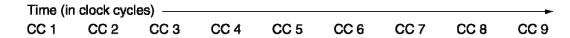
# Chapter 4

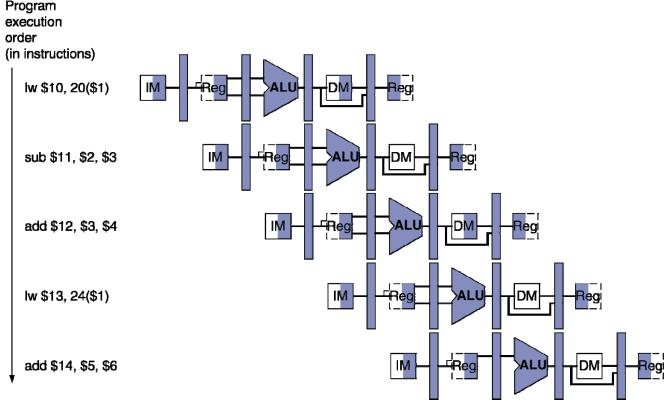
The Processor



### Multi-Cycle Pipeline Diagram

### Form showing resource usage







## Multi-Cycle Pipeline Diagram

#### Traditional form

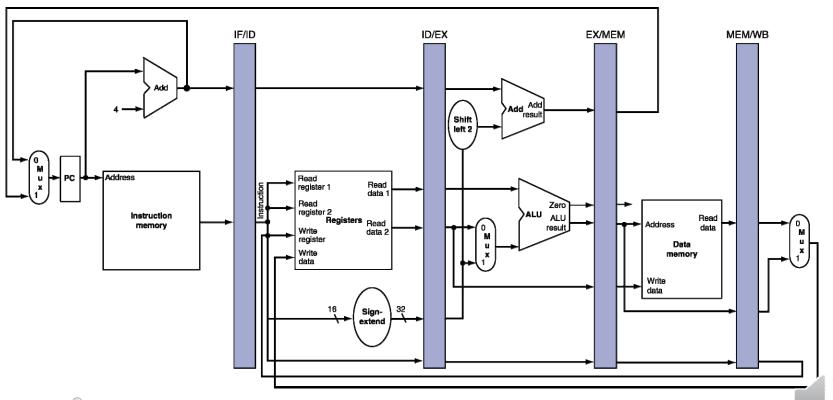
Time (in clock cycles) CC<sub>1</sub> CC 2 CC 3 CC 4 CC<sub>5</sub> CC<sub>6</sub> CC 7 CC8 CC9 Program execution order (in instructions) Instruction Instruction Data Execution lw \$10, 20(\$1) Write back fetch decode access Instruction Instruction Data sub \$11, \$2, \$3 Execution Write back fetch decode access Instruction Instruction Data add \$12, \$3, \$4 Execution Write back decode access fetch Instruction Data Instruction lw \$13, 24(\$1) Execution Write back fetch decode access Instruction Data Instruction add \$14, \$5, \$6 Write back Execution fetch decode access



# Single-Cycle Pipeline Diagram

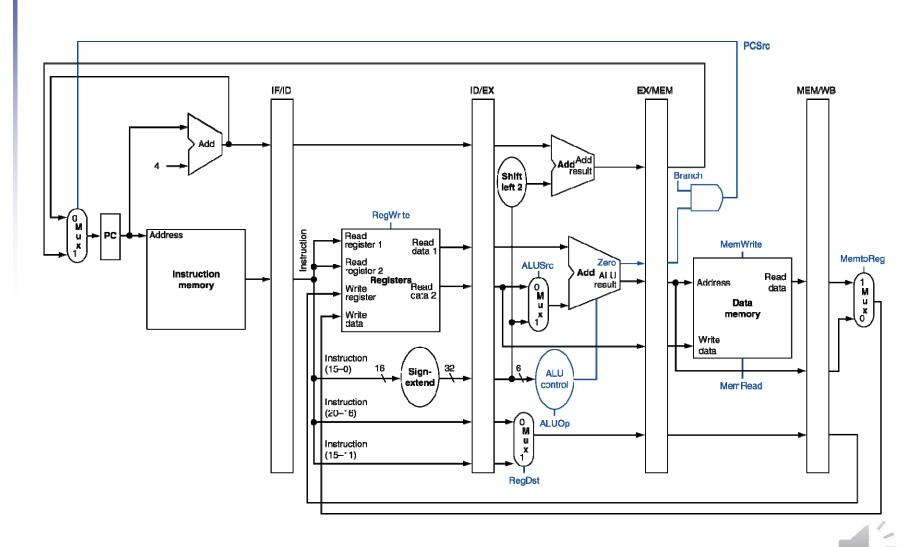
#### State of pipeline in a given cycle

add \$14, \$5, \$6	lw \$13, 24 (\$1)	add \$12, \$3, \$4	sub \$11, \$2, \$3	lw \$10, 20(\$1)
Instruction fetch	Instruction decode	Execution	Memory	Write-back





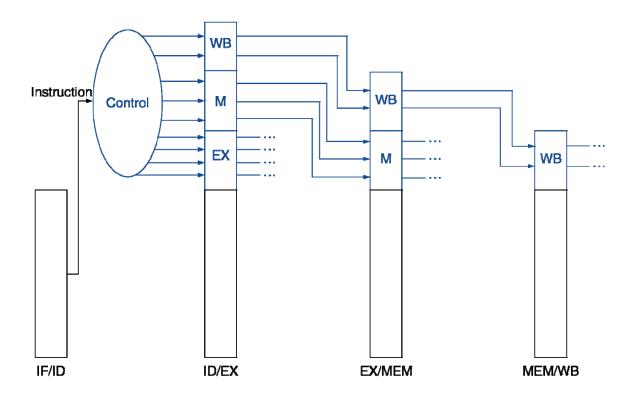
# **Pipelined Control (Simplified)**





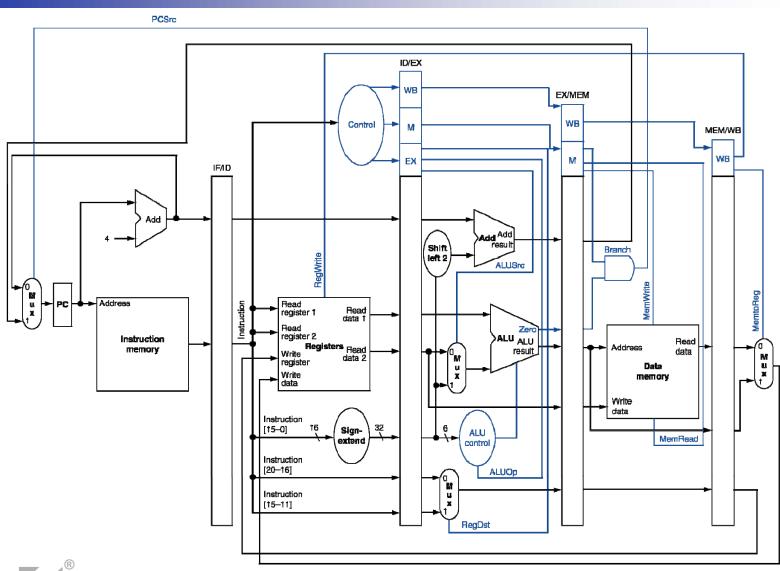
### **Pipelined Control**

- Control signals derived from instruction
  - As in single-cycle implementation





# **Pipelined Control**





# **Pipelined Control Signals**

	Execution Stage Control Lines				Memory Stage Control Lines			Write Back Stage Control Lines	
Instruction	RegDst	ALU Op1	ALU Op0	ALUSrc	Branch	Mem Read	Mem Write	RegWrite	MemtoReg
R-Format	1	1	0	0	0	0	0	1	0
lw	0	0	0	1	0	1	0	1	1
sw	х	0	0	1	0	0	1	0	Х
beq	Х	0	1	0	1	0	0	0	Х



