PSATSim
An Interactive Graphical Superscalar Architecture Simulator for Power and Performance Analysis

Clint W. Smullen, IV
Tarek M. Taha

Clemson University
Department of Electrical and Computer Engineering

WCAE 2006
Purposes

1. Tool for instructors
   - Demonstrate superscalar architectures
   - Use in-class

2. Framework for students
   - Explore the power and performance

3. Interactive execution

4. Wide range of configuration options
Power Modeling

- Uses the Wattch power model
  - D. Brooks, V. Tiwari, and M. Martonosi 2000
- High-level modeling of major components
Power Modeling

- Tracks activity use of each component
- Average activity use scales maximum energy consumption
- Averages the sum of component energy usage over length of execution
Capabilities

- Uses SimpleScalar ISA
  - Related to MIPS ISA
  - Easy to understand instruction format
- Statistically models branch misspeculation
  - This improves accuracy of power model
Capabilities

• Statistically models cache hierarchy

• Uses trace files
  o SPEC benchmark traces are provided with program
  o Reduces overhead in demonstrations
  o Shortens iteration latency for students
In-Order Front-end

- Misspeculated instructions displayed with strikethrough
Coloration

- Makes it easy to see dependencies
Renaming Table

- Provides false hazard resolution
- Instructions without color have already produced a value
Reorder Buffer

- Provides in-order completion of instructions
- Uncolored opcodes have finished and await commit
- Up to the superscalar width in instructions are committed each cycle
Reservation Stations

Distributed:

Centralized:

Hybrid:
Functional Units

Standard:

Simple:

Complex:
Configuration

New Simulation

**General**
- Superscalar Factor (1-16): 3
- # of Rename Entries (1-512): 8
- # of Reorder Entries (1-512): 12
- Separate Decode and Dispatch

Enter the path for the trace file:
traces/applu.tra

Enter the path for the output file:
output.xml

**Execution**
- Execution Unit Architecture: Standard
- Reservation Architecture: Distributed
- # of Entries per Reservation Station (1-8): 2
- # of Integer Execution Units (1-8): 2
- # of Floating Point Execution Units (1-8): 2
- # of Branch Execution Units (1-8): 1
- # of Memory Execution Units (1-8): 1

New Simulation

**General**

**Execution**

**Memory/Branching**
Interactive Use

- User can force a branch misspeculation
- Single- and auto-step through the execution
- Pause automated execution
- Quickly finish execution
Project Use

- Use of traces gives shorter simulation time
- Wide range of architectural options
- Exploration within a given set of constraints

Simulation Results:
- Instructions: 71771
- Cycles: 47363
- Power: 21.8652 W
- IPC: 1.51534
- Execution Time: 78938.3 ns
  (Cycle time = 1.66667 ns)
Use in the Classroom

• Used in the undergraduate and graduate courses at Clemson
• Used for demonstration
• Students are asked to maximize performance within a given power envelope
Implementation

- Written in C++
  - Uses GTK+2
  - LibXML2, LibPCRE, PThreads
- 14K lines code
Availability

- Software available from:
  http://www.ces.clemson.edu/~tarek/psatsim/

- Currently available for Windows
  - Linux version should be available soon
Questions?