Pipeline Control Hazards and Instruction Variations

Hakim Weatherspoon
CS 3410, Spring 2011
Computer Science
Cornell University

See P&H Appendix 4.8 & 2.16 and 2.17

Announcements

PA1 available: mini-MIPS processor

PA1 due next Friday

Work in pairs

Use your resources

- FAQ, class notes, book, Sections, office hours, newsgroup, CSUGLab

Prelims:

- Thursday, March 10th in class
- Thursday, April 28th Evening

Goals for Today

Recap: Data Hazards

- Data dependencies
- Problem, detection, and solutions

  (delaying, stalling, forwarding, bypass, etc)

Control Hazards

- What is the next instruction to execute if a branch is taken? Not taken?

Instruction Variations

- sdf

Data Hazard Recap

Delay Slot(s)

- Modify ISA to match implementation

Stall

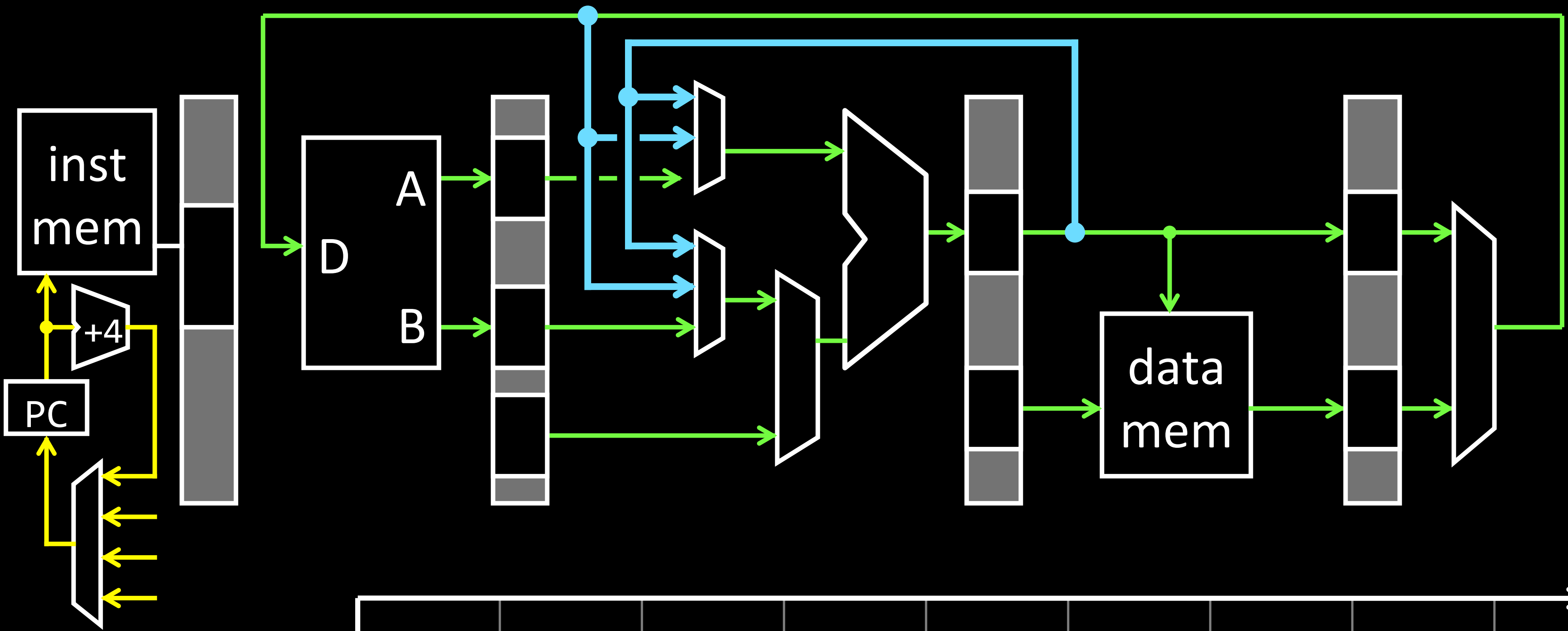
- Pause current and all subsequent instructions

Forward/Bypass

- Try to steal correct value from elsewhere in pipeline
- Otherwise, fall back to stalling or require a delay slot

Tradeoffs?

More Hazards



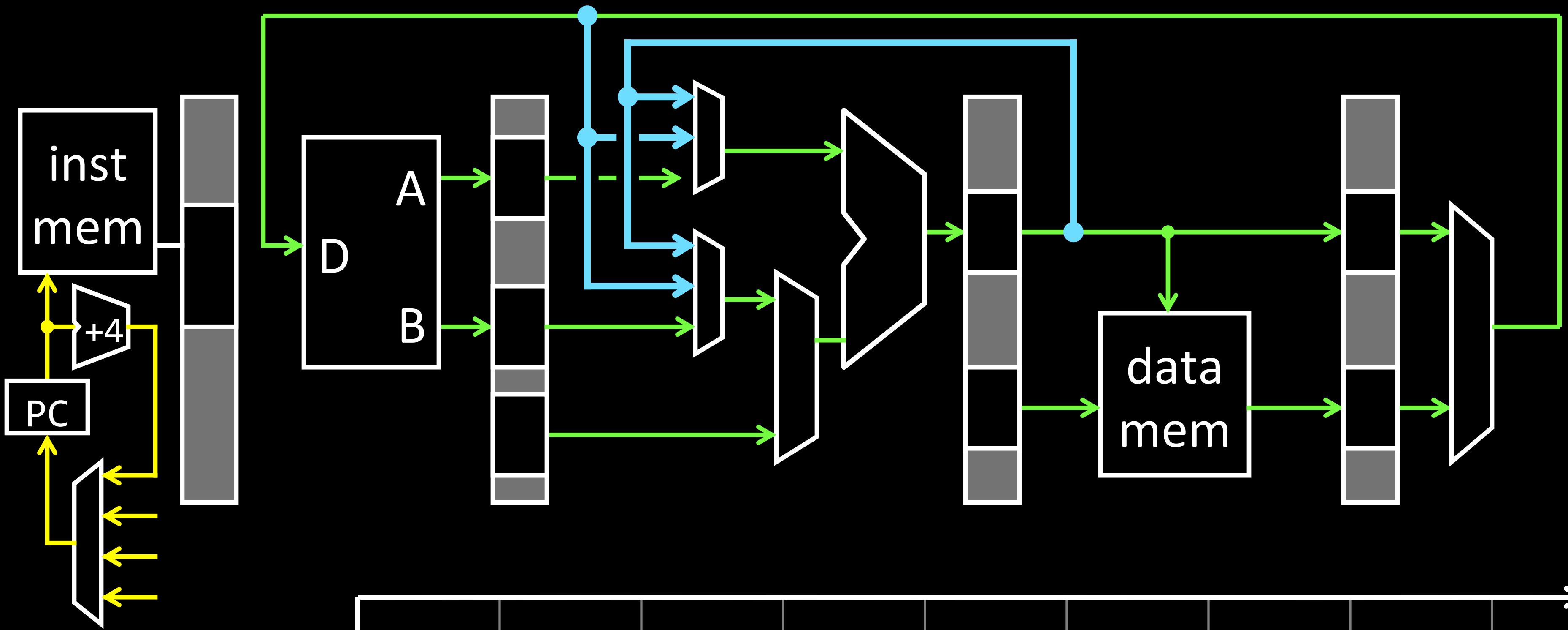
beq r1, r2, L

add r3, r0, r3

sub r5, r4, r6

L: or r3, r2, r4

More Hazards



beq r1, r2, L

add r3, r0, r3

sub r5, r4, r6

L: or r3, r2, r4

Control Hazards

Control Hazards

- instructions are fetched in stage 1 (IF)
- branch and jump decisions occur in stage 3 (EX)
- i.e. next PC is not known until 2 cycles after branch/jump

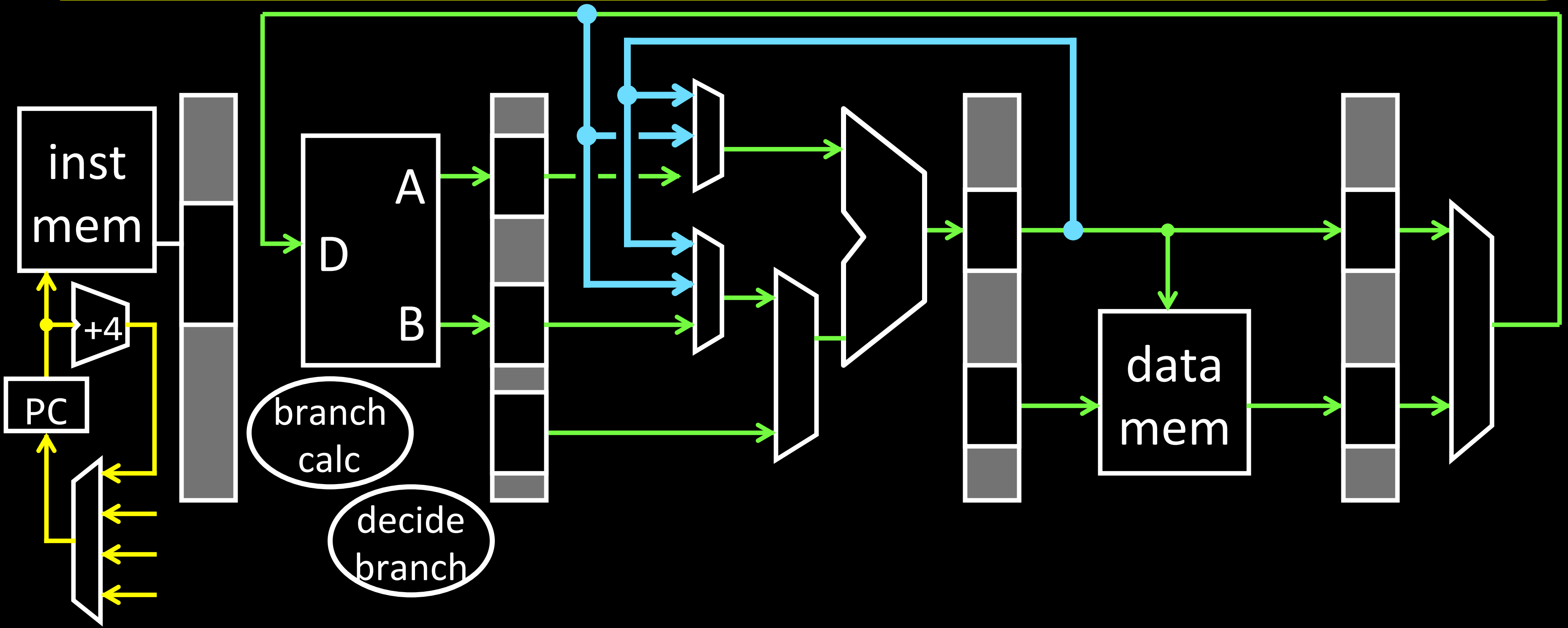
Delay Slot

- ISA says N instructions after branch/jump always executed
  - MIPS has 1 branch delay slot

Stall (+ Zap)

- prevent PC update
- clear IF/ID pipeline register
  - instruction just fetched might be wrong one, so convert to nop
- allow branch to continue into EX stage

Delay Slot



beq r1, r2, L									
ori r2, r0, 1									
L: or r3, r1, r4									

Control Hazards: Speculative Execution

Control Hazards

- instructions are fetched in stage 1 (IF)
- branch and jump decisions occur in stage 3 (EX)
- i.e. next PC not known until 2 cycles after branch/jump

Stall

Delay Slot

Speculative Execution

- Guess direction of the branch
  - Allow instructions to move through pipeline
  - Zap them later if wrong guess
- Useful for long pipelines

Loops

Branch Prediction

Pipelining: What Could Possibly Go Wrong?

Data hazards

- register file reads occur in stage 2 (IF)
- register file writes occur in stage 5 (WB)
- next instructions may read values soon to be written

Control hazards

- branch instruction may change the PC in stage 3 (EX)
- next instructions have already started executing

Structural hazards

- resource contention
- so far: impossible because of ISA and pipeline design

Variations

More shifts

Variations

Condition flags

Variations

Condition flags

Variations

Conditional instructions

```
top:  CMP r3, r4
      SUBGT r3, r3, r4
      SUBLT r4, r4, r3
      BNE top
```

Variations

Stack operations & multiple destinations

Variations

String instructions

```
MOV AX, FFh
MOV DI, 5000h
MOV CX, 2000h
REP STOSB
```

Variations

Vector instructions

vaddubs v3, v1, v2