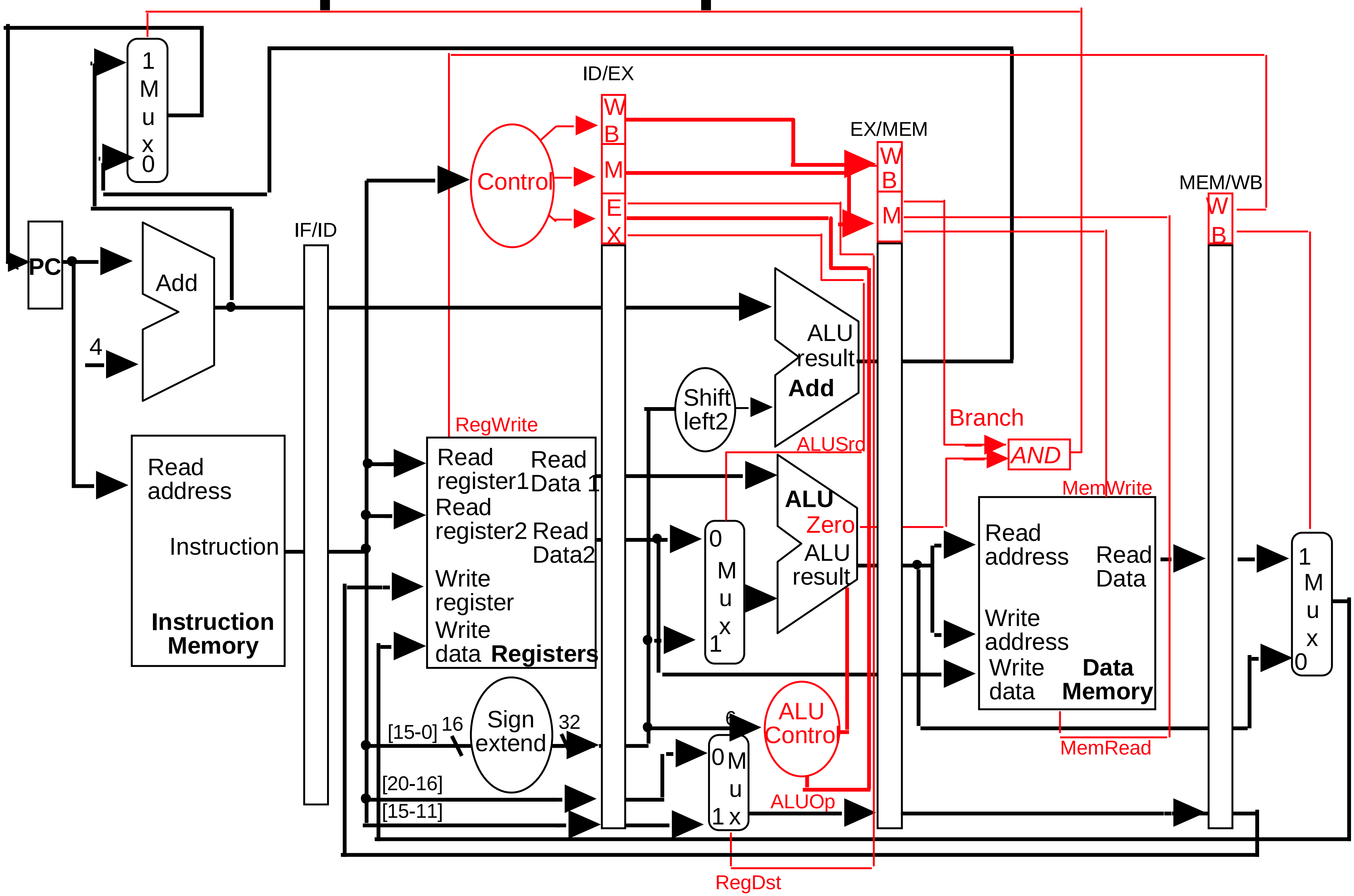
Control of Pipelined Datapaths

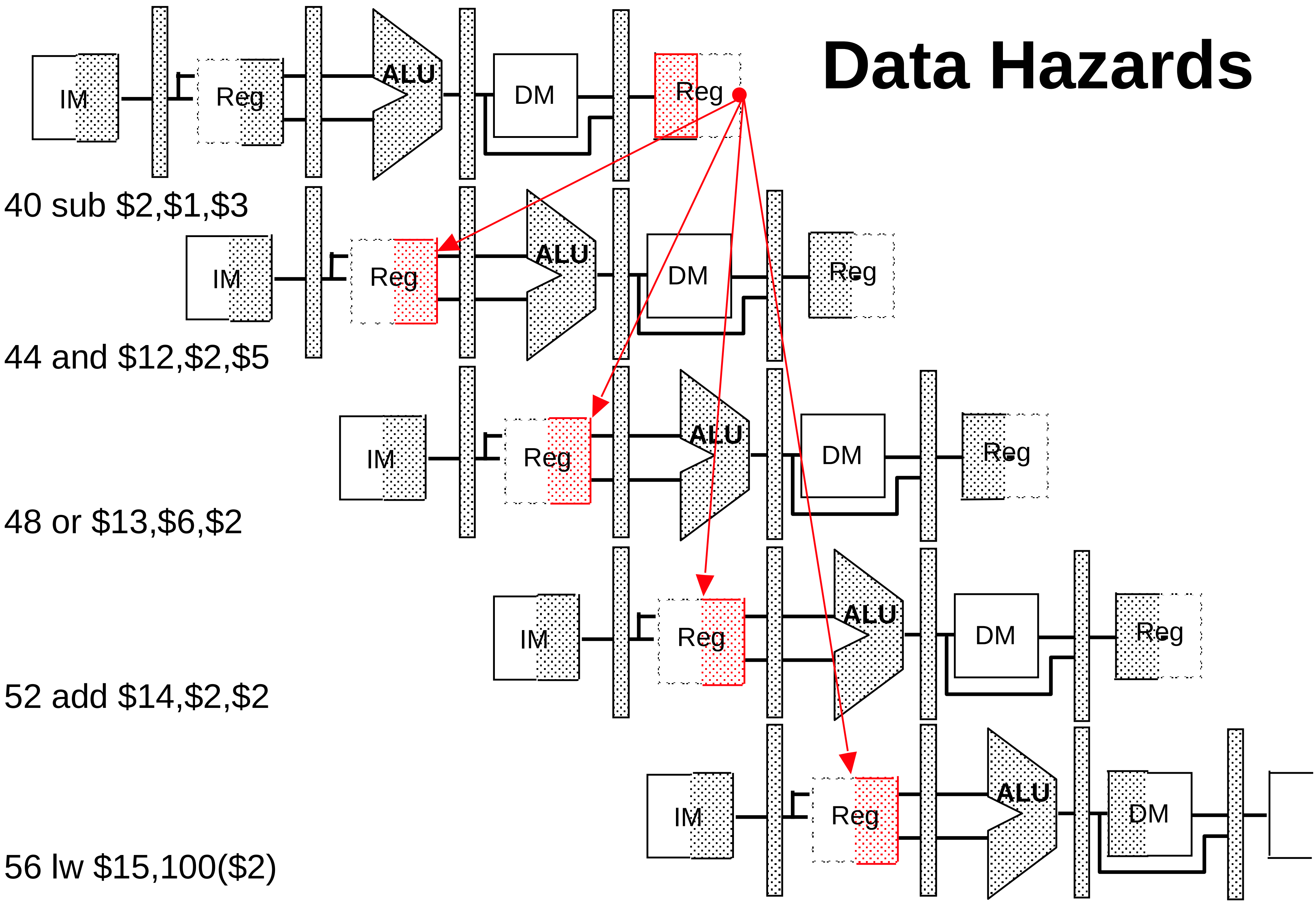
The datapath is controlled by forwarding the control information along the pipeline in the pipeline registers.

Datapath control is complicated by data hazards, the reading of register values before they have the proper value written.

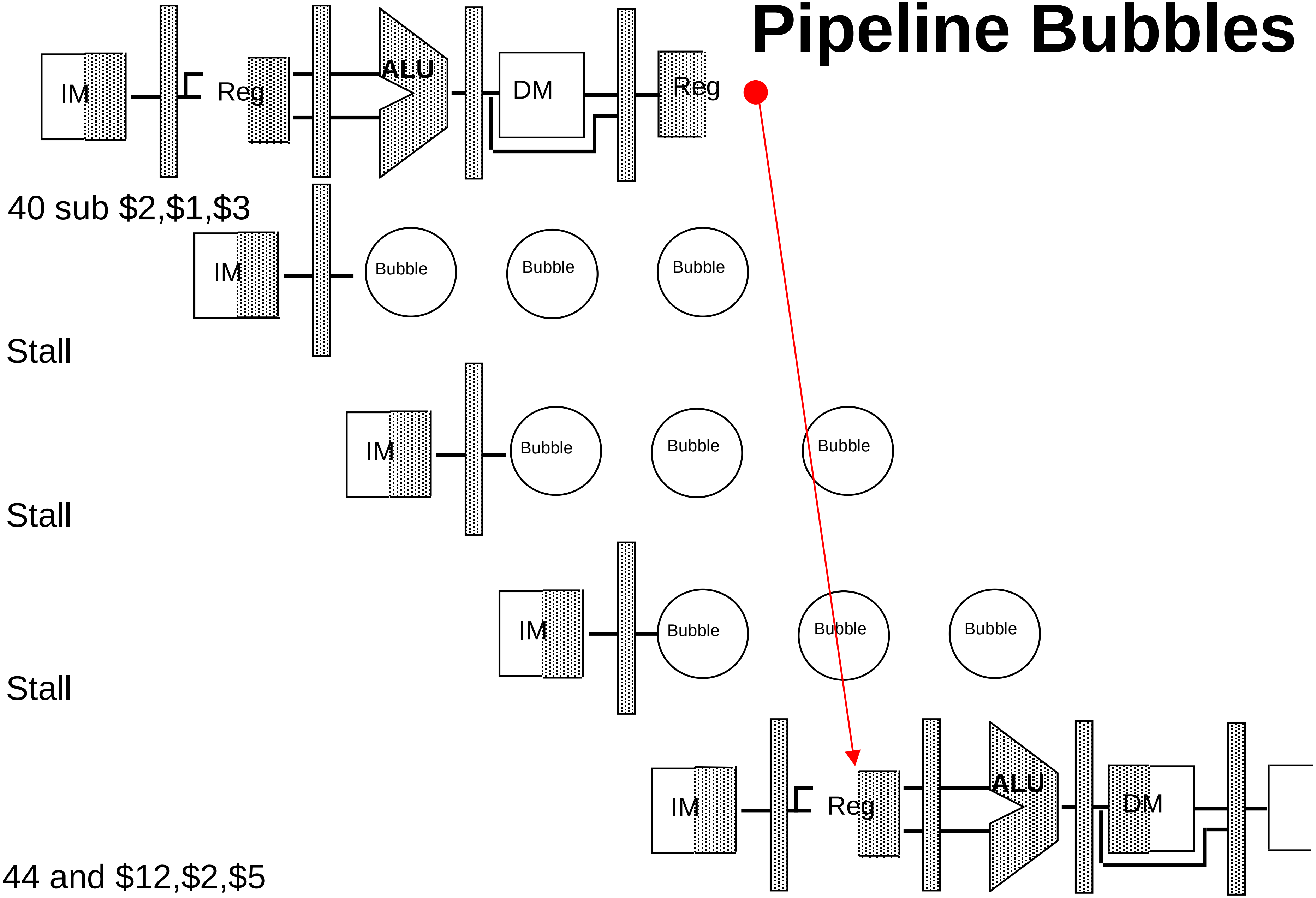
Pipeline Datapath Control



Data Hazards



Pipeline Bubbles



Detecting Data Hazards

EX Hazard:

if (EX/MEM.RegWrite
and (EX/MEM.RegisterRd != 0)
and (EX/MEM.RegisterRd = ID/EX.RegisterRs))

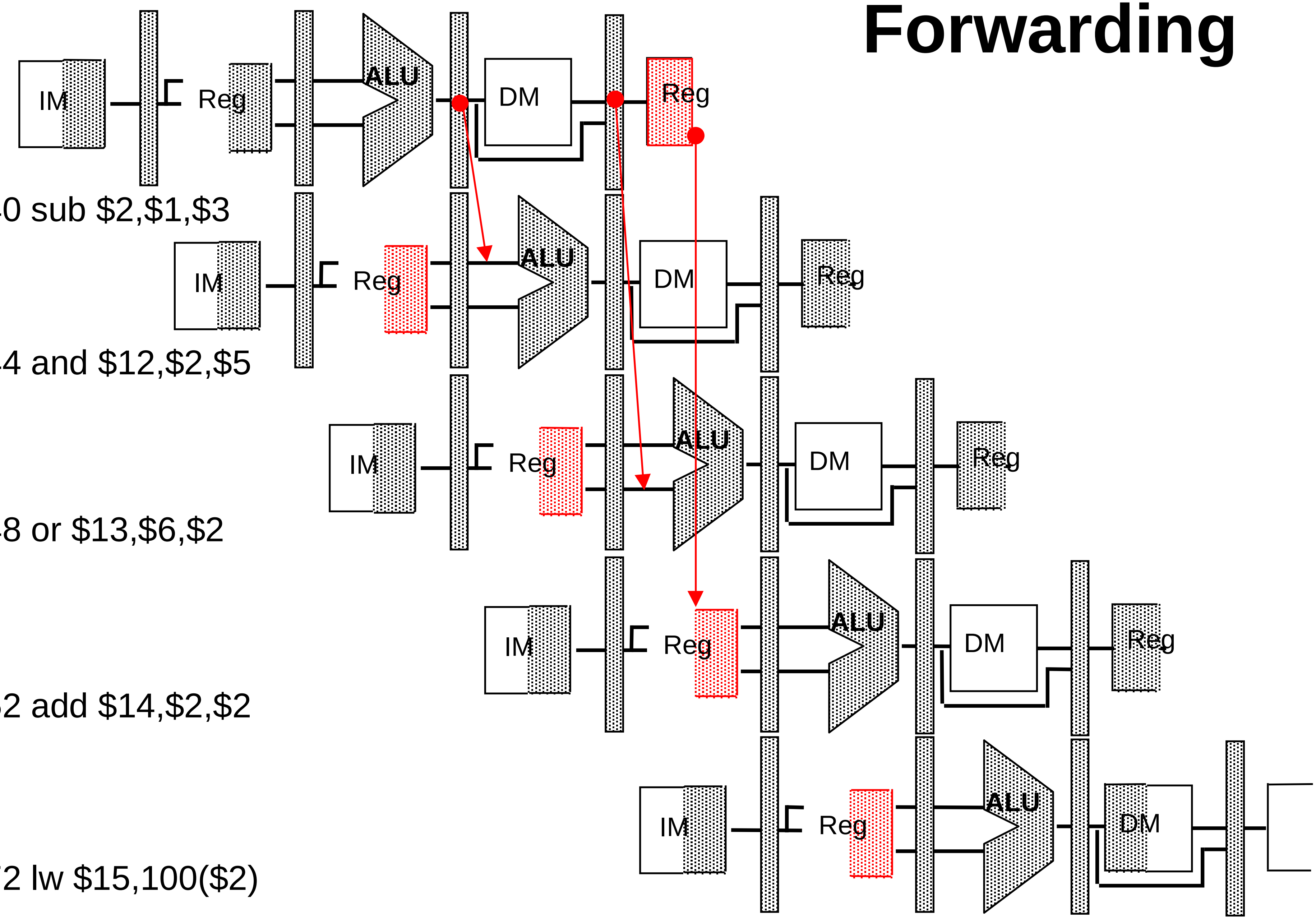
Corresponding rule for ID/EX.RegisterRt

MEM Hazard:

if (MEM/WB.RegWrite
and (MEM/WB.RegisterRd = ID/EX.RegisterRs)
and (MEM/WB.RegisterRd != 0)
and (EX/MEM.RegisterRd != ID/EX.RegisterRs)

Corresponding rule for ID/EX.RegisterRt

Forwarding



40 sub $2,$1,$3

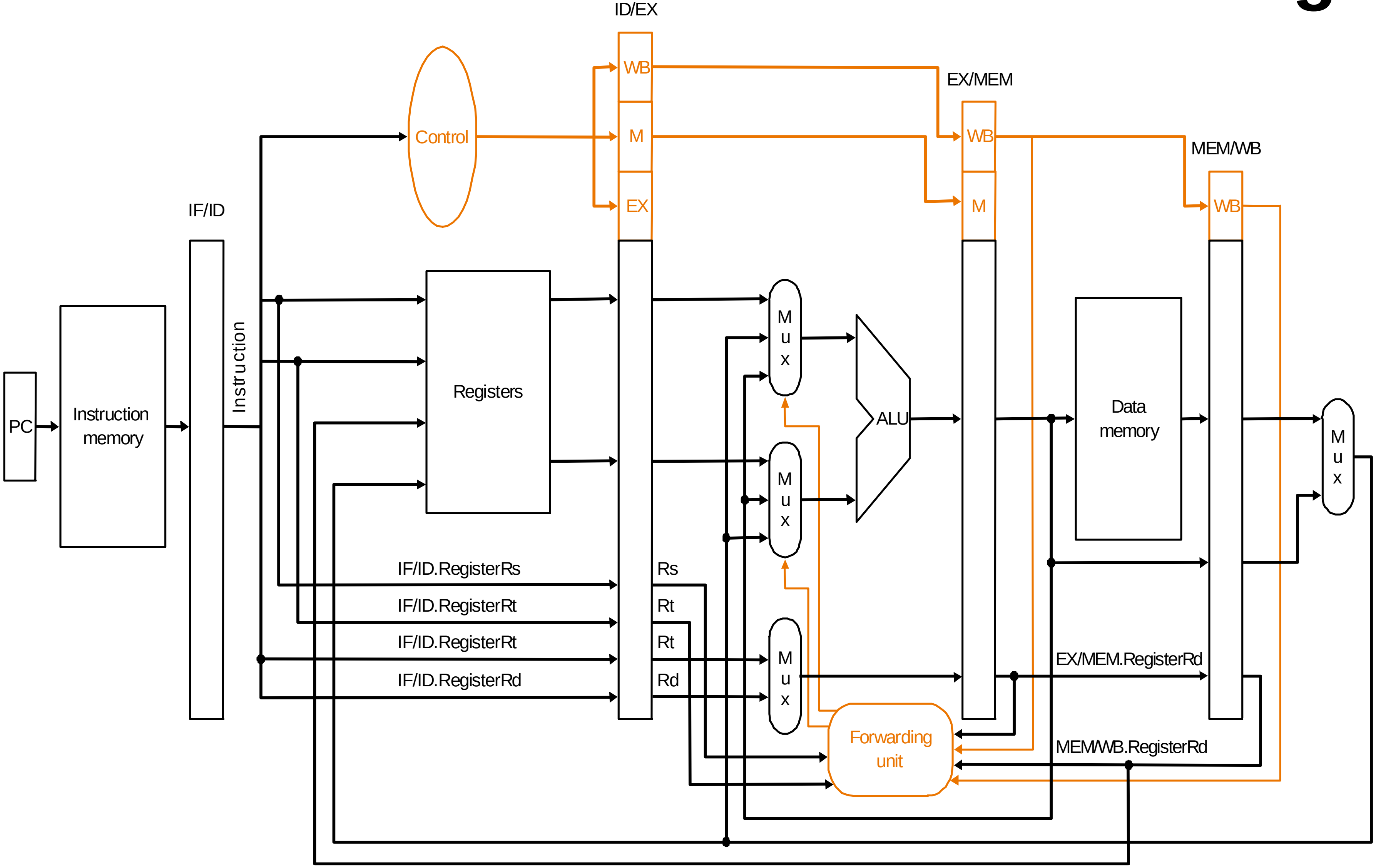
44 and $12,$2,$5

48 or $13,$6,$2

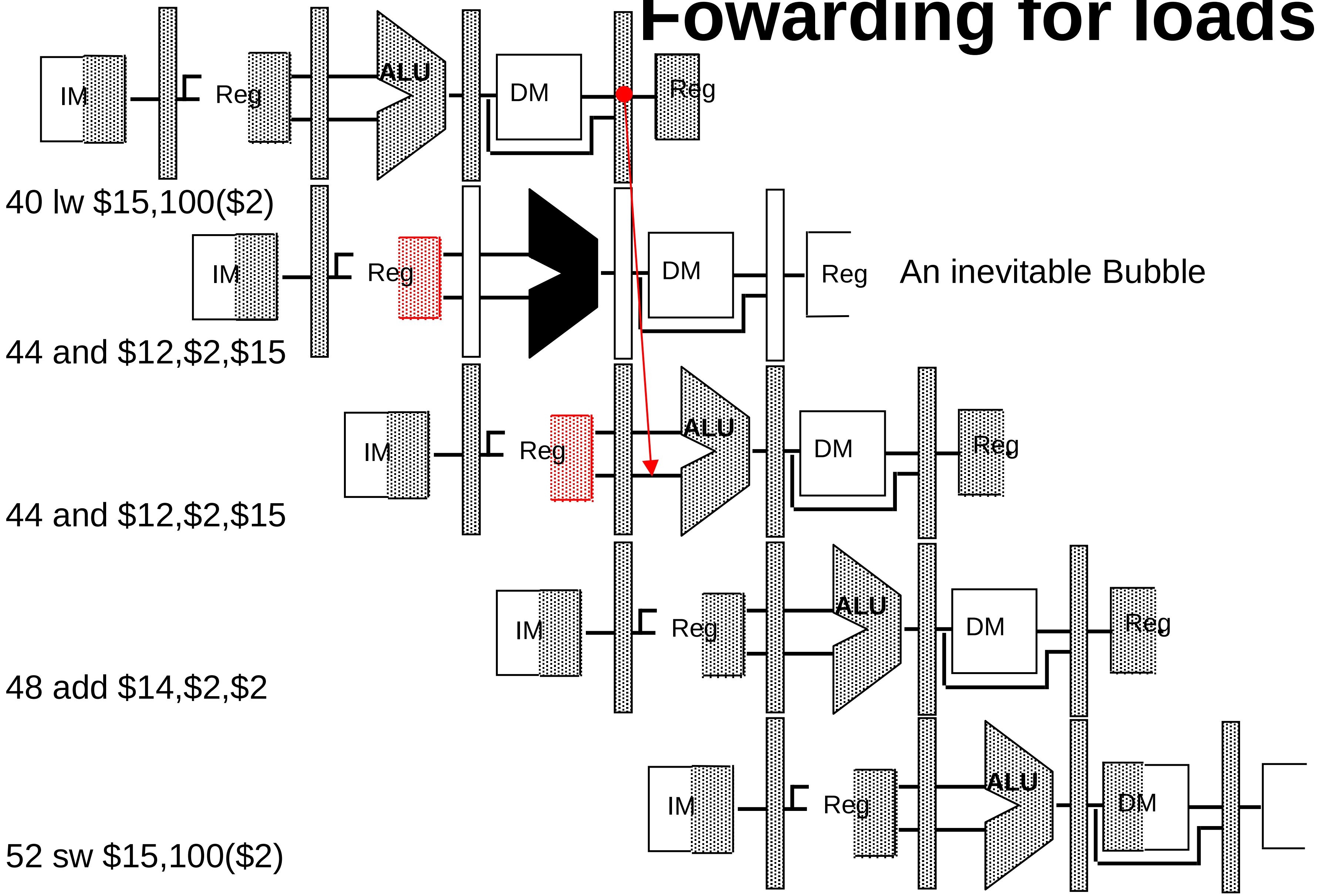
52 add $14,$2,$2

72 lw $15,100($2)

Forwarding


ID/EX
WB
M
EX
Control
EX/MEM
WB
M
MEM/WB
WB
IF/ID
PC
Instruction memory
Instruction
Registers
M u x
ALU
M u x
Data memory
M u x
IF/ID.RegisterRs
IF/ID.RegisterRt
IF/ID.RegisterRt
IF/ID.RegisterRd
Rs
Rt
Rt
Rd
M u x
EX/MEM.RegisterRd
Forwarding unit
MEM/WB.RegisterRd

Fowarding for loads



Load Hazards

Delay Slots vs Hazard Detection

If (ID/EX.MemRead and
((ID/EX.RegisterRt = IF/ID.RegisterRs) or
(ID/EX.RegisterRt = IF/ID.RegisterRt)))

Consider the sequence to swap stack items

```
lw   $10, 0($29)   -- Load top item
lw   $11, 4($29)   -- Load second item
sw   $11, 0($29)   -- Replace top item
sw   $10, 4($29)   -- Replace second item
```

Which can be replaced by

```
lw   $10, 0($29)   -- Load top item
lw   $11, 4($29)   -- Load second item
sw   $10, 4($29)   -- Replace second item
sw   $11, 0($29)   -- Replace top item
```

Load Hazards