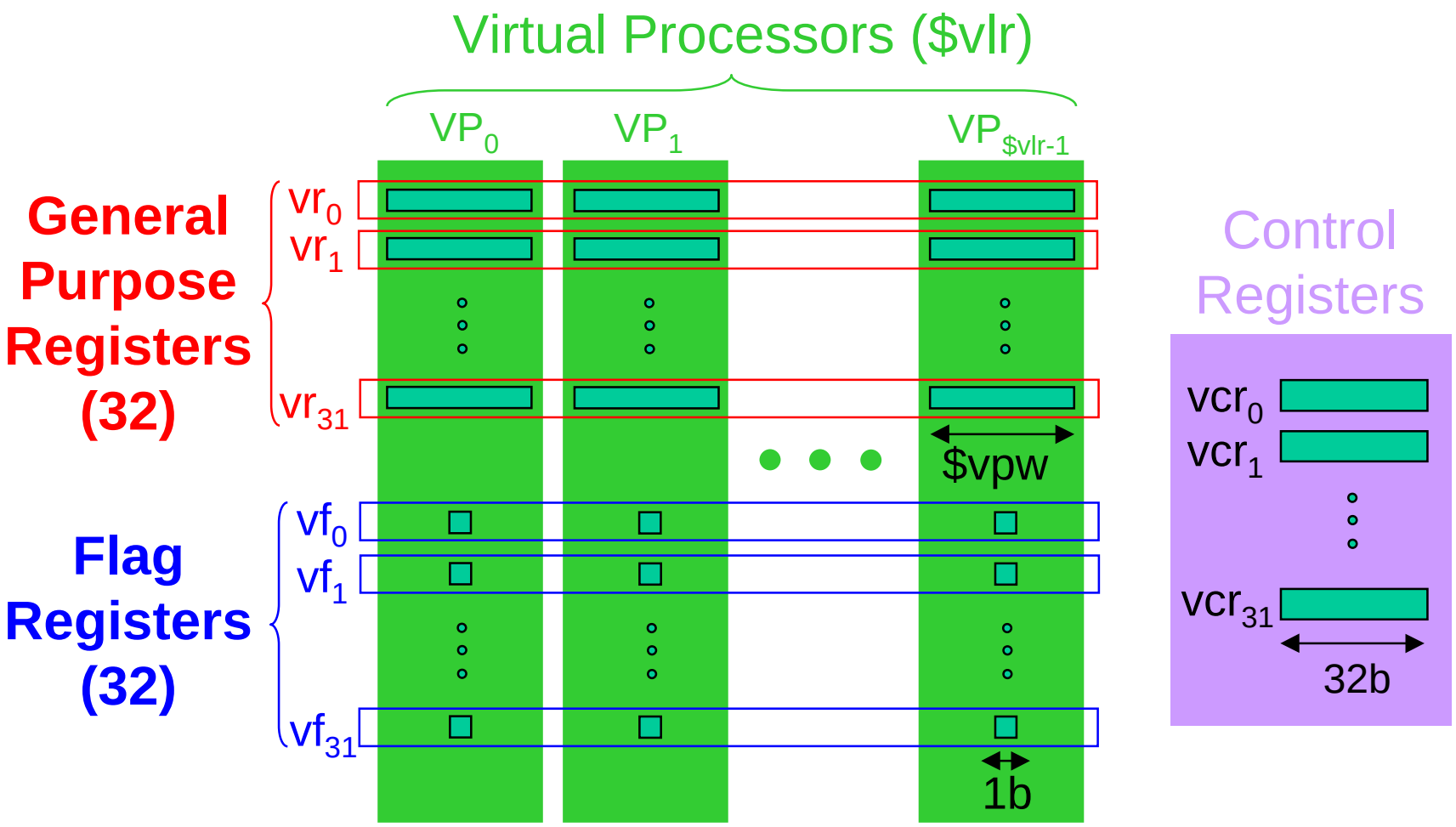

```
add r3, r1, r2
```

VECTOR
(N operations)




```
add.vv v3, v1, v2
```

Vector Architectural State



Overview of V-IRAM ISA Extensions

Scalar MIPS-V scalar instruction set

Vector ALU { alu op } { s.int, u.int, s.fp, d.fp } { 8, 16, 32, 64 } { .v, .vv, .vs, .sv }

All ALU / memory operations under mask

Vector Memory { load, store } { s.int, u.int } { 8, 16, 32, 64 } { 8, 16, 32, 64 } { unit stride, constant stride, indexed }

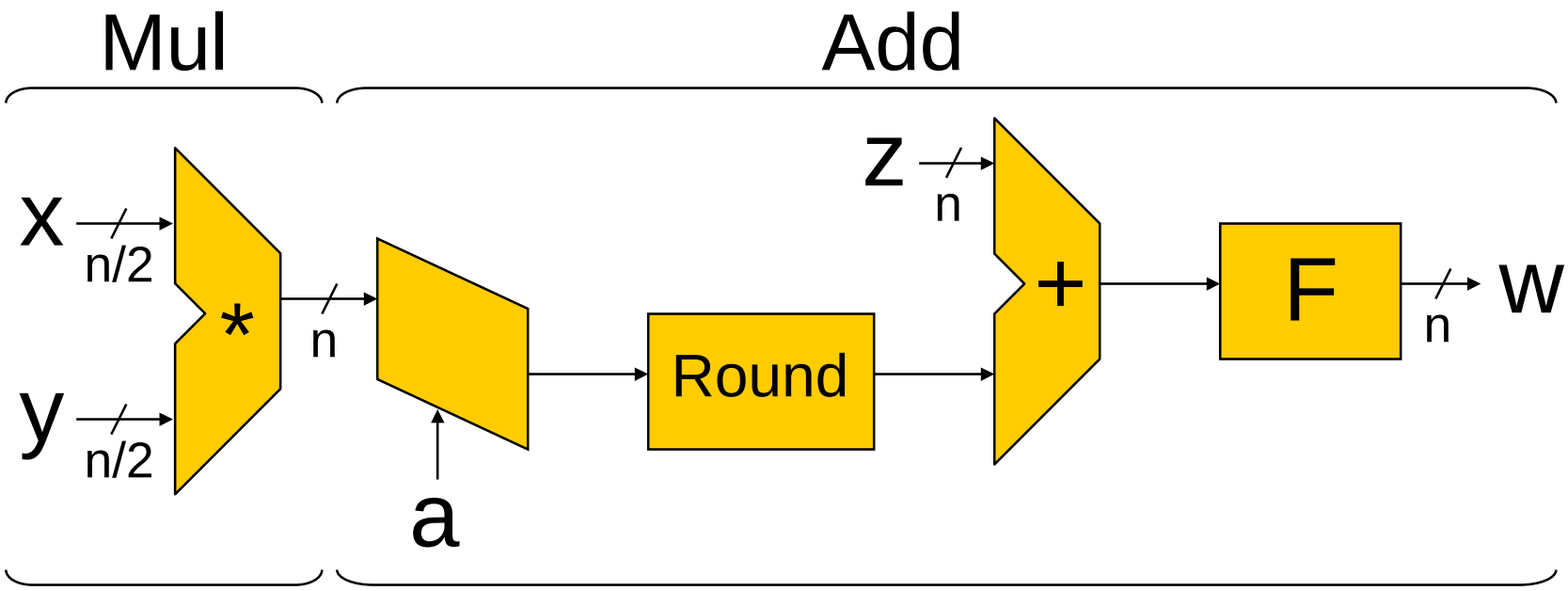
Vector Registers { 32 x VL x 64b data; 32 x 4VL x 32b data; 32 x 8VL x 16b data } + { 32 x VL x 1b flag; 32 x 2VL x 1b flag; 32 x 8VL x 1b flag }

Plus: **flag**, **convert**, **fixed-point**, and **transfer** operations

Fixed-point and DSP support

- GOAL: Competitive DSP performance
- Many DSP features already provided
  - narrow data widths [provided]
  - high speed MACs [instruction chaining]
  - multiple LD/ST per cycle [multiple memory units]
  - auto increment / decrement [strided memory access]
  - zero overhead loops [vector instructions]
  - fixed ✞ floating convert [provided]
  - bit reverse addressing [use better FFT algorithm]

Fixed-point Multiply-Add Model



Round = { truncate | round nearest even | round nearest up | jam }

F = { signed saturate | unsigned saturate | shift by one }

Fixed-point instructions

- Vector half-width integer multiply
- Vector fixed-point shift and add
- Vector saturate
- Vector saturating left arithmetic shift

Conditional (Predicated) Execution

- Almost every vector instruction is executed subject to one of two vector masks
- 15 GP flag register provided to buffer masks or operate on them
- 6 flag logical and 13 flag processing instructions (like population count, iota etc)
- 15 flag registers used for sticky exception bits for arithmetic/FP operations and speculative operations

Speculative Execution

- Vectorizing loops with conditional exit conditions
  - Need to speculate past loop exit
  - Need to temporarily suppress exceptions
- Speculation controlled by software
- Solution:
  - A duplicate set of arithmetic exception flag registers
  - A flag register reserved for load faults
  - Speculative loads and speculative arithmetic instructions write these duplicate exception bits

Speculative Execution (cont.)

- Perform loads and enough arithmetic to determine loop exit condition
  - Stores cannot be speculated!
- Generate mask to exclude iterations after loop exit (flag processor instruction)
- `VCOMMIT` instruction (under mask):
  - ORs speculative flags into real flags
  - Raises memory exceptions

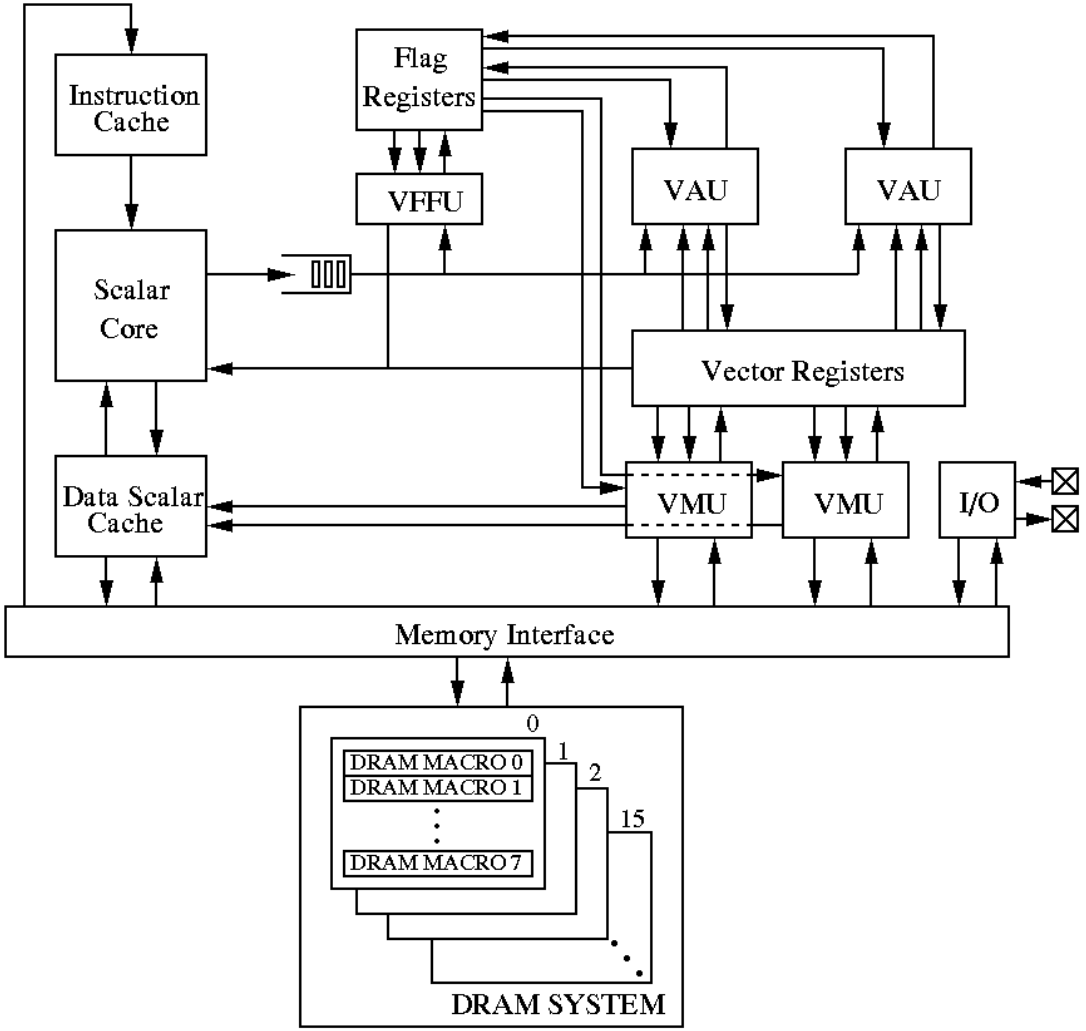
Memory Model

- Relaxed consistency to simplify hardware: no guarantee about ordering of memory operations, even within the same VP
- Register interlocks provided on a per-element basis
- Vector memory barrier used for ordering between scalar unit and vector unit and between VPs
- Indexed memory operations do not specify ordering; separate ordered indexed store instruction

Outline

- Project motivation and goals
- Vector IRAM ISA
- VIRAM-1 micro-architecture
  - Overview of VIRAM-1 micro-architecture
  - Vector pipelines
  - Memory system architecture
- Project status

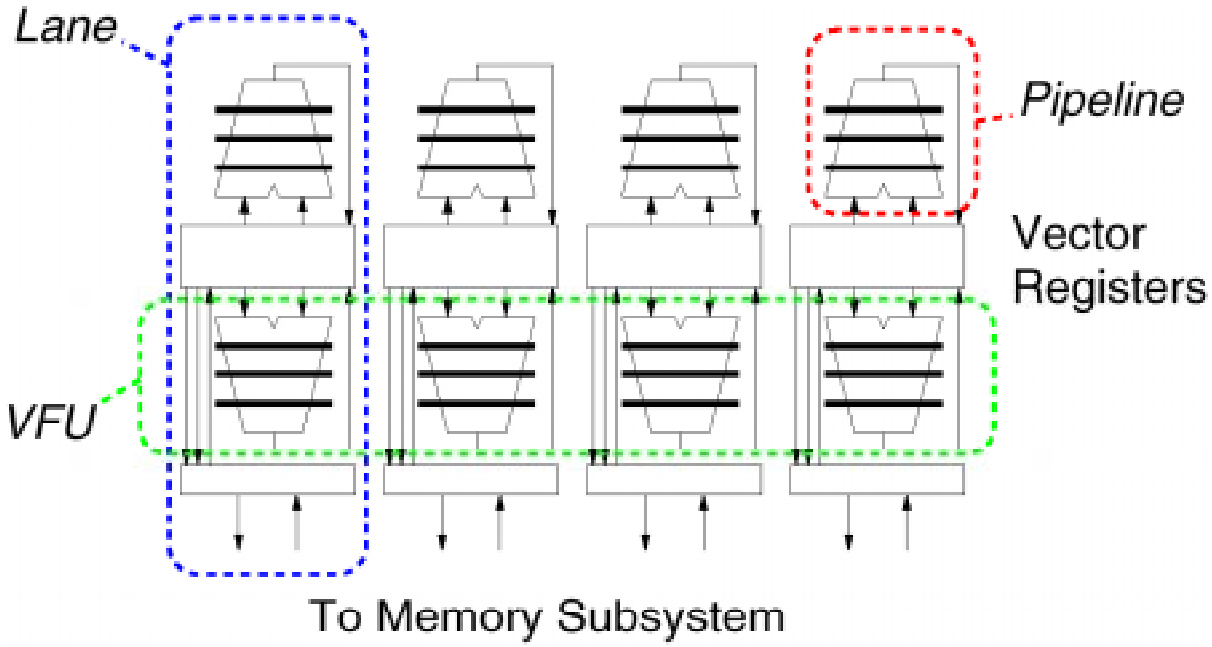
VIRAM-1 Block Diagram



VIRAM-1 Features

- Scalar unit
  64-bit MIPS core with FP unit
  8KB I+D caches, write-through
  cache invalidation interface
- Vector unit
  maximum vector length 32
  64, 32, 16 bit data-types
  2 vector arithmetic units
  2 vector flag processing units
  4 pipelines per functional unit
  2 vector load/store units
  64 entry vector TLB, multi-ported

Vector Pipelines

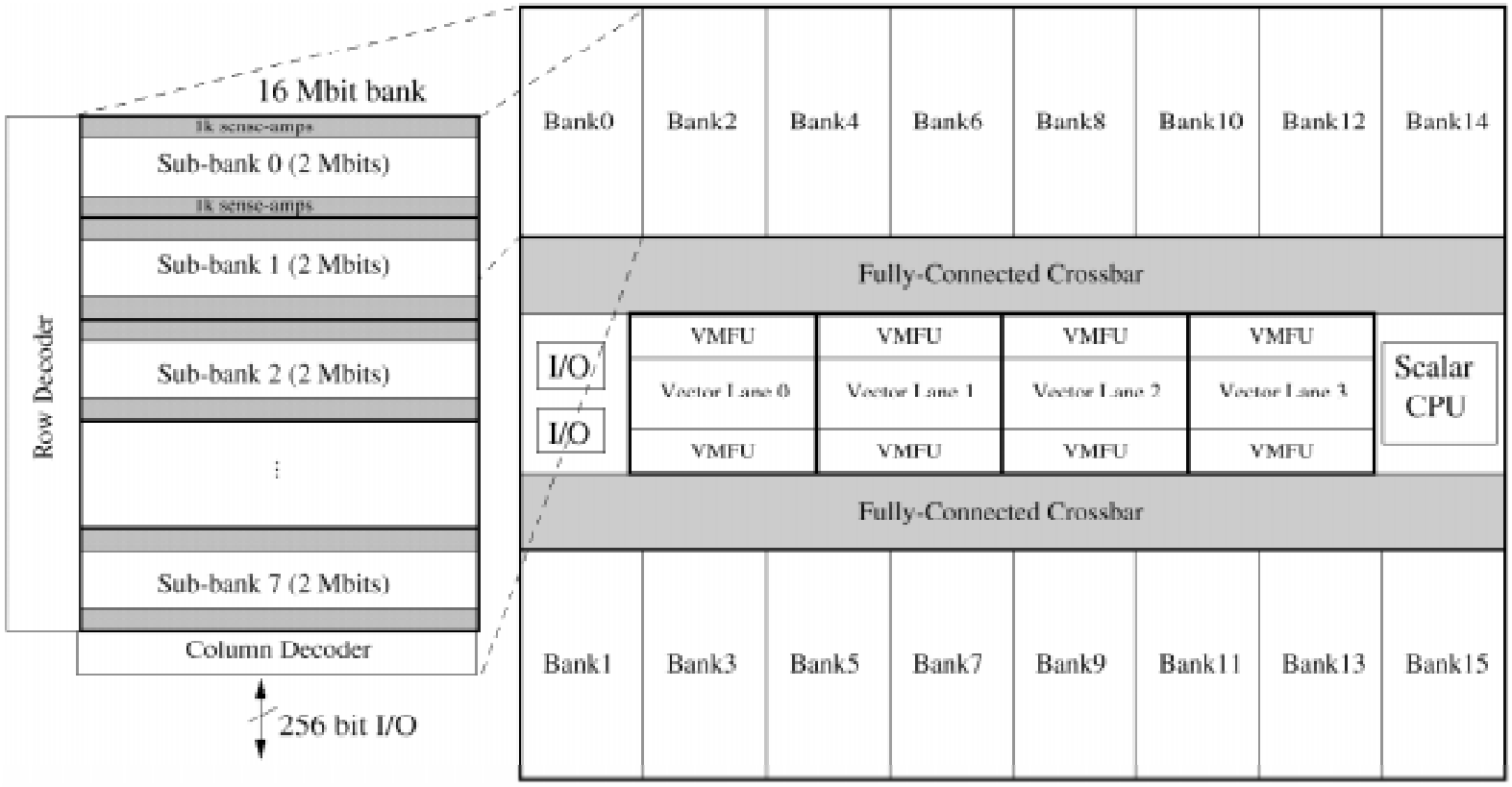


- Multiple pipelines can increase performance OR
- Energy decrease by decreasing clock frequency and power supply

VIRAM-1 Memory System

- 16 to 32MB DRAM
- 16 independently addressed banks
- 8 2Mbit DRAM macros per bank with 256-bit synchronous interface
- Memory crossbar
  - interconnects scalar, vector unit and I/O to memory
  - 8 addresses per cycle
  - 12.8GB/sec maximum data bandwidth per direction
  - implemented using low-swing techniques

VIRAM-1 Floorplan

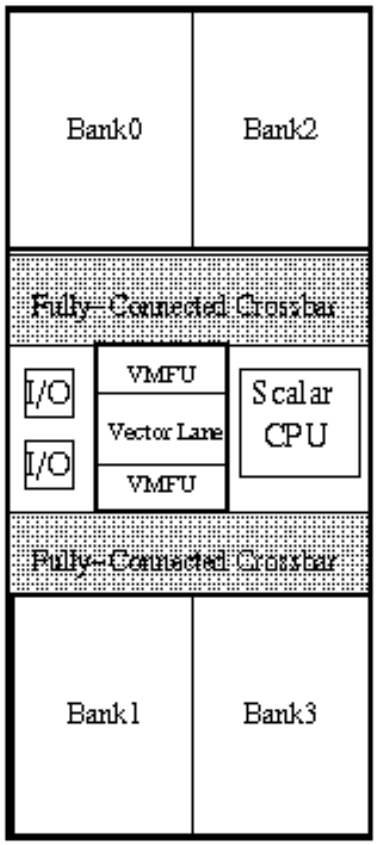


VIRAM-1 Goals

Technology	**0.20 micron, 5 metal layers, embedded DRAM-logic process**
Memory	**16-32 MB**
Die size	**250-300 mm^2**
Vector pipelines	**4 64-bit (or 8 32-bit or 16 16-bit)**
Clock Frequency	**200MHz scalar, 200MHz vector, 100MHz DRAM**
Serial I/O	**4 lines @ 1 Gbit/s**
Power	**2 W @ 1.5 volt logic**
Performance	**1.6 $GFLOPS_{64}$ – 6.4 $GOPS_{16}$**

First microprocessor above 0.25B transistors?

Scaling Down VIRAM-1



- Scaled-down version automatically generated from the the original
- 8 MB in 4 banks
- Vector unit with single pipeline per functional unit => same control
- die: 80 mm^2
- transistors: 70M
- power: 0.5 Watts
- performance: 0.4 $GFLOPS_{64}$
  1.6 $GOPS_{16}$

Project Status

- ISA extensions frozen
- Micro-architecture still under development but design has started
- Developing simulation infrastructure
- Designed 2 test-chips for circuit evaluation
  - serial I/O @ 1Gbit/s
  - embedded DRAM and on-chip crossbar
- Expected VIRAM-1 tape-out: early 2000

Acknowledgments

- Thanks for advice/support: DARPA, California MICRO, ARM, Hitachi, IBM, Intel, LG Semicon, Microsoft, Mitsubishi, Neomagic, Samsung, SGI/Cray, Sun Microsystems
- The IRAM/ISTORE cast: D. Patterson, K. Asanovic, A. Brown, J. Gebis, B. Gribstad, R. Fromm, J. Golbus, K. Keeton, C. Kozyrakis, J. Kubiatowicz, D. Martin, S. Perissakis, R. Thomas, N. Treuhaft and K. Yelick