ECE 571 – Advanced Microprocessor-Based Design Lecture 7

Vince Weaver

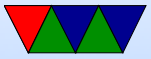
http://www.eece.maine.edu/~vweaver

vincent.weaver@maine.edu

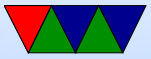
23 September 2014

Announcements

- Homework #2 will be delayed
- The previous Homework grades will be ready soon

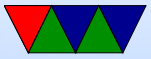


CPUs



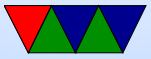
Simple CPUs

- Ran one instruction at a time.
- Could take one or multiple cycles (IPC 1.0 or less)



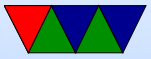
Pipelined CPUs

- 5-stage MIPS pipeline
- IF, ID, EX, MEM, WB
- From 2-stage to Pentium 4 31-stage



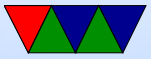
Pipelined CPUs

- IF = Instruction Fetch. Fetch 32-bit instruction from L1-cache
- ID = Decode
- EX = execute (ALU, maybe shifter, multiplier, divide) Memory address calculated
- MEM = Memory – if memory had to be accessed, happened now.
- WB = register values written back to the register file



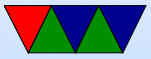
Pipelined CPUs

- What's the performance benefit?
- What power/energy implications are there?



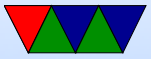
Data Hazards

- Data Hazards
- RAW – "true" dependency – problem. Bypassing?
- WAR – "anti" dependency – not a problem if commit in order
- WAW – "output" dependency – not a problem as long as ordered
- RAR – not a problem



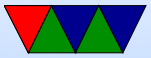
Structural Hazards

- CPU can't just provide. Not enough multipliers for example



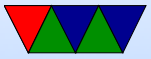
Control Hazards

- How quickly can we know outcome of a branch
- Branch prediction? Branch delay slot?



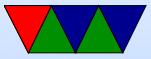
Memory Delay

- Memory/cache is slow
- Need to bubble



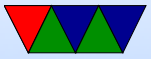
Multi-Issue

- Dual issue example. Can have theoretical IPC of 2.0
- Can have unequal pipelines.
- Energy cost? Extra area for doubled pipeline (static) and cost of NOPs when instruction cannot be issued.



Out-of-Order

- Tries to exploit instruction-level parallelism
- Register Renaming
- Re-order buffer
- Speculative execution
- Energy concerns?
- TODO: include diagram



Counters

- Front-end stalls – fetch, decode, icache misses
- Back-end stalls – memory accesses