Chapter 7: Microarchitecture

Pipelined Performance

Pipelined Processor Performance Example

- SPECINT2000 benchmark:
  - 25% loads
  - 10% stores
  - 13% branches
  - 52% R-type
- Suppose:
  - 40% of loads used by next instruction
  - 50% of branches mispredicted
- What is the average CPI? (Ideally it's 1, but…)

Pipelined Processor Performance Example

Pipelined processor critical path:

$$T_{c\_pipelined} = \max \text{ of}$$

$$t_{pcq} + t_{mem} + t_{setup} \quad \text{Fetch}$$

$$2(t_{RFread} + t_{setup}) \quad \text{Decode}$$

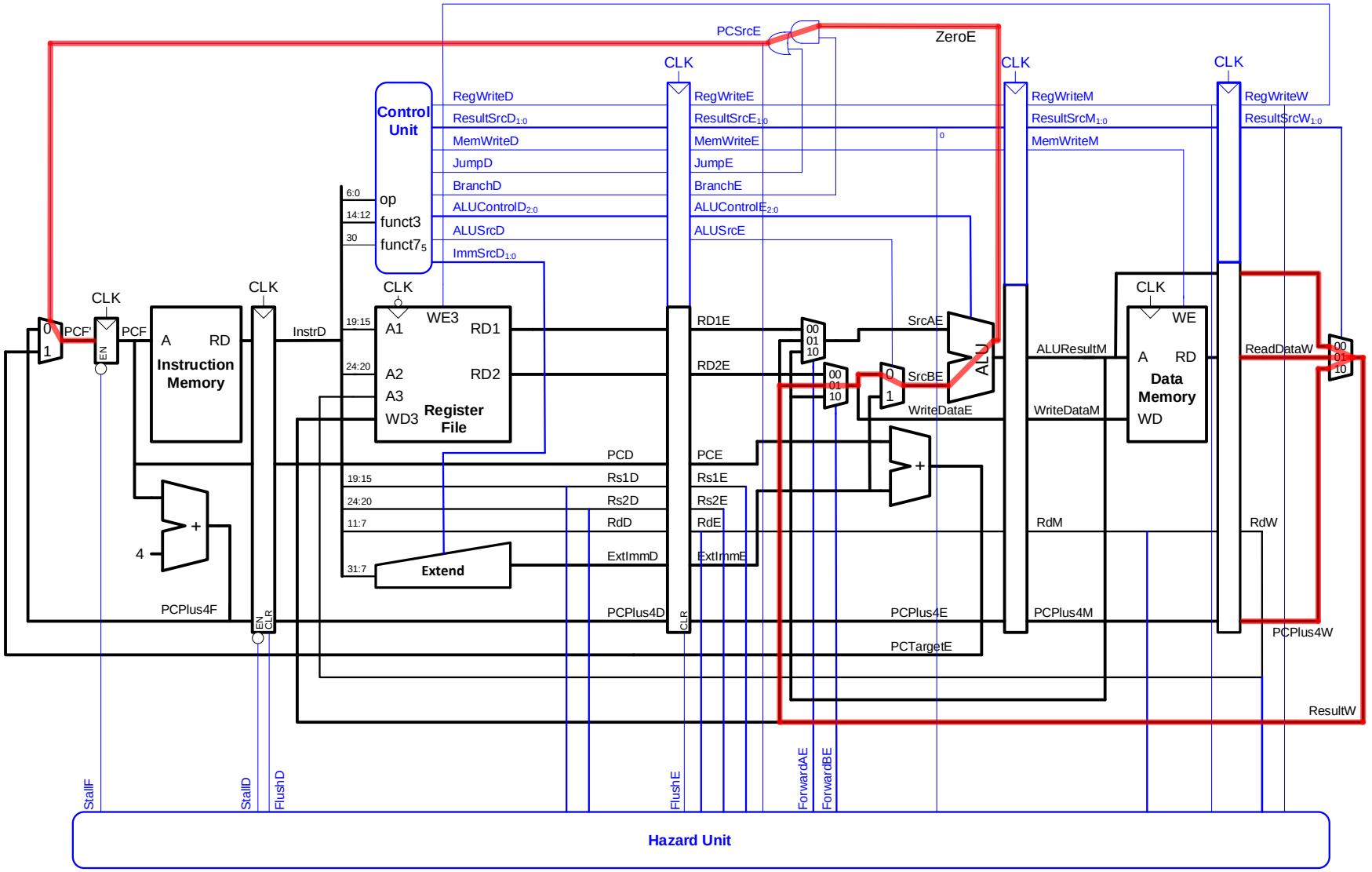
$$t_{pcq} + 4t_{mux} + t_{ALU} + t_{AND\text{-}OR} + t_{setup} \quad \text{Execute}$$

$$t_{pcq} + t_{mem} + t_{setup} \quad \text{Memory}$$

$$2(t_{pcq} + t_{mux} + t_{RFwrite}) \quad \text{Writeback}$$

- Decode and Writeback stages **both use the register file** in each cycle
- So each stage gets half of the cycle time ($T_c/2$) to do their work
- Or, stated a different way, **2x of their work** must fit in a cycle (T_c)

Pipelined Critical Path: Execute Stage



Pipelined Performance Example

Element	Parameter	Delay (ps)
Register clock-to-Q	t_{pcq_PC}	40
Register setup	t_{setup}	50
Multiplexer	t_{mux}	30
AND-OR gate	$t_{AND\text{-}OR}$	20
ALU	t_{ALU}	120
Decoder (Control Unit)	t_{dec}	25
Extend unit	t_{dec}	35
Memory read	t_{mem}	200
Register file read	t_{RFread}	100
Register file setup	$t_{RFsetup}$	60

$$T_{c\_pipelined} = t_{pcq} + 4t_{mux} + t_{ALU} + t_{AND\text{-}OR} + t_{setup}$$

$$=$$

Pipelined Performance Example

Program with 100 billion instructions

Execution Time = (# instructions) × CPI × T_c

= $(100 \times 10^9)(1.23)(350 \times 10^{-12})$

= 43 seconds

Processor Performance Comparison

Processor	Execution Time (seconds)	Speedup (single-cycle as baseline)
Single-cycle	75	1
Multicycle	155	0.5
Pipelined	43	1.7