Vectors and GPUs

Ryan Lee

Adapted from prior course offerings

6.823 Fall 2023

Vector Computers

- Idea: Operate on vectors instead of scalars
  - ISA is more expressive, therefore captures more information
  - Extract **data-level parallelism** (same operation on multiple pieces of data in parallel)
- Advantages:
  - No dependences within a vector
  - Reduced instruction fetch bandwidth
  - Amortized cost of instruction fetch and decode
  - (Sometimes) regular memory access pattern
  - No need to explicitly code loops
- Pitfalls:
  - Only works if code sequence (or parallelism) is regular

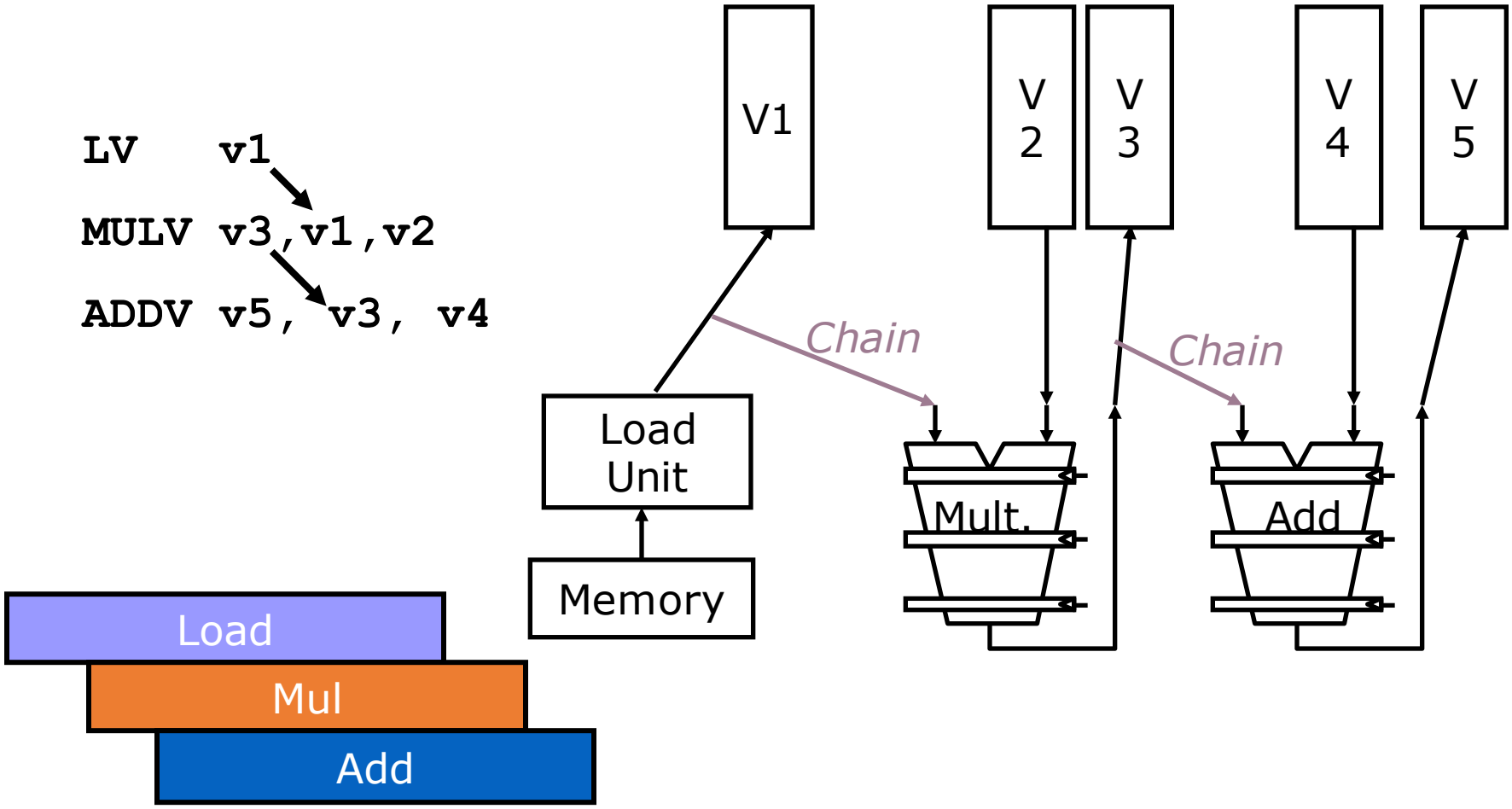
Vector Computers

Terminology:

- Vector length register (VLR)
- Conditional execution using vector mask (VM)
- Vector lanes
- Vector chaining

Vector Computers

```
LV   v1
MULV v3,v1,v2
ADDV v5, v3, v4
```



GPU: Graphics Processing Unit

» Originally designed as a graphics acceleration engine

» Has evolved into a hardware accelerator for massively parallel applications

» Think of as Multithreading + Vector Processor!

- What types of parallelism does this architecture target?

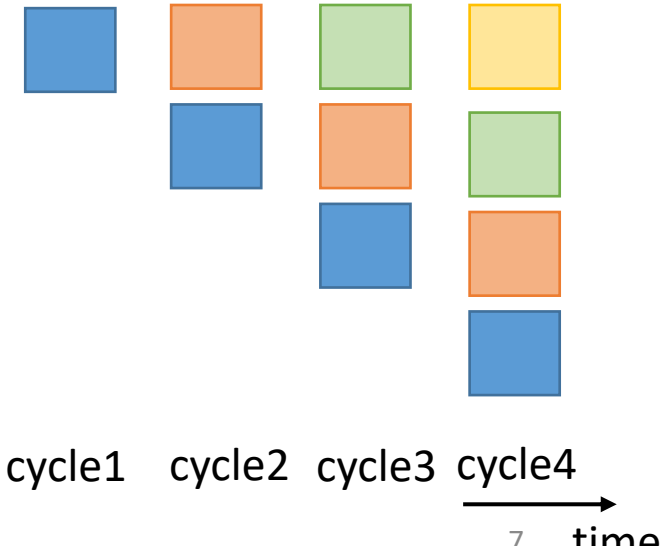
Types of Parallelism

- ILP: Instruction-level parallelism
  - Between independent instructions in a sequential program
- TLP: Thread-level parallelism
  - Between independent execution contexts (threads)
- DLP: Data-level parallelism
  - Between elements of a vector (say); same operation on multiple elements

How to Utilize Parallelism?

» Horizontal parallelism: More units working in parallel

» Vertical parallelism: Pipelining: Keep units busy when waiting for memory dependences etc.



How to Extract Parallelism?

	Horizontal	**Vertical**
ILP	Superscalar	Pipelining/OoO
TLP	Multi-core	SMT
DLP	SIMD/SIMT/Vector	Temporal SIMT

GPUs focus on TLP, DLP

Why care about GPUs?

- Massive data parallelism in today's popular workloads
  - Machine Learning
  - DNNs, LLMs
  - Graph analytics
  - Scientific Computing

Key Concepts

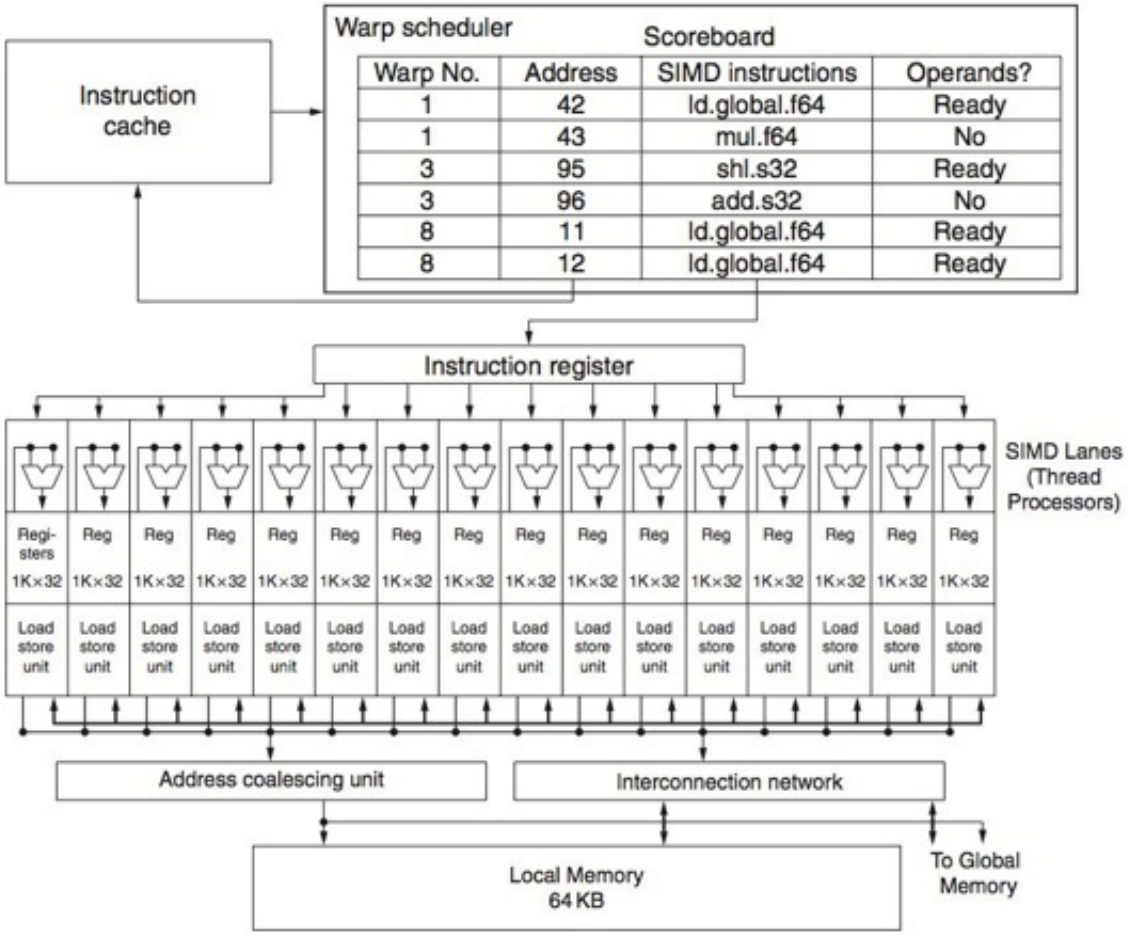
» SIMT: Single-instruction multiple-thread
  - Multiple instruction streams of scalar instructions

» Warps: A set of threads executing the same instruction (grouped dynamically by the hardware)
  - Essentially a SIMD operation formed in hardware

» SM: Streaming multi-processor

» Branch divergence: Masking

Streaming Multiprocessor



Example:

» 16 physical lanes

» Tens of warps with 32 threads per warp

» Warp scheduler issues SIMD instruction, when all threads ready