CS152/252A
Spring 2026

Multithreading, Vector Machines

Discussion 9

1 Comparing Architectures

1.1 Architectures exploiting Instruction-Level Parallelism

	How is vertical waste reduced?	How is horizontal waste reduced?	Limitations/disadvantages compared to an in-order superscalar RISC machine?
Out-of-order superscalar			
VLIW			
Vector			

1.2 Architectures exploiting Thread-Level Parallelism

	How is vertical waste reduced?	How is horizontal waste reduced?	Limitations/disadvantages compared to an in-order superscalar RISC machine?
Fine-grained / vertical multithreading			
Simultaneous multithreading			

2 Fine-grained Multithreading

In this problem, we would like to investigate the performance of the following C program on a multithreaded architecture. The arrays `A`, `B`, and `C` contain double-precision floating-point numbers.

```
for (int i=0; i<M; i++) {
    C[i] = A[i] + B[i];
}
```

We rewrite the loop to split the work across N threads:

```
// TID is the thread ID (0 to N - 1)
for (int i = TID; i < M; i += N) {
    C[i] = A[i] + B[i];
}
```

The equivalent assembly for the multithreaded version can be found below:

```
loop: fld  f1, 0(x1)
    fld  f2, 0(x2)
    fadd f3, f1, f2
    fsd  f3, 0(x3)
    addi x1, x1, 8N
    addi x2, x2, 8N
    addi x3, x3, 8N
    addi x4, x4, -1
    bnez x4, loop
```

Assume:

- Single-issue in-order processor
- 1 cycle integer operations, 3 cycle floating-point arithmetic operations, 2 cycle memory operations
- Fine-grained multithreading with fixed round-robin scheduling
- Perfect branch prediction.

2.1 How many threads need to fully utilize the pipeline?

2.2 What is the peak performance in FLOPs/cycle?

2.3 Can peak performance be reached with a single thread by reordering instructions in the loop? Use arrows to reorder instructions.

```
loop:
1: fld  f1, 0(x1)
2: fld  f2, 0(x2)
3: fadd f3, f1, f2
4: fsd  f3, 0(x3)
5: addi x1, x1, 8N
6: addi x2, x2, 8N
7: addi x3, x3, 8N
8: addi x4, x4, -1
9: bnez x4, loop
```

3 Simultaneous Multithreading

3.1 Which resources must be duplicated to support simultaneous multithreading? Which resources can be shared?

Program Counter	
Fetch Unit	
Rename Table	
Physical Register File	
Issue Window	
Functional Units	
Reorder Buffer	

3.2 ICOUNT policy prioritizes fetching from the thread with the least in-flight instructions. Why does this improve throughput?

4 Vector Programming Lane Tradeoff

Suppose we want to add two vector registers (`add v1, v2, v3`), followed by another addition to different registers (`add v4, v5, v6`). The next instruction after that uses a different functional unit. `VLR=MAXVL=32`. What would you choose between an ALU with 8 lanes and 2 cycles dead time, and an ALU with 16 lanes and 8 cycles dead time?

5 Vector v.s. Packed-SIMD

5.1 What are the distinguishing features between a vector architecture and a packed-SIMD architecture? List theadvantages of each.

- Distinguishing features:

- Advantages for Vector architecture:

- Advantages for Packed-SIMD: