

Fundamentals of Computer Systems

A Multicycle MIPS Processor

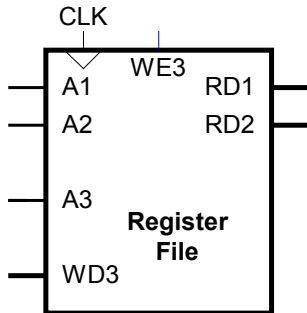
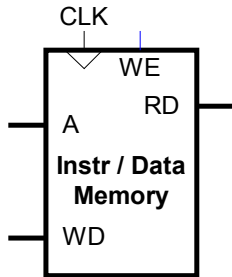
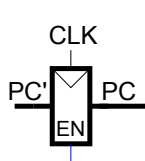
Stephen A. Edwards

Columbia University

Summer 2016

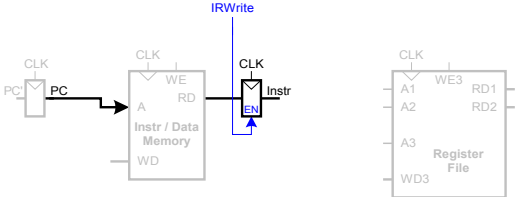
Illustrations Copyright © 2007 Elsevier

State Elements



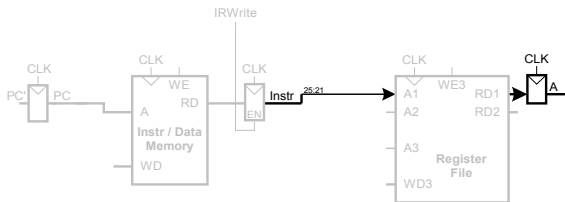
Multicycle Datapath

Fetch instruction from memory



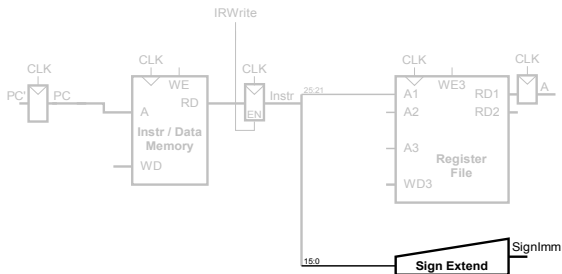
Multicycle Datapath

Read source operands from register file



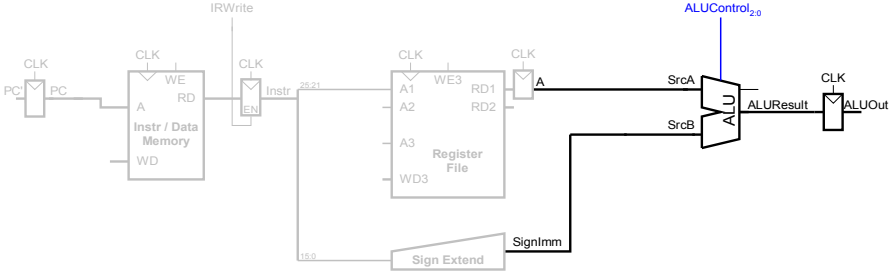
Multicycle Datapath

Sign-extend the immediate



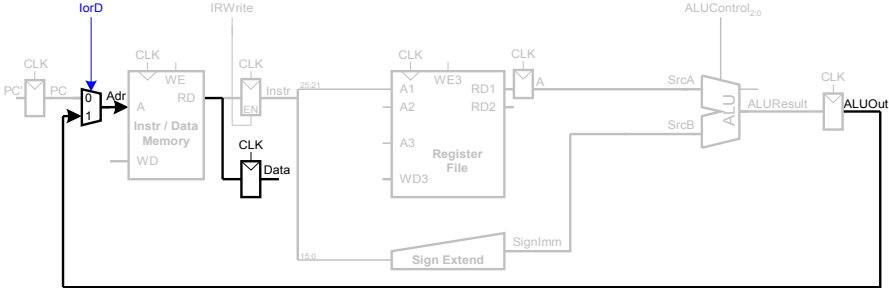
Multicycle Datapath

Add base address to offset



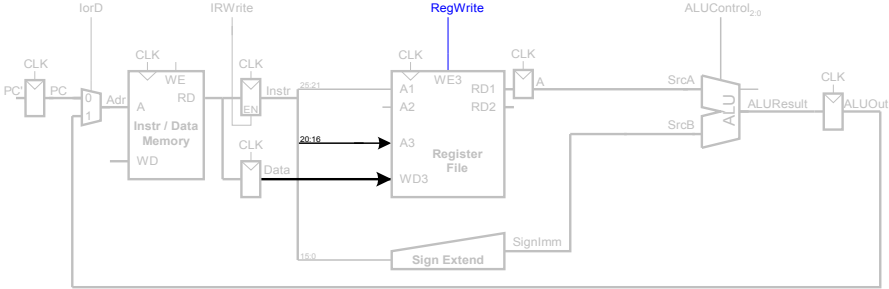
Multicycle Datapath

Load data from memory



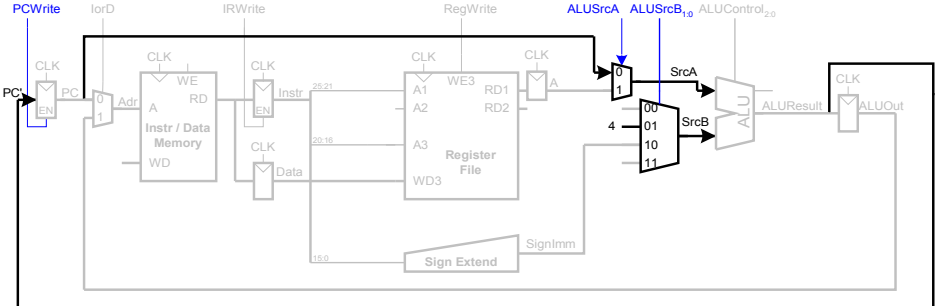
Multicycle Datapath

Write data back to register file



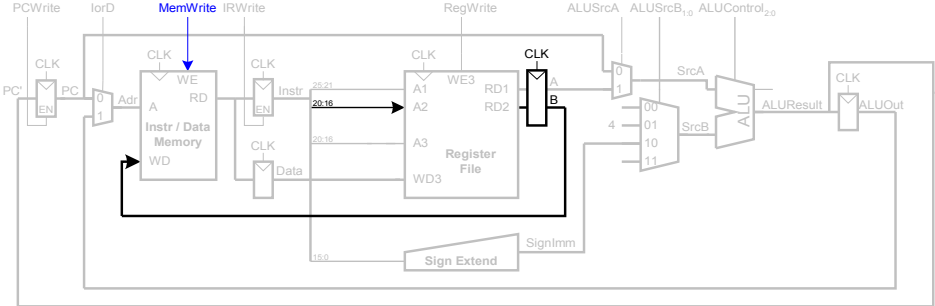
Multicycle Datapath

Add 4 to PC



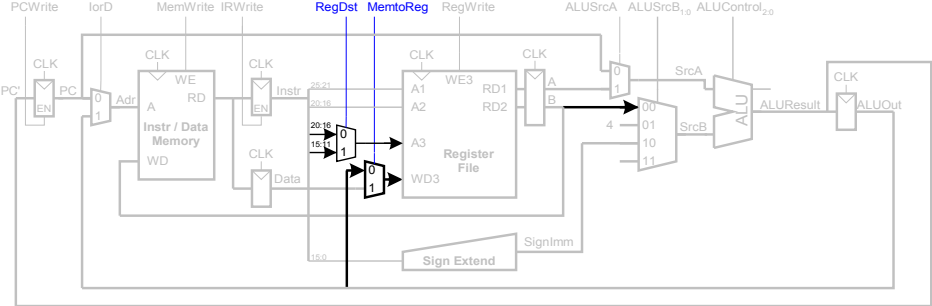
Multicycle Datapath

For sw: Write register data to memory



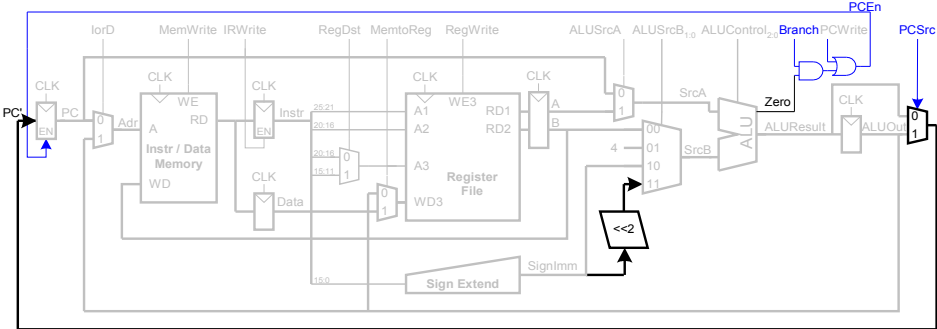
Multicycle Datapath

For R-type instructions: Write ALU result to registers



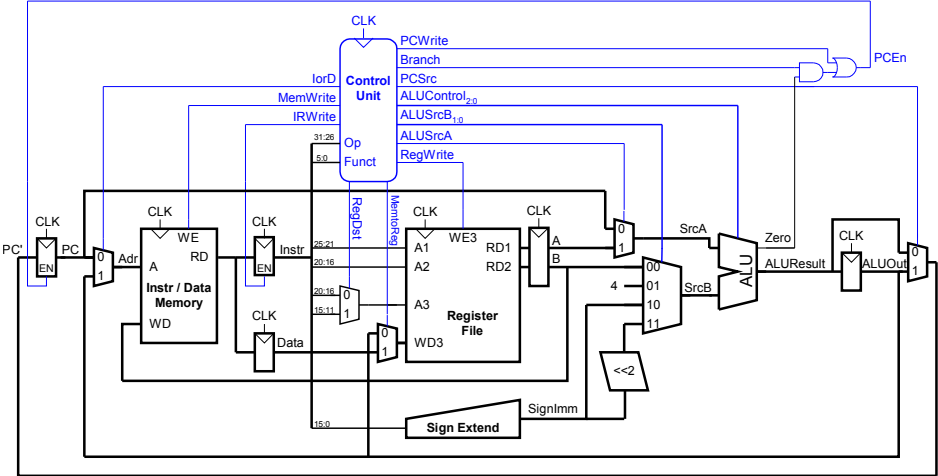
Multicycle Datapath

For bne: Add immediate to PC

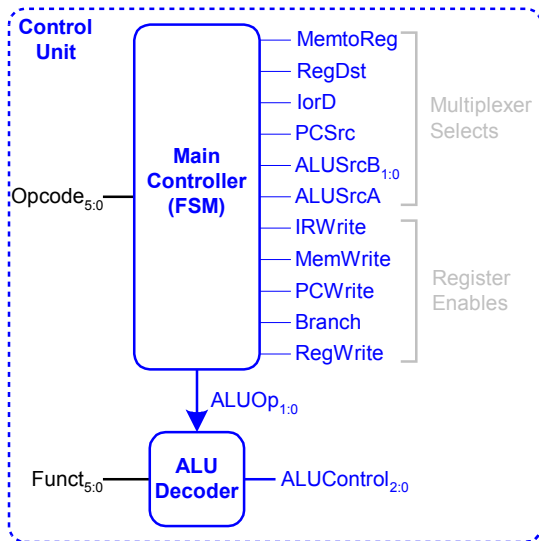


Multicycle Datapath

Add Controller



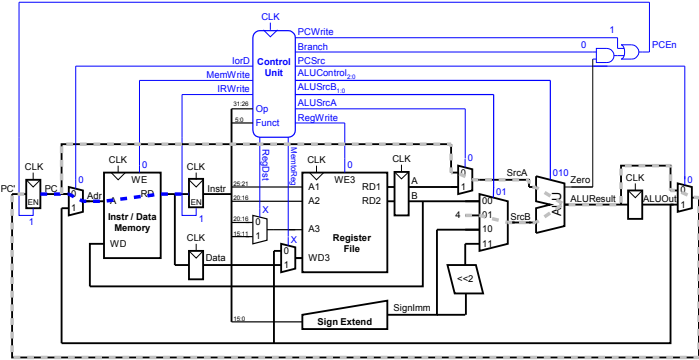
Controller Internals



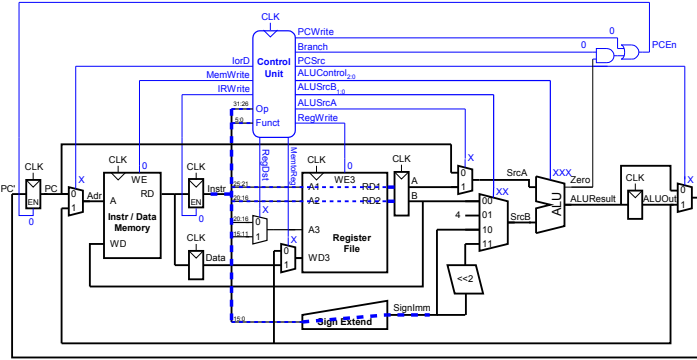
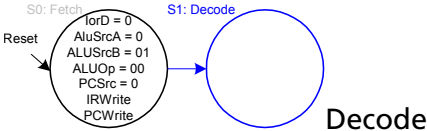
Controller Behavior



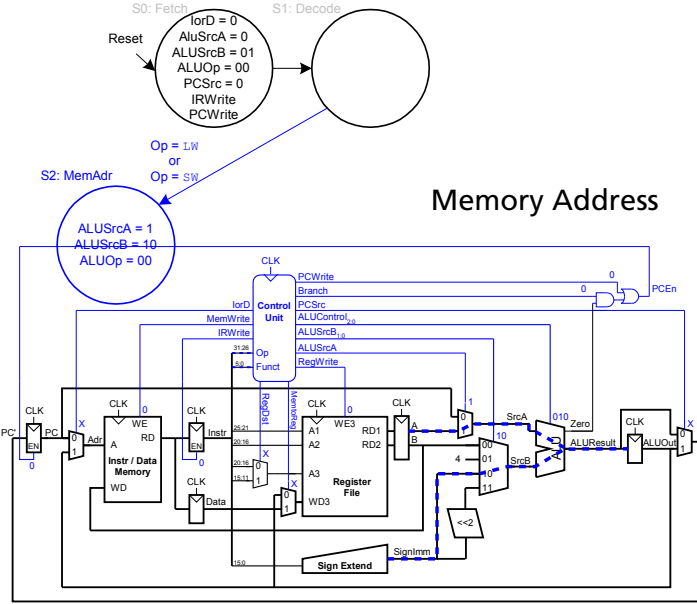
Fetch



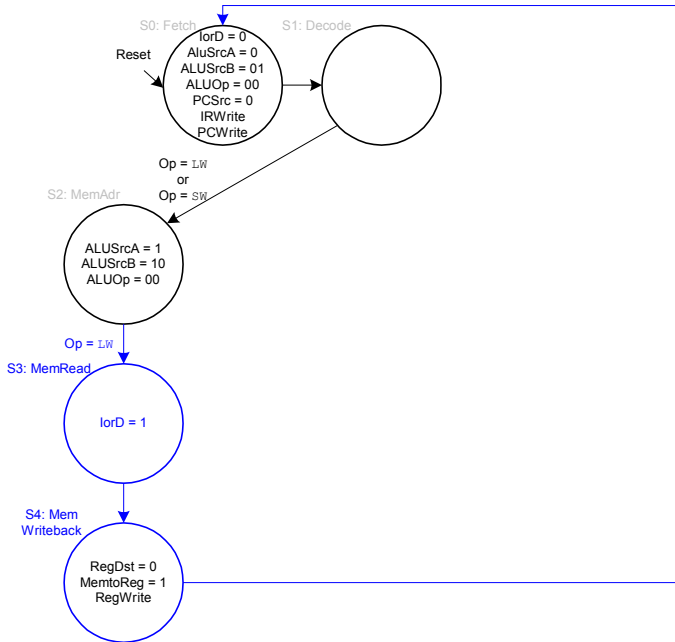
Controller Behavior



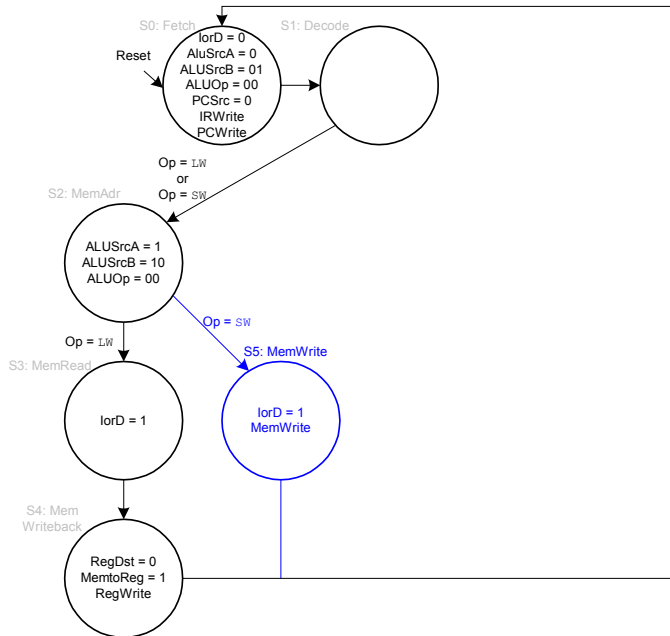
Controller Behavior



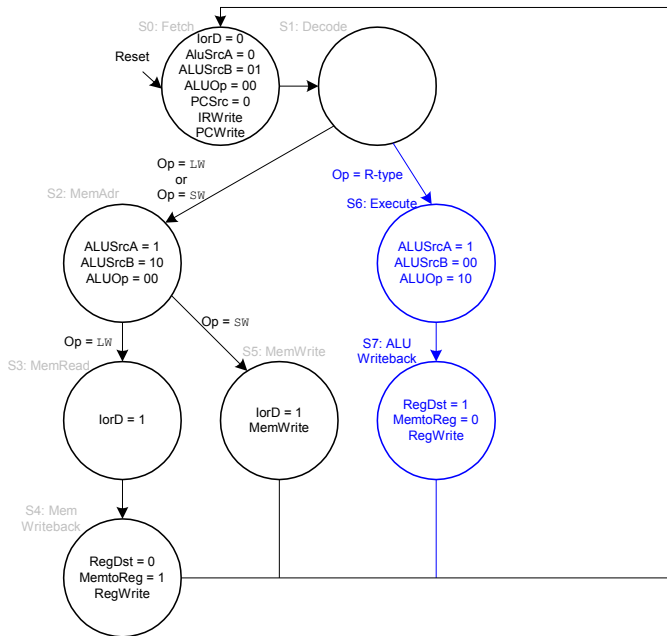
Controller Behavior



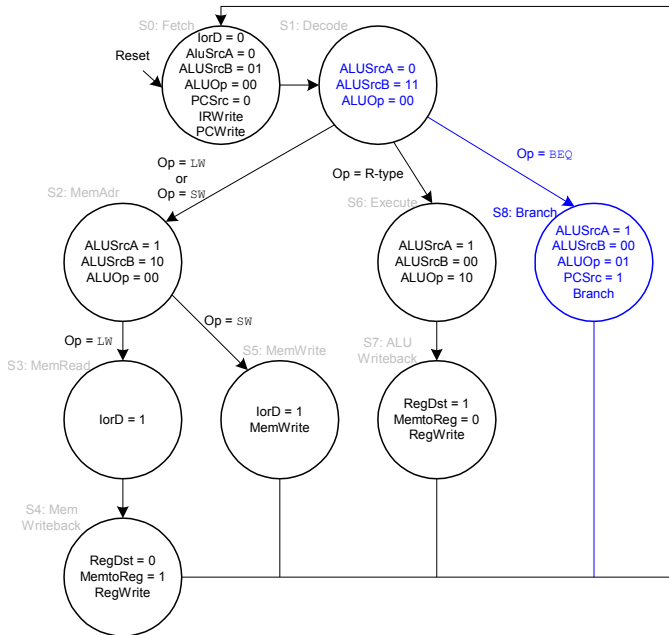
Controller Behavior



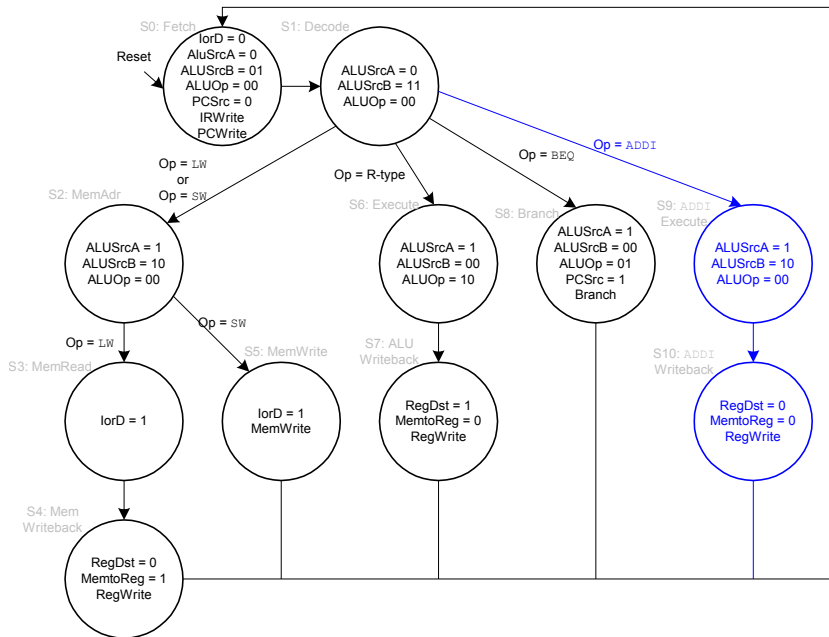
Controller Behavior



Controller Behavior

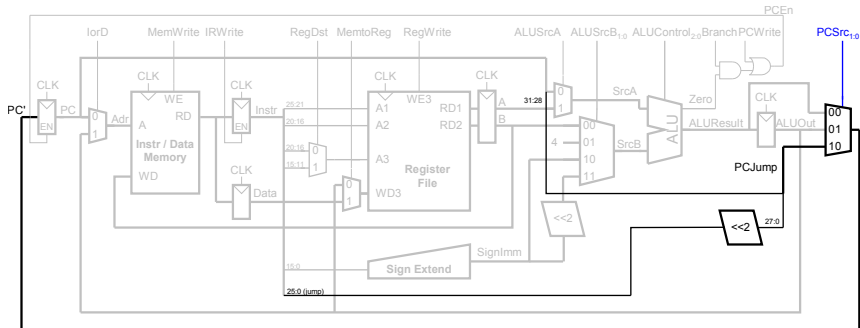


Controller Behavior

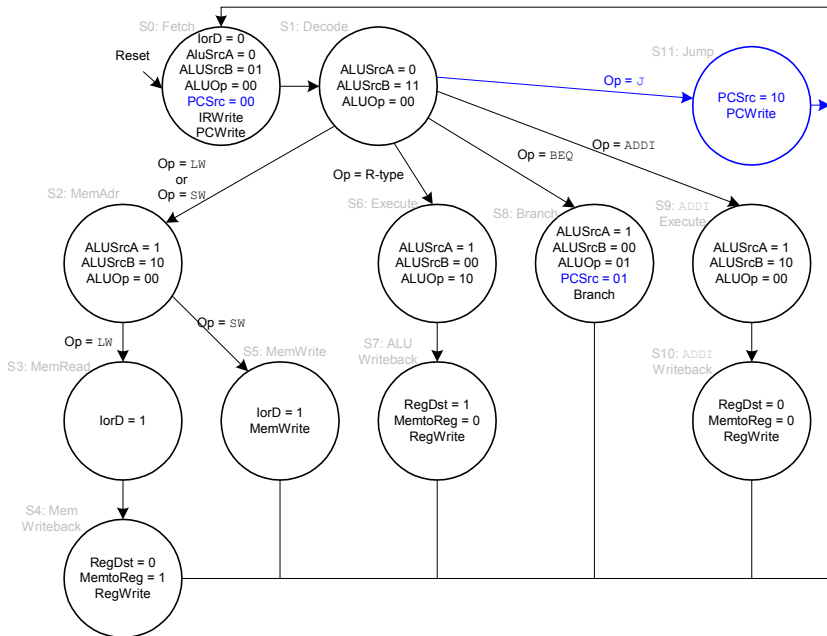


Controller Behavior

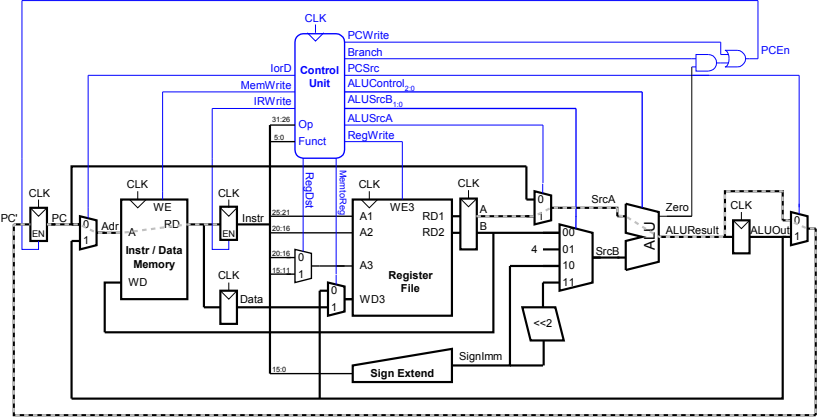
Additional circuitry for the jump instruction



Controller Behavior



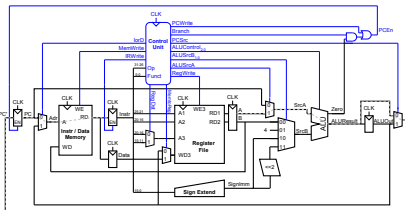
Multicycle Critical Path



Two hypotheses: Reading memory or going through the ALU

Multicycle Clock Period

Element	Delay
Register clk-to-Q	t_{pcq-PC} 30 ps
Register setup	t_{setup} 20
Multiplexer	t_{mux} 25
ALU	t_{ALU} 200
Memory Read	t_{mem} 250
Register file read	t_{RFread} 150
Register file setup	$t_{RFsetup}$ 20



$$\begin{aligned}T_C &= t_{pcq-PC} + t_{mux} + \max\{t_{ALU} + t_{mux}, t_{mem}\} + t_{RFsetup} \\&= (30 + 25 + \max\{200 + 25, 250\} + 20) \text{ ps} \\&= 325 \text{ ps} \\&= 3.08 \text{ GHz}\end{aligned}$$

vs. 925 ps for our single-cycle processor

Execution Time for Our Multi-Cycle Processor

For a 100 billion-instruction task on our multi-cycle processor, each instruction takes 4.12 cycles on average. With a 325 ps clock period,

$$\begin{aligned}\frac{\text{Seconds}}{\text{Program}} &= \frac{\text{Instructions}}{\text{Program}} \times \frac{\text{Clock Cycles}}{\text{Instruction}} \times \frac{\text{Seconds}}{\text{Clock Cycle}} \\ &= 100 \times 10^9 \times 4.12 \times 325 \text{ ps} \\ &= 133.9 \text{ seconds}\end{aligned}$$

vs. 92.5 seconds for our single-cycle processor.