Instructor:

Dr. Mainak
Chaudhuri

Instructor:

Dr. S. K. Aggarwal

Instructor:

Dr. Rajat Moona

Course Name: Program Optimization for
Multi-core Architecture
Department: Computer Science and
Engineering
IIT Kanpur

Next

The Lecture Contains:

- Mind-boggling Trends in Chip Industry

- Agenda

- Unpipelined Microprocessors

- Pipelining

- Pipelining Hazards

- Control Dependence

- Data Dependence

- Structural Hazard

- Out-of-order Execution

- Multiple Issue

- Out-of-Order Multiple Issue

- Moore's Law

◀▌▌ Previous   Next ▌▌▶

Mind-boggling Trends in Chip Industry

- Long history since 1971
  - Introduction of Intel 4004
  - http://www.intel4004.com/
- Today we talk about more than one billion transistors on a chip
  - Intel Montecito (in market since July'06) has 1.7B transistors
  - Die size has increased steadily (what is a die?)
    - **Intel Prescott:** 112mm 2 , Intel Pentium 4EE: 237 mm 2 , Intel Montecito: 596 mm 2
  - Minimum feature size has shrunk from 10 micron in 1971 to 0.045 micron today

Agenda

- Unpipelined microprocessors
- **Pipelining:** simplest form of ILP
- **Out-of-order execution:** more ILP
- **Multiple issue:** drink more ILP
- Scaling issues and Moore's Law
- Why multi-core
  - TLP and de-centralized design
- Tiled CMP and shared cache
- Implications on software
- Research directions

Previous   Next

Unpipelined Microprocessors

- Typically an instruction enjoys five phases in its life
    - Instruction fetch from memory
    - Instruction decode and operand register read
    - Execute
    - Data memory access
    - Register write
- Unpipelined execution would take a long single cycle or multiple short cycles
    - Only one instruction inside processor at any point in time

Pipelining

- One simple observation
    - Exactly one piece of hardware is active at any point in time
- Why not fetch a new instruction every cycle?
    - Five instructions in five different phases
    - Throughput increases five times (ideally)
- Bottom-line is
    - If consecutive instructions are independent, they can be processed in parallel
    - The first form of instruction-level parallelism (ILP)

◀║║Previous   Next ║║▶

Pipelining Hazards

- Instruction dependence limits achievable parallelism
  - Control and data dependence (aka hazards)
- Finite amount of hardware limits achievable parallelism
  - Structural hazards
- Control dependence
  - On average, every fifth instruction is a branch (coming from if-else, for, do-while,…)
  - Branches execute in the third phase
    - Introduces bubbles unless you are smart

Control Dependence

```
Branch      IF    ID    EX    MEM  WB
Instr. X          IF    ID    EX   MEM  WB
Instr. X                IF    ID   EX   ...
Target                        IF   ID   ...
```
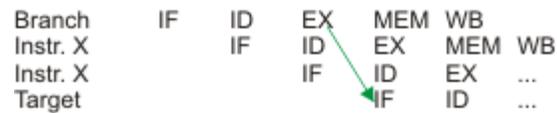
What do you fetch in X and Y slots?

Options: Nothing, fall-through, learn past history and predict (today best predictors achieve on average 97% accuracy for SPEC2000)

Data Dependence

◀|||**Previous**   **Next** |||▶

Take three bubbles?

Back-to-back dependence is too frequent

Solution: Hardware bypass paths

Allow the ALU to bypass the produced value in time: not always possible

Data Dependence

```
load   r1,add      IF    ID    EX    MEM  WB
add    r3,r2,r1           IF    ID    EX    MEM  WB

                              Reads Available      Value Needed
```
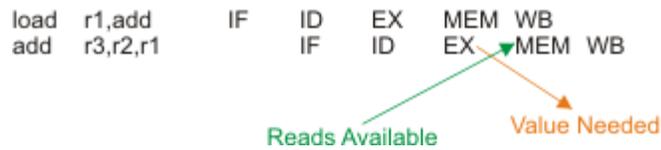
Need a live bypass! (requires some negative time travel: not yet feasible in real world)

No option but to take one bubble

Bigger Problems: load latency is often high; you may not find the data in cache

Structural Hazard

```
Instr. 1       IF    ID    EX    MEM  WB

Instr. 2             IF    ID    EX    MEM  WB
Instr. 3                   IF    ID    EX    MEM...

Instr. 4                         IF    ID    EX
...
```
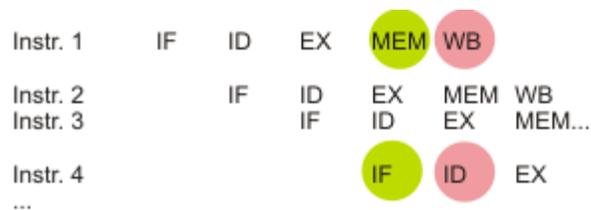
Usual solution is to put more resources

◀||| **Previous**   **Next** |||▶

Out-of-order Execution

Results must become visible in-order

Multiple Issue

Results must become visible in-order

Previous   Next

Out-of-order Multiple Issue

- Some hardware nightmares
    - Complex issue logic to discover independent instructions
    - Increased pressure on cache
        - Impact of a cache miss is much bigger now in terms of lost opportunity
        - Various speculative techniques are in place to "ignore" the slow and stupid memory
    - Increased impact of control dependence
        - Must feed the processor with multiple correct instructions every cycle
        - One cycle of bubble means lost opportunity of multiple instructions
    - Complex logic to verify

Moore's Law

- Number of transistors on-chip doubles every 18 months
    - So much of innovation was possible only because we had transistors
    - Phenomenal 58% performance growth every year
- Moore's Law is facing a danger today
    - Power consumption is too high when clocked at multi-GHz frequency and it is proportional to the number of switching transistors
- Wire delay doesn't decrease with transistor size

◀┃┃Previous   Next┃┃▶