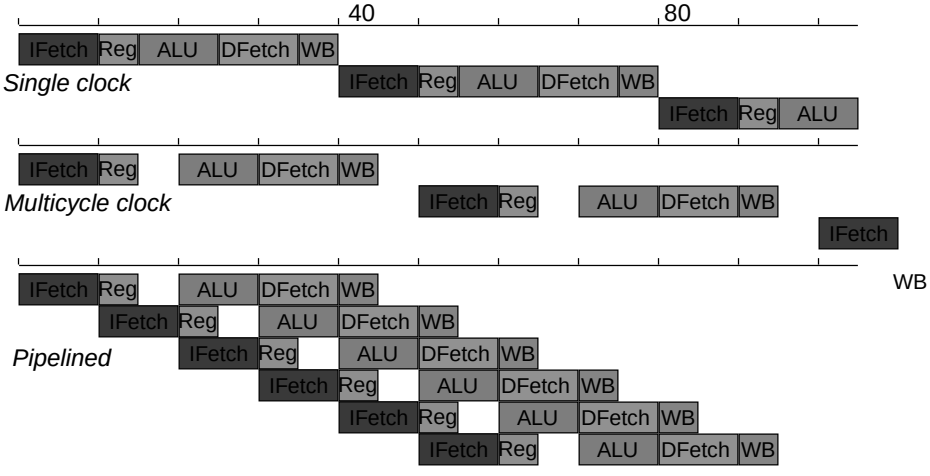
Pipelining

Pipelining is a standard technique for exploiting parallelism in tasks that are ordered. Two considerations are: Balancing the step size and avoiding hazards

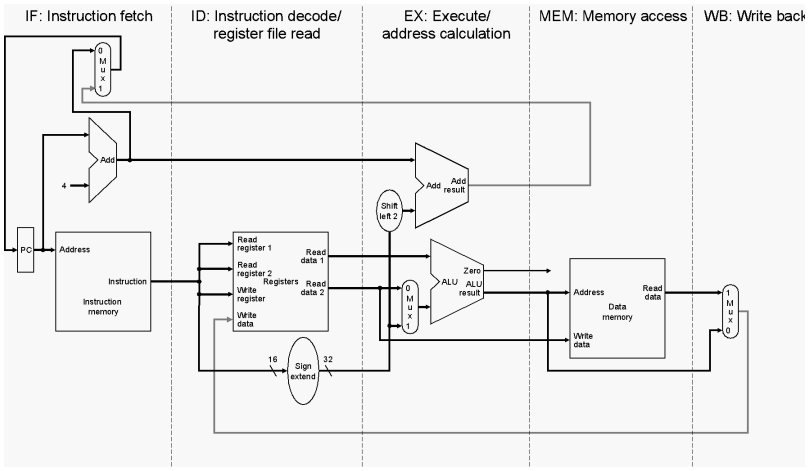
Benefits of Pipelining

Assume:
Memory(10ns), ALU(10ns), Register(5ns),
All other operations are 0ns.

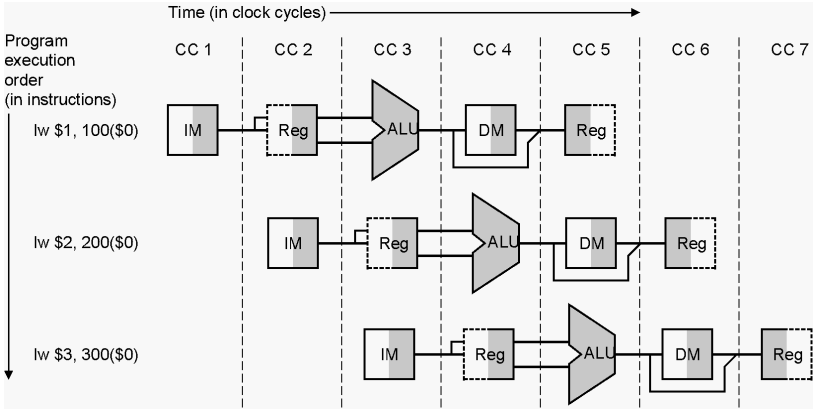
Charges for instructions are ...
- –R-format: 30ns
- –Load inst: 40ns
- –Store inst: 35ns
- –Branch: 25ns
- –Jump: 10ns



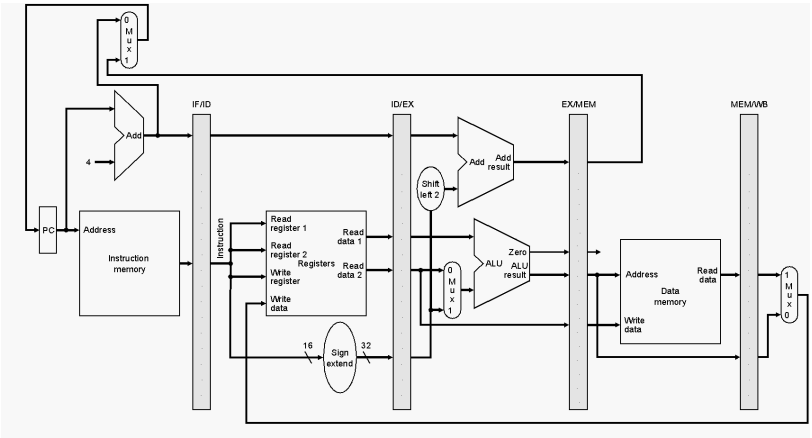
Decomposing the Fetch/Execute Cycle



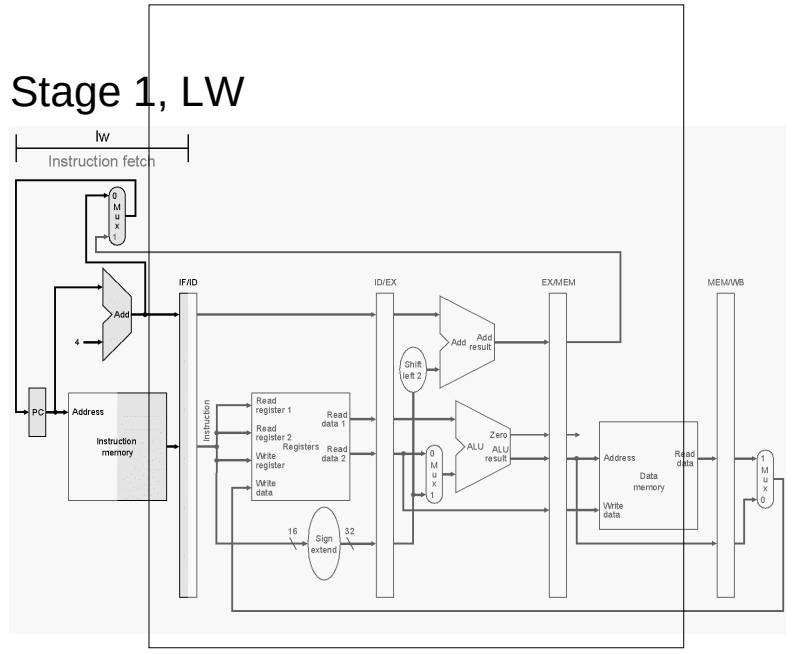
Abstracting the Pipeline



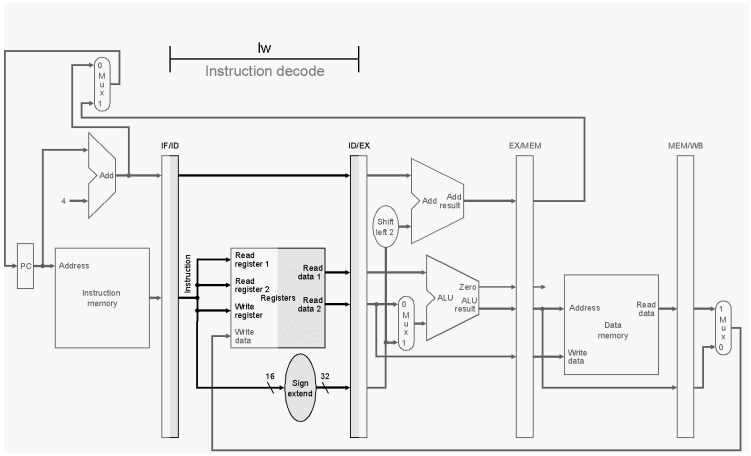
Keeping State In The Pipeline



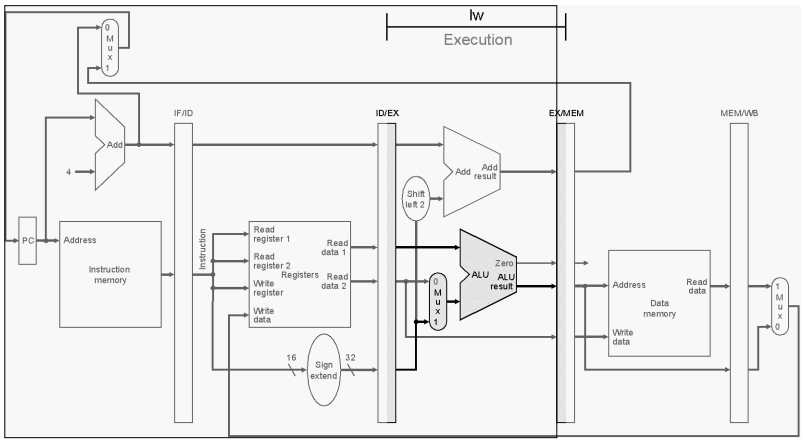
Stage 1, LW



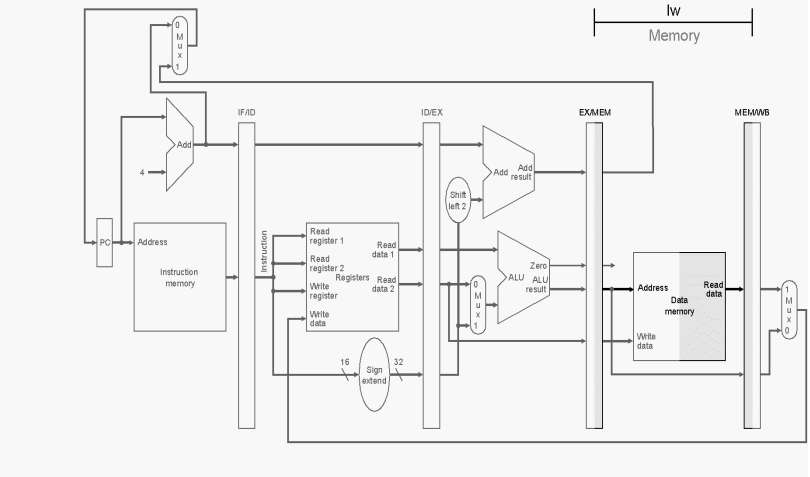
Stage 2, LW

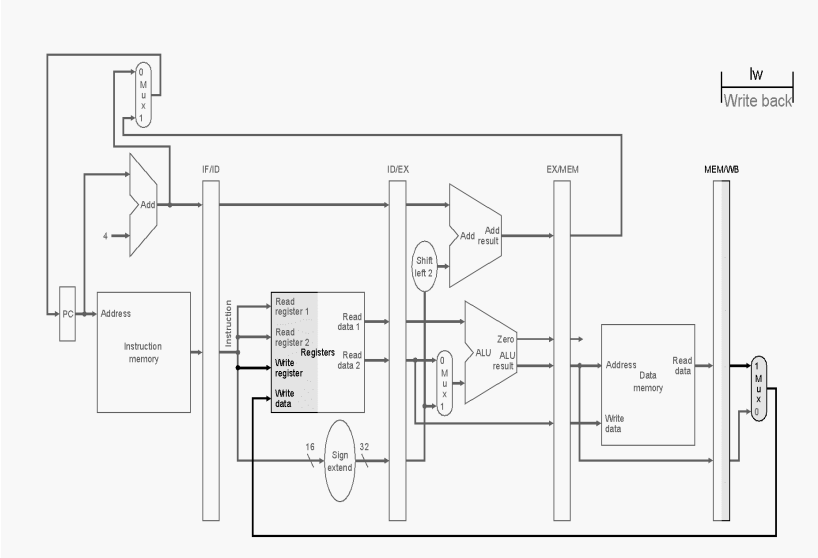


Stage 3, LW



Stage 4, LW


lw
Memory
IF/ID
ID/EX
EX/MEM
MEM/WB
Add
4
PC
Address
Instruction memory
Instruction
Read register 1
Read register 2
Registers
Write register
Write data
Read data 1
Read data 2
Shift left 2
Add
Add result
Zero
ALU
ALU result
Address
Data memory
Read data
Write data
16
Sign extend
32
M u x


lw
Write back
IF/ID
ID/EX
EX/MEM
MEM/WB
Add
4
PC
Address
Instruction memory
Instruction
Read register 1
Read register 2
Registers
Write register
Write data
Read data 1
Read data 2
Shift left 2
Add
Add result
Zero
ALU
ALU result
Address
Data memory
Read data
Write data
16
Sign extend
32
M u x

LW Write Back