Section 11: VLIW, Vectors, & Multithreading

CS 152
Spring 2009

TA: Scott Beamer

Problem 1: VLIW
Why is predicated execution useful for a VLIW system? Why doesn't every superscalar do it as well?

Problem 2: Vector
Vector machines often have a lot of memory bandwidth (SX-9 has 256GB/s!). Why do they need it and why do current superscalars not provide as much?

Problem 3: Comparison
Please fill in the table on the next page. At a high level we will try to get a qualitative view of the different architectures we have discussed and some of the tradeoffs between them. This table sums up a great deal of content, so it will be approximate and gloss over many details.

- Hardware Features - Comment on what makes up this architecture, especially what structures or components it might have that the others don't.
- Performance/Cost - To simply compare these architectures on performance would be unfair since with more or less resources they could be made better or worse. Of course this is highly workload dependent, so just consider each architecture at its peak performance.
- Ideal Workloads - What programs will achieve the best performance on this architecture.
- Adversarial Workloads - What workloads will achieve the worst performance on this architecture.
- Variance in Performance - How big is the gap between the best and worst performance.
- Flexibility - How many workloads achieve peak or near peak performance?

	Simple 5 Stage In-Order	Out-of-Order Superscalar	Vector	VLIW	Multithreaded
Hardware Features					
Performance/ Cost					
Ideal Workloads					
Adversarial Workloads					
Variance in Performance					
Flexibility					