The University of Alabama in Huntsville
ECE Department
CPE 431 01, CPE 531 01/01R
Processor Basics
Fall 2022

Due September 17, 2022
1.0 (30)

1.0 **<4.1>** When processor designers consider a possible improvement to the processor datapath, the decision usually depends on the cost/performance trade-off. In the following three problems, assume that we are starting with the datapath shown, where I Mem, Add, Mux, ALU, Regs, D Mem, and Control blocks have latencies of 320 ps, 150 ps, 20 ps, 180 ps, 90 ps, 350 ps, and 120 ps, respectively, and costs of 900, 80, 10, 195, 250, 1000, and 180, respectively.

Consider the addition of a multiplier to the ALU. This addition will add 250 ps to the latency of the ALU and will add a cost of 450 to the ALU. The result will be 10 % fewer instructions executed since we wil no longer need to emulate the MUL instruction.

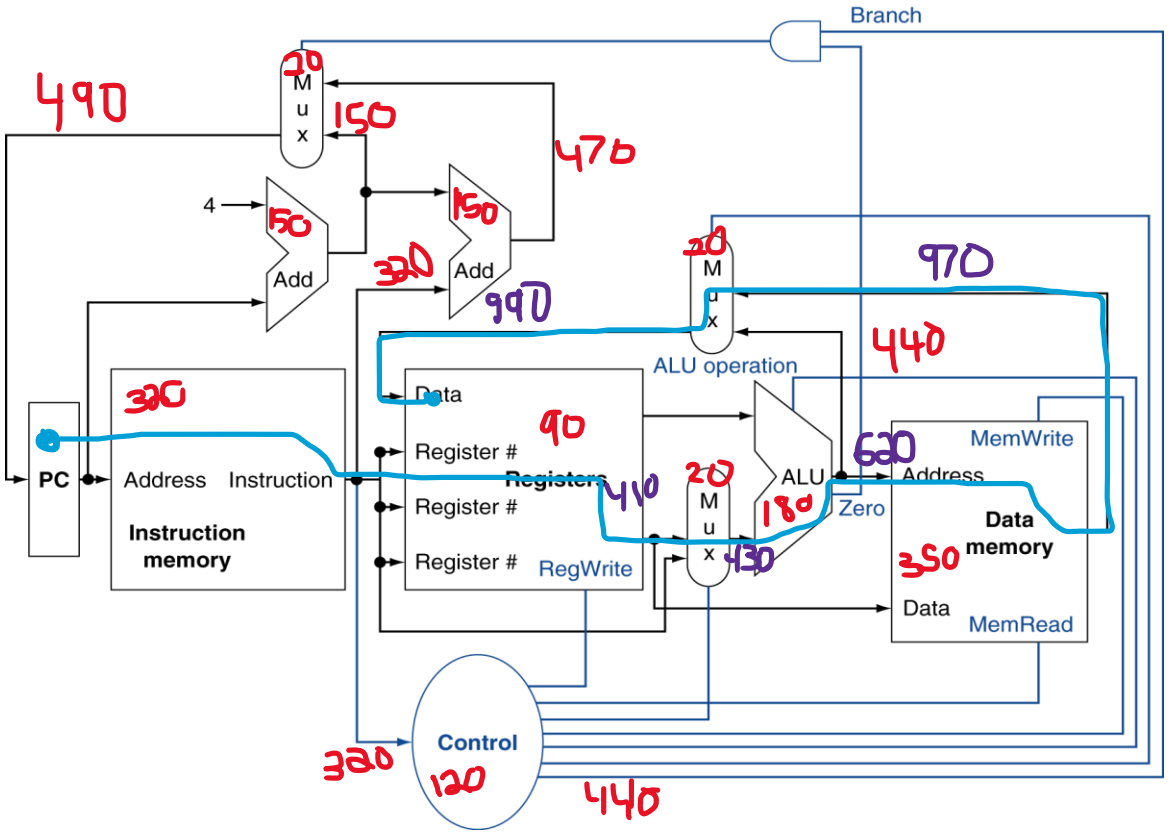
1.0.1 **<4.1> (15)** What is the clock cycle time with and without this improvement?
Path from PC lower: I-Mem, Registers, Mux, ALU, D-Mem, Mux, (end at reg)
990 ps without improvement
1240 ps with improvement

1.0.2 **<4.1> (10)** What is the speedup achieved by adding this improvement?
Given 1000 instructions. 10% is 900.
No improvement: 990000
Improvement: 1116000
Running time decreased: 126000, Slower by 0.887
There is no speedup. The improvement is slower than no improvement by 0.887

1.0.3 **<4.1> (5)** Compare the cost/performance ratio with and without this improvement.
I-Mem, Reg, 3 Mux, 2 Add, D-Mem, ALU, Control
900, 250, 30, 160, 1000, 195, 180 = 2715
With improvement = 3165
Ratio: cost/performance where performance = 1/ET
No improvement: 2715/1/990ps = 2.688 microseconds
Improvement: 3165/1/1240 ps = 3.925 microseconds
There is no improvement between the cost/performance ratios.

	I Mem	Add	Mux	ALU	Regs	D Mem	Control
Latency	320	150	20	180	90	350	120
Cost	900	80	10	195	250	1000	180

With Improvement: ALU will be 430 with 645 cost


Branch
M
u
x
4
Add
Add
M
u
x
ALU operation
PC
Address
Instruction
Instruction
memory
Data
Register #
Register #
Register #
Registers
RegWrite
ALU
Zero
M
u
x
MemWrite
Address
Data
memory
Data
MemRead
Control