

EN164: Design of Computing Systems

Lecture 14: Processor / Single-Cycle Design 3

Professor Sherief Reda

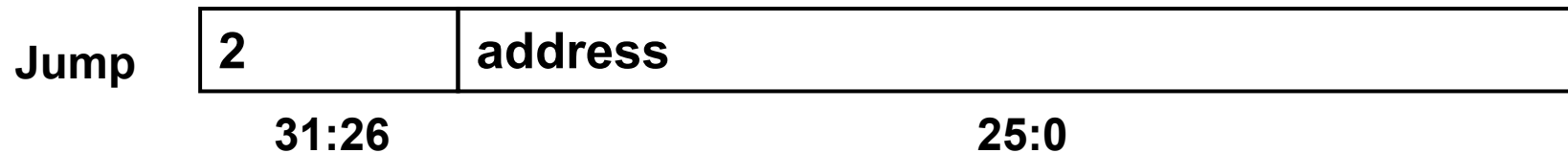
<http://scale.engin.brown.edu>

Electrical Sciences and Computer Engineering
School of Engineering
Brown University
Spring 2011



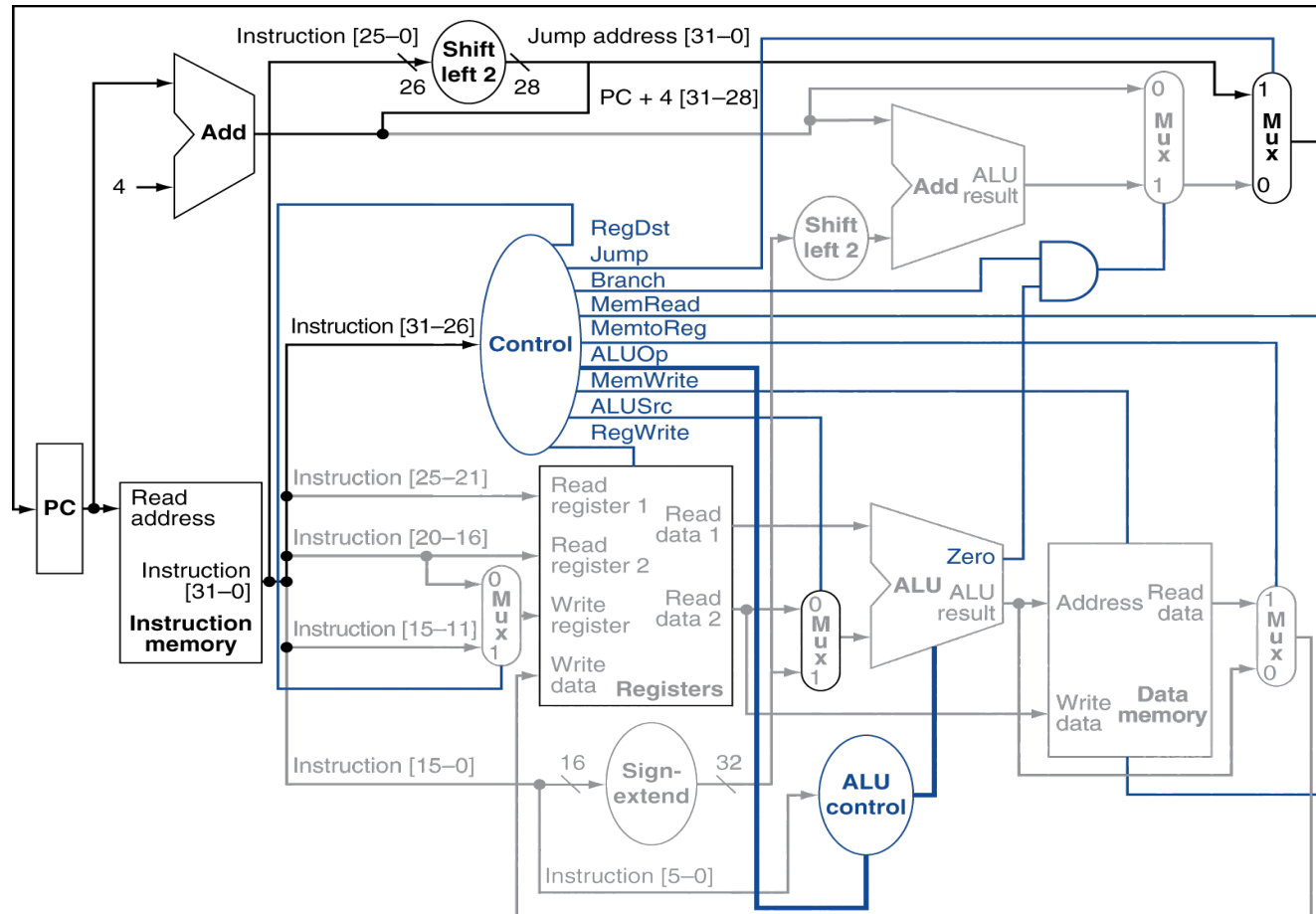
[material from Patterson & Hennessy, 4th ed and Harris 1st ed]

Implementing jumps



- Jump uses word address
- Update PC with concatenation of
 - Top 4 bits of old PC
 - 26-bit jump address
 - 00
- Need an extra control signal decoded from opcode

Datapath and control with jumps



Instruction	Op _{5:0}	RegWrite	RegDst	AluSrc	Branch	MemWrite	MemtoReg	ALUOp _{1:0}	Jump
j	000100	0	X	X	X	0	X	XX	1

Processor performance

Program Execution Time

= (# instructions)(cycles/instruction)(seconds/cycle)

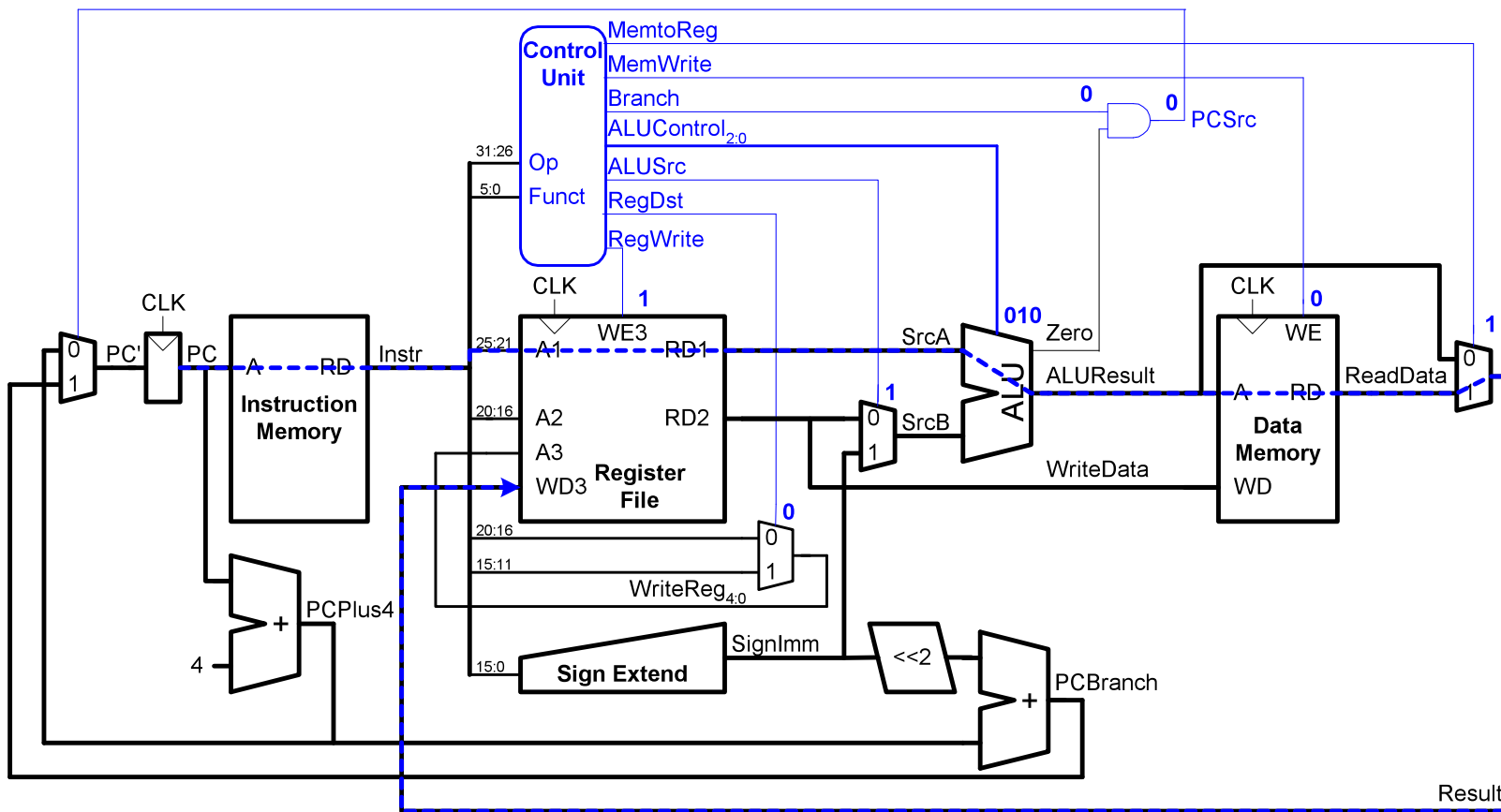
= # instructions x CPI x T_C

CPI = 1

What is T_C ?

Critical path

- T_C is limited by the critical path ($\perp w$)



- Single-cycle critical path:

$$T_c = t_{pcq_PC} + t_{mem} + \max(t_{RFread}, t_{sext} + t_{mux}) + t_{ALU} + t_{mem} + t_{mux} + t_{RFsetup}$$

Critical path delay

Element	Parameter	Delay (ps)
Register clock-to-Q	t_{pcq_PC}	30
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

$$\begin{aligned} T_c &= t_{pcq_PC} + 2t_{mem} + t_{RFread} + t_{mux} + t_{ALU} + t_{RFsetup} \\ &= [30 + 2(250) + 150 + 25 + 200 + 20] \text{ ps} \\ &= 925 \text{ ps} \end{aligned}$$

Performance issues

- Longest delay determines clock period
 - Critical path: load instruction
 - Instruction memory → register file → ALU → data memory → register file
- Not feasible to vary period for different instructions
- Violates design principle
 - Making the common case fast
- We will improve performance by pipelining