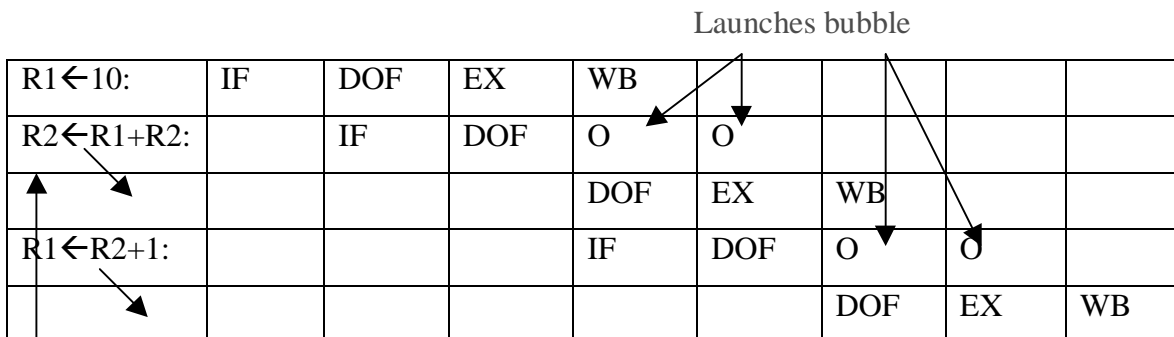


Hazards

- If an instruction tries to access a register that will be modified by an instruction in the pipeline
 - o This is a “data hazard”
 - o Ex
 - 1. $R2 \leftarrow 10$
 - 2. $R2 \leftarrow R1 + R2$
 - o note: in the example architecture, the WB and DOF can be done in the same cycle
 - if there was an instruction between the ones above, no hazard
 - a hazard can also occur when the programmer requests a conditional branch
 - o ex
 - 1. BRZset
 - ADD
 - o Instruction 2 would be inserted into the pipeline before 1 is finished
 - o If the branch is taken, it shouldn't be executed
 - o A “control hazard”
 - Possible solution 1: Do nothing
 - o Make the programmer/compiler figure it out
 - o No change in the circuits, just warn the programmer
 - o Need to add a NOP (no operation) instruction
 - o EX
 - 1. $R1 \leftarrow 10$ $R1 \leftarrow 10$
 - 2. $R2 \leftarrow R1 + R2$ hazard NOP
 - 3. $R1 \leftarrow R2 + 1$ no hazard $R2 \leftarrow R1 + R2$
 - 4. no hazard NOP
 - 5. $R1 \leftarrow R2 + 1$
 - o this requires the programmer/compiler to know a lot about the underlying architecture
 - o makes it hard to program
 - o no chance of backward compatibility with older code
- Possible solution 2: stall
 - o Have the processor delay an instruction when necessary
 - o When the processor notice a hazard it launches a “bubble”
 - I.e. keeps the instruction from moving down the pipe

- E.g.



Notice hazard

- Designing a control unit to do this is tricky
 - Must check DR of instruction in the pipe
 - If they correspond to an operand in the pipe (current instruction), stall
 - To stall: in the next cycle, send the same instruction through the pipe
 - Don't increment PC or update IR
 - We must make sure the bubble has RW=MW=0
 - So it doesn't have any effect
 - For a control hazard, we don't want to always do the next instruction
 - More later
- Possible Solution 3: data forwarding (for data hazards only)
 - If the data is already in the pipeline somewhere use it from there
 - E.g.

R1 ← 10:	IF	DOF	EX	WB	
R2 ← R1+R2		IF	DOF	EX	WB

Get the RI value from the function unit, not reg. file