Future Microprocessors and Scalable Systems

Guri Sohi and Jim Smith
University of Wisconsin-Madison

Future Microprocessor Trends

- Lots of on-chips transistors available
- Lots of on-chip RAM structures available
- On-chip wires/communication expensive

New opportunities

Research Objectives

Use available opportunities to

- Take leadership role in scalable system design
- Rethink traditional architecture compact
- Break traditional architecture compact

Dependences: Core of Compact

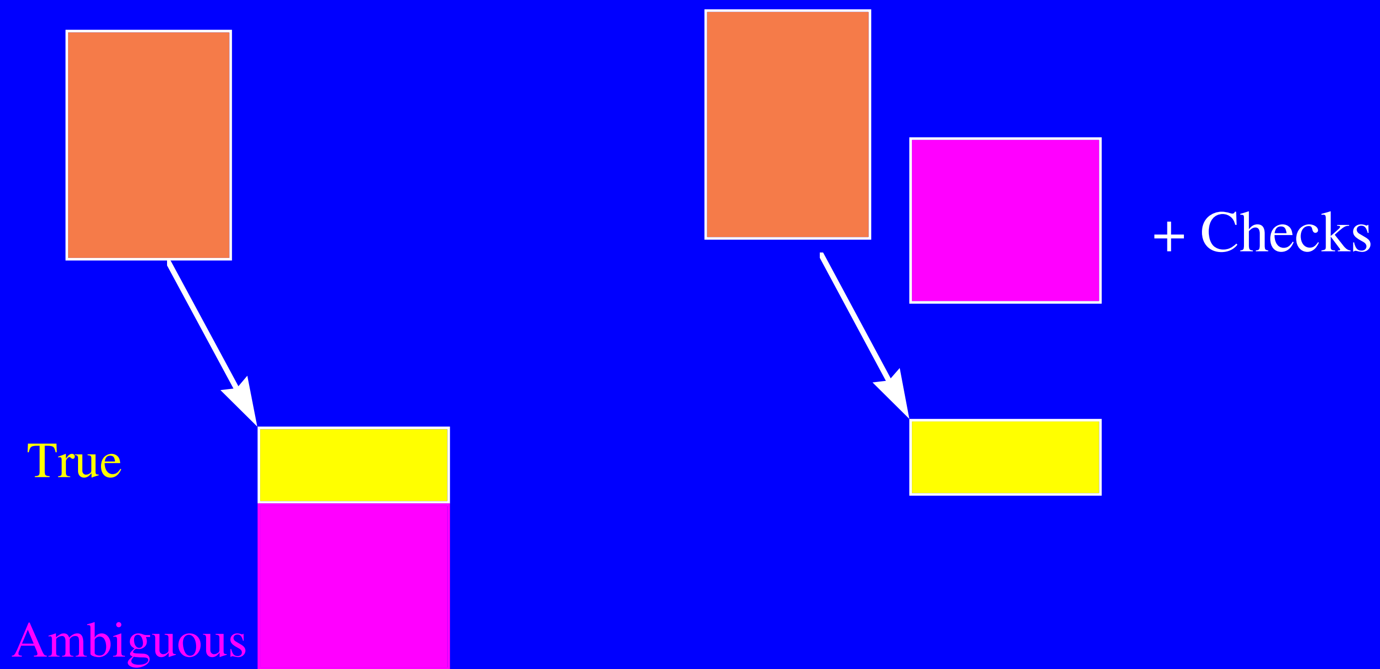
- Dependences expose latencies
- Exposed (and aggravated) dependences degrade performance

Breaking architecture compact ==
Overcome dependences

Dependences

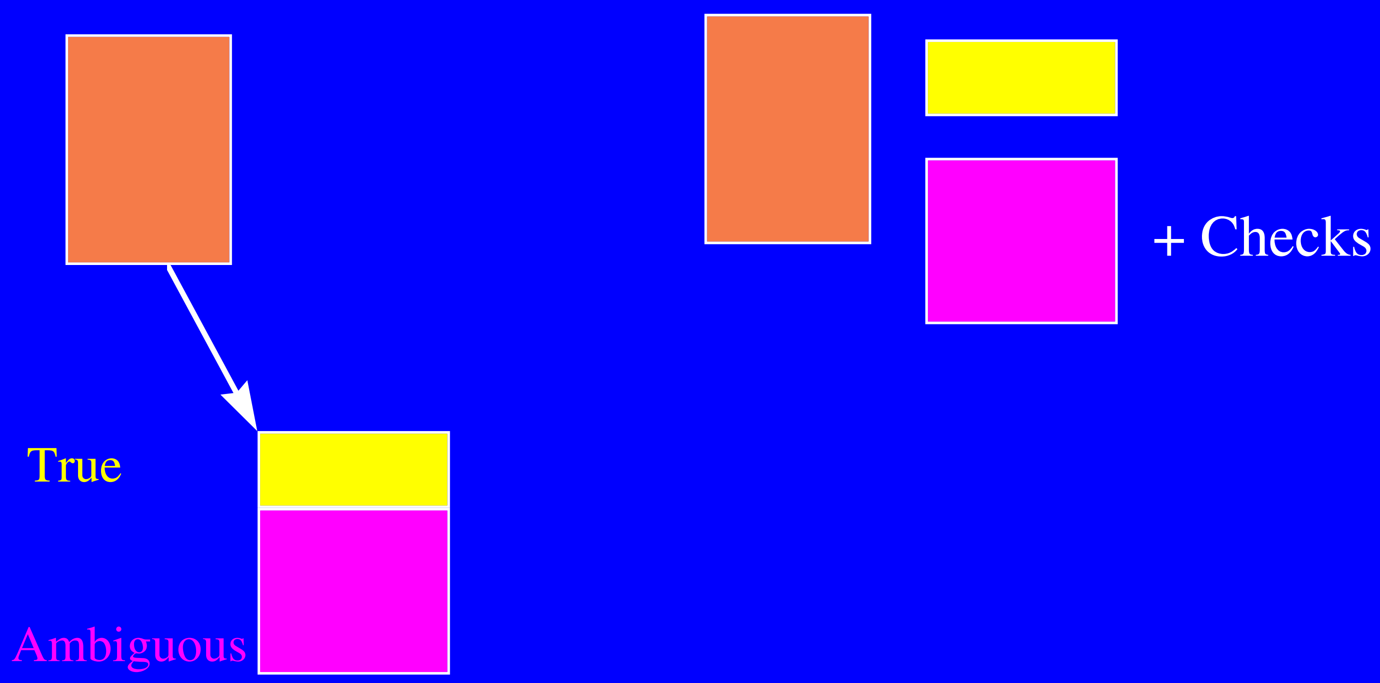
- Control Dependences
- Artificial (Name) Dependences
- Ambiguous Dependences
- True Data Dependences

Breaking Ambiguous Dependences



Data Dependence Speculation

Breaking True Dependences



Data Value Speculation

“Breaking” Dependences

- How to do so in a practical and efficient manner?
  - within single chip?
  - in a larger-scale system?
- What is the impact on other aspects of computing (e.g.., algorithms, compilers)?

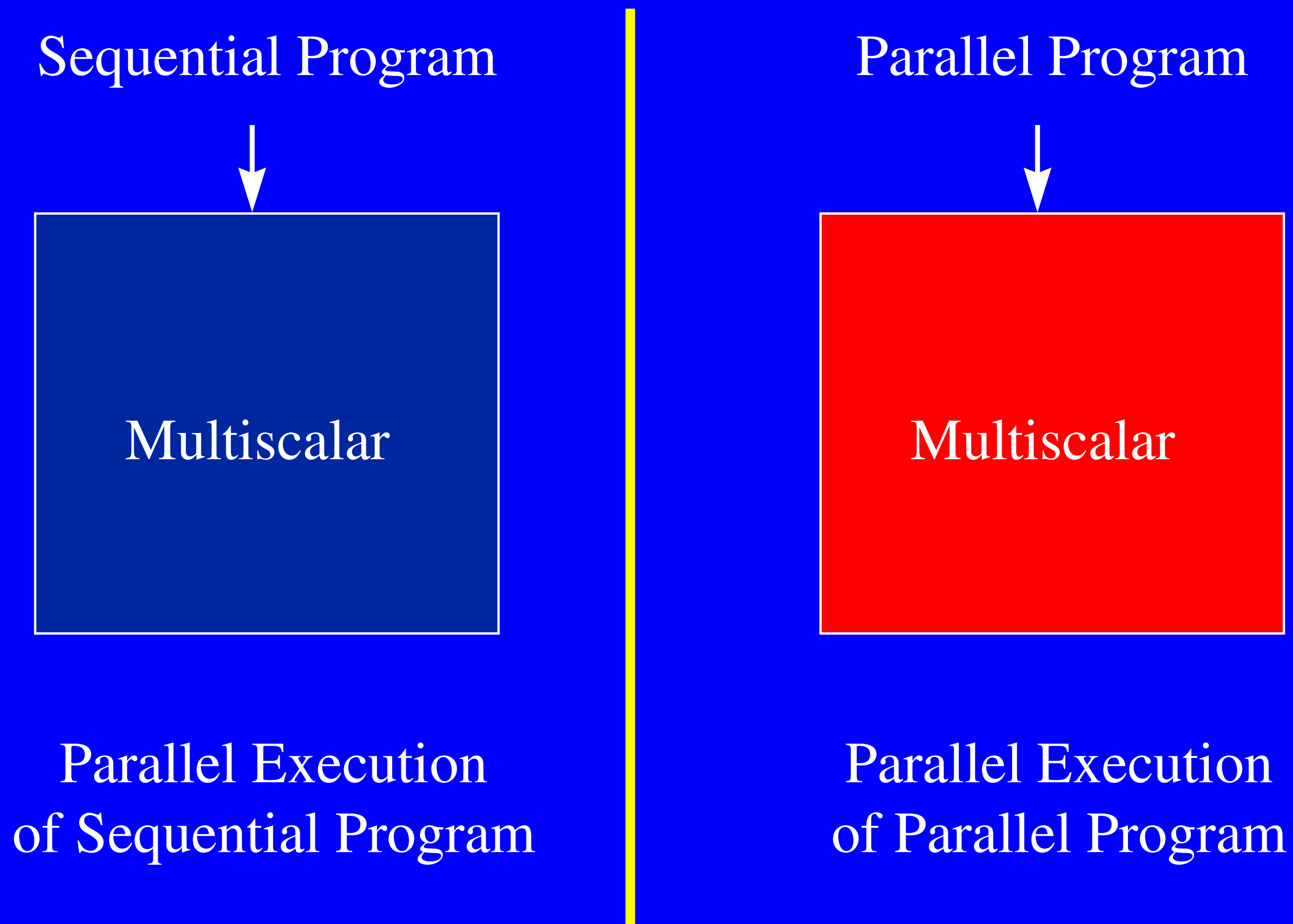
Multiscalar Goals

- Break traditional architecture compact in single-chip environment
  - execute "sequential" program in "parallel"
- Do so in a practical and efficient manner

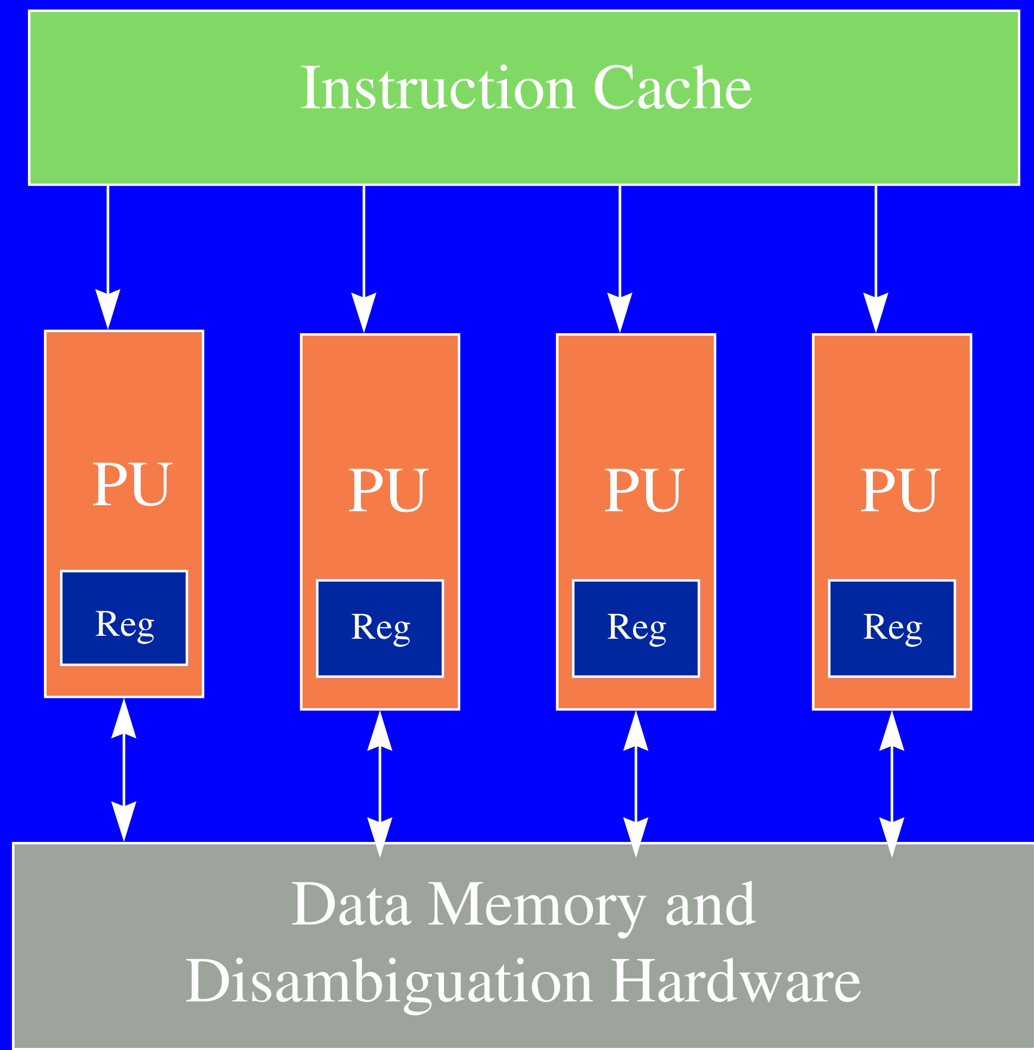
Important Practical Issues

- Fast clocks
  - On-chips wires/communication expensive
- Ease of design/validation
- Ease of test
- Varying power/performance spectrum

Multiscalar Microarchitecture



Multiscalar Microarchitecture



Kestrel Project

- Assess viability and potential of Multiscalar paradigm
- Hardware/microarch design and simulation
  - clock level
  - Verilog/Synopsys gate-level
  - circuit level
- Integrated compilers
  - front-end SUIF, back-end GCC

Kestrel Project Status

- Can execute arbitrary C/C++ programs on execution-driven clock-level simulator
- Verilog model on track
  - poster session
- Front-end and back-end compiler integration on track

Summary

- Use available opportunity to rethink traditional architecture compact
- Develop methods that allow compact to be broken in single-chip environment
- Assess impact on other aspects of computing
- Consider use of novel methods for larger scale systems