LECTURE - 10

Data Hazards

	CC1	CC2	CC3	CC4	CC5	CC6	CC7	CC8	CC9	CC10	CC11	CC12
LD F4, 0(R2)	IF	ID	EX	MEM	WB							
MULTD F0, F4, F6		IF	ID	STL	M1	M2	M3	M4	M5	M6	M7	MEM

RAW hazards cause more stalls now

	CC1	CC2	CC3	CC4	CC5	CC6	CC7	CC8	CC9	CC10	CC11
MULTD F0, F4, F6	IF	ID	M1	M2	M3	M4	M5	M6	M7	MEM	**WB**
...		IF	ID	EX	MEM	WB					
...			IF	ID	EX	MEM	WB				
ADDD F2, F4, F6				IF	ID	A1	A2	A3	A4	MEM	**WB**
...					IF	ID	EX	MEM	WB		
LD F2, 0(R2)						IF	ID	EX	MEM	**WB**	

WAW hazard: an example

Handling WAW Hazards

- Occurs only when the result of ADDD is overwritten without any instruction using it!
  - Otherwise, RAW hazard stall would have occurred
- Hazard can be detected in ID stage of latter instruction
- Two ways to handle:
  - Delay issue of load until ADDD enters MEM
  - Stamp out result of ADDD

Control Hazard Complications

- An example:
  - DIVF F0, F2, F4 // Finishes last; excepn.
  - ADDF F10, F10, F8 // Finishes first
  - SUBF F12, F12, F14 // Finishes second
- Out-of-order completion causes problems!
  - Precise exceptions are difficult to implement

Achieving Precise Exceptions

- **Approach 1:** Ostrich algorithm
  - Don't care
  - May be provide a slower precise mode
    - Example: special instructions to check for FP exceptions
- **Approach 2:** allow instruction issue to continue only if previous instructions will complete without exception
  - Stall to maintain precise exceptions

Achieving Precise Exceptions (continued)

- **Approach 3:** save state to undo
  - Two possibilities
    - History file: keep track of original value of registers
    - Future file: keep track of current value; main register file updated after all previous instructions are done
  - More buffer space required
  - Hazard checks and control become very complex

Achieving Precise Exceptions (continued)

- **Approach 4:** imprecise, but keep enough state for OS to recover
  - Keep track of incomplete instructions
  - OS then runs those instructions before returning control
  - Complicated to execute these instructions properly!

Further Topics…

- Instruction Level Parallelism (ILP)