Advanced Concepts/Strategies

Modern Machines

- MIPS R2000/3000 looks a lot like the machine we built.
- Modern implementations are more complicated/sophisticated:
  - Multiple pipelines: superscalar
  - Trend: exploit instruction level parallelism by executing multiple instructions at once
  - Modern machines issue 4 or more instructions at once.
  - Challenge: scheduling instructions
- Deep pipelines: superpipelines
  - Trend: reduce cycle time by adding more pipe stages (8 or more)
  - Challenge: longer load and branch delays, more forwarding required.

Modern Machines: Buzzwords

- All modern processors improve performance via pipelining. Additional gains:
  - Superscalar
  - Superpipelined
  - Dynamic scheduling
  - Register renaming
  - Branch prediction
  - Speculative execution
  - Dynamic translation (into other instruction sets...)

Scheduling

- Show the dependencies that exist in the following code:
  lw $t0, 8($fp)
  addi $at, $0, 2
  sllv $t0, $t0, $at
  lw $t1, 68($fp)
  add $t1, $t1, $t0
  lw $t0, 72($fp)
  sw $t0, 0($t1)

- 3 kinds of dependencies: read-after-write (data), write-after-read (anti), write-after-write (output)
- The above instructions can be scheduled at compile-time or run-time.

Register Renaming

- Modern processors also rename registers.

  The same example with registers renamed:
  lw $P0, 8($fp)
  addi $P1, $0, 2
  sllv $P2, $P0, $P1
  lw $P3, 68($fp)
  add $P4, $P3, $P2
  lw $P5, 72($fp)
  sw $P5, 0($P4)

- Eliminates write-after-write and write-after-read dependencies.
- Allows for more flexible scheduling...

Block Diagram
What do the units do?

- **Instruction queue**: holds a pile of to-be-executed instructions. These may come from different paths on a branch.
- **Dispatch unit**: tries to find the best set of instructions to send to the functional units. It may also rename registers.
- **Functional units**: integer ALU, fp ALU, branch, load/store
- **Branch predictor**: maintains branch history
- **Instructions** wait at reservation stations until their operands are ready
- The **commit unit** makes sure instructions change the register file in an orderly manner (or maybe not at all if they were on the wrong side of a branch).

Branch Prediction

- Keep a table mapping branches to history (did we take the branch last time?).
- With 256 entries: If we find a branch at address N, we locate its entry in the table like this:
  \[
  \text{index} = N \mod 256 \\
  (\text{Or better: } \text{index} = N \& 0x000000FF)
  \]
- Simplest predictors are one bit: taken/not-taken
- Think about a loop that goes around 10 times.
  - How accurate is the one bit scheme?

Branch Prediction 2

- A two-bit scheme is better:
- How well will this scheme perform on the previous loop?
- If calculating a branch target takes a long time, there may also be a branch target buffer.

AMD Athlon

- 30+ million transistors; clock speeds > 1GHz
- Front end of the processor translates incoming CISC stream into RISC86/MacroOp instructions
- RISC86 instructions are passed to instruction control unit, which manages up to 72 instructions at once. The instructions are scheduled here. Up to 9 instructions may issue per cycle
- There are 9 pipelines: 3 integer; 3 address calculation; 3 FP/MMX. Integer pipelines have 10 stages.
- The instruction control unit also handles commits (up to 9 per cycle)
- Branch prediction table has 2048 entries.

Athlon Specifics

- 30+ million transistors; clock speeds > 1GHz
- Front end of the processor translates incoming CISC stream into RISC86/MacroOp instructions
- RISC86 instructions are passed to instruction control unit, which manages up to 72 instructions at once. The instructions are scheduled here. Up to 9 instructions may issue per cycle
- There are 9 pipelines: 3 integer; 3 address calculation; 3 FP/MMX. Integer pipelines have 10 stages.
- The instruction control unit also handles commits (up to 9 per cycle)
- Branch prediction table has 2048 entries.