PIPELINING: INTRODUCTION

Mahdi Nazm Bojnordi

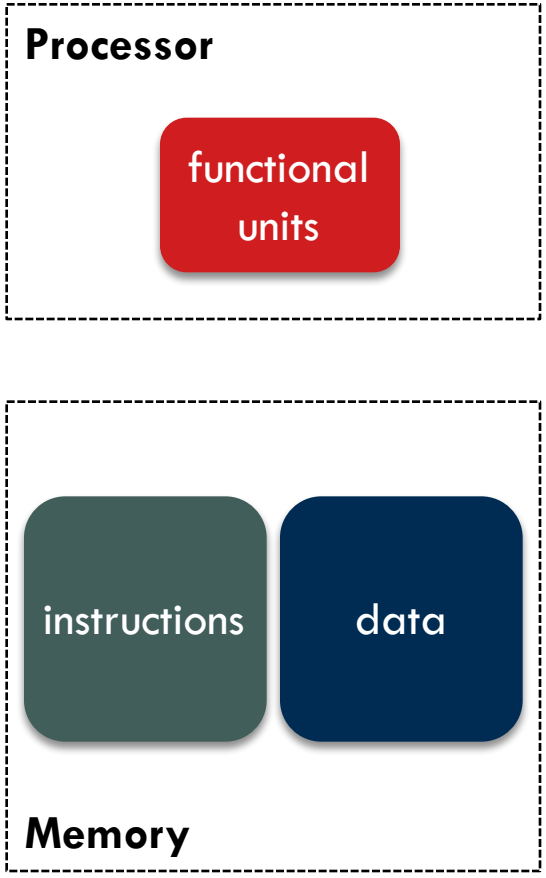
Assistant Professor

School of Computing

University of Utah

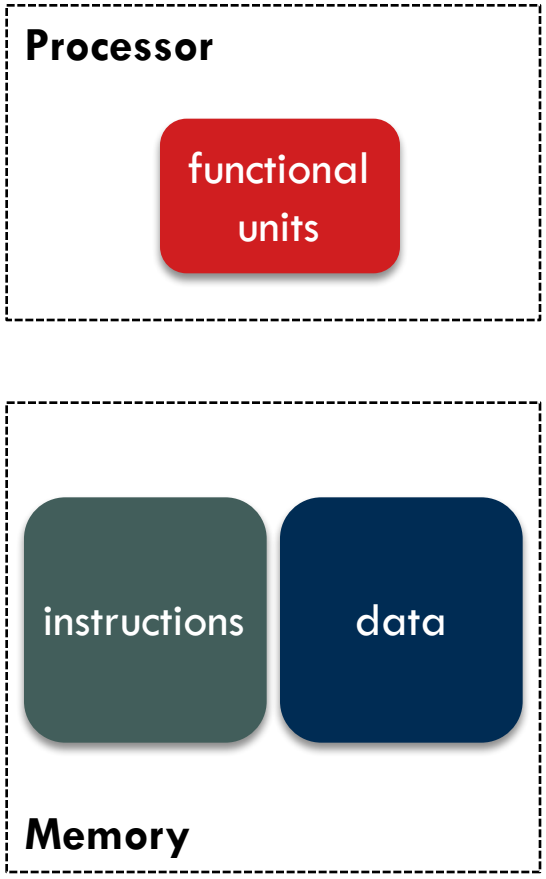
Processing Instructions

- Every RISC instruction may require multiple processing steps



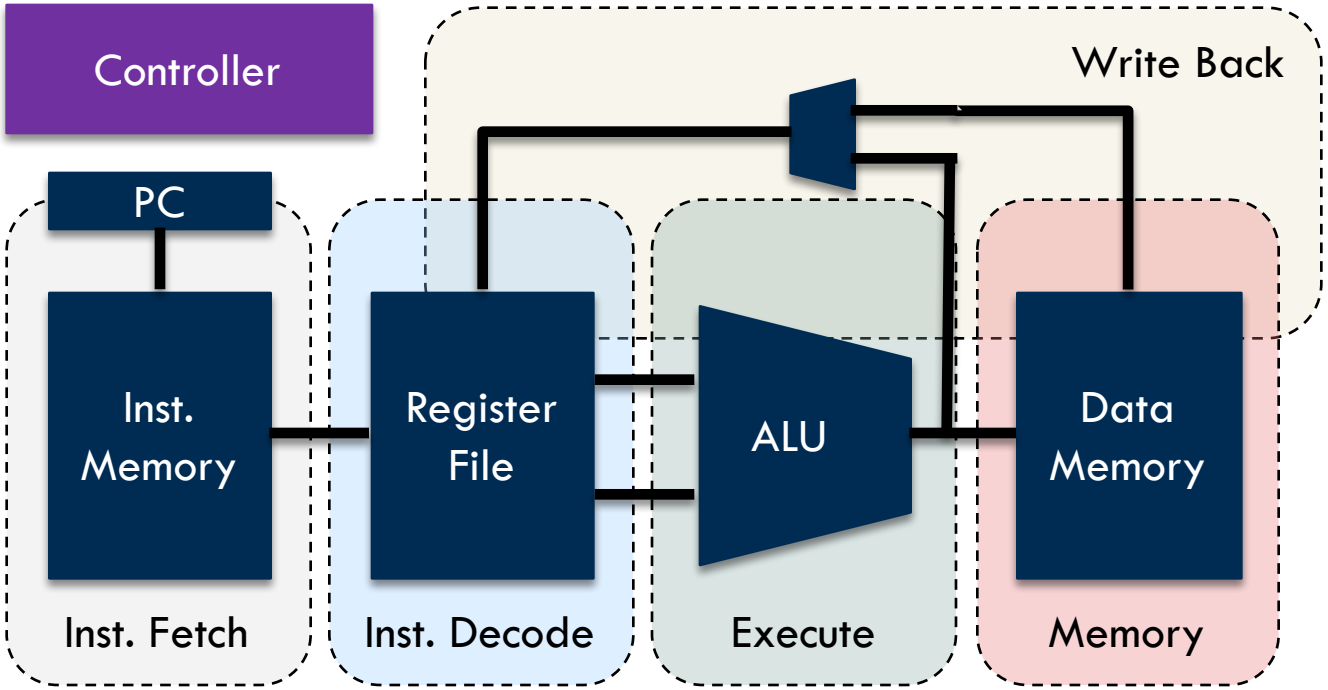
Processing Instructions

- Every RISC instruction may require multiple processing steps
  - Instruction Fetch (IF)
  - Instruction Decode (ID)
  - Register Read (RR)
    - All instructions?
  - Execute Instructions (EXE)
  - Memory Access (MEM)
    - All instructions?
  - Register Write Back (WB)



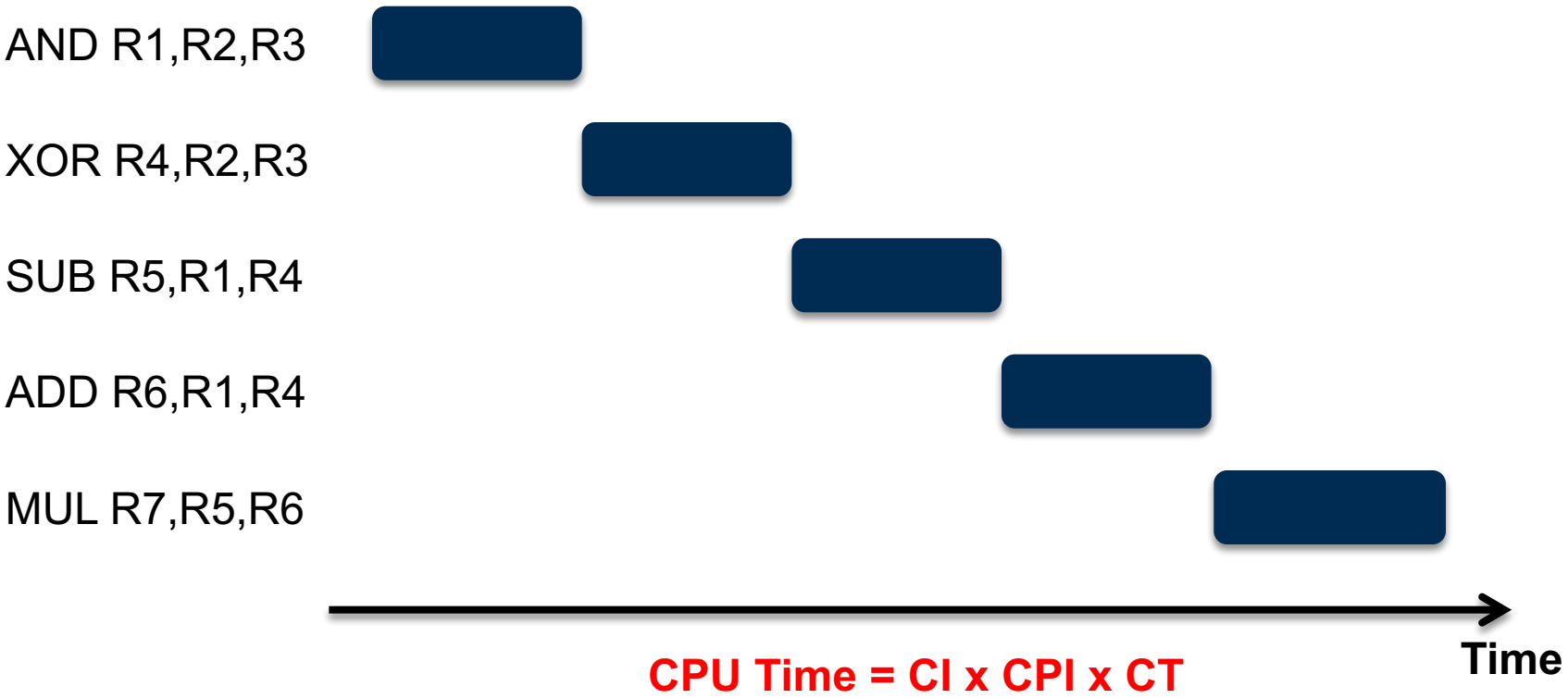
Single-cycle RISC Architecture

- Example: simple MIPS architecture
  - Critical path includes all of the processing steps



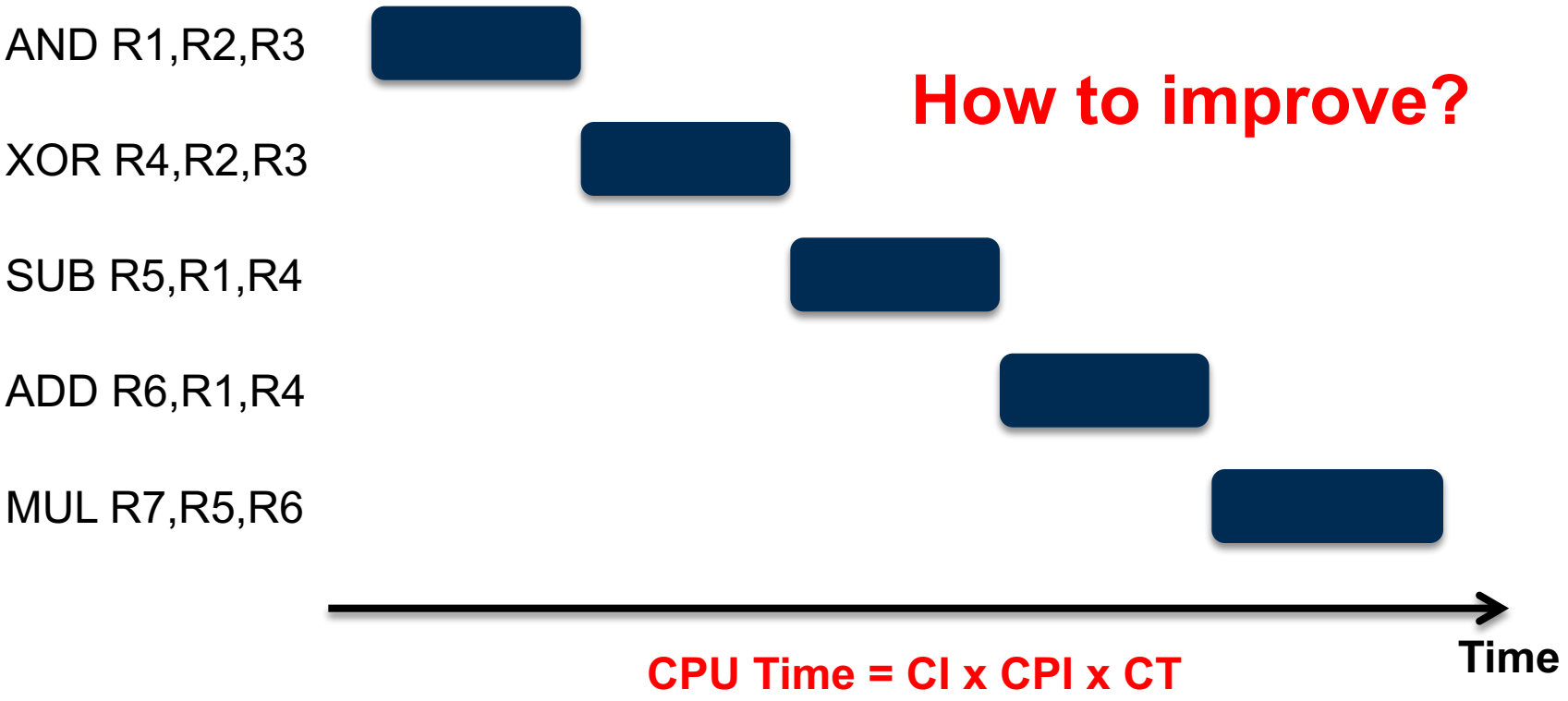
Single-cycle RISC Architecture

- Example program
  - CT=6ns; CPU Time = ?



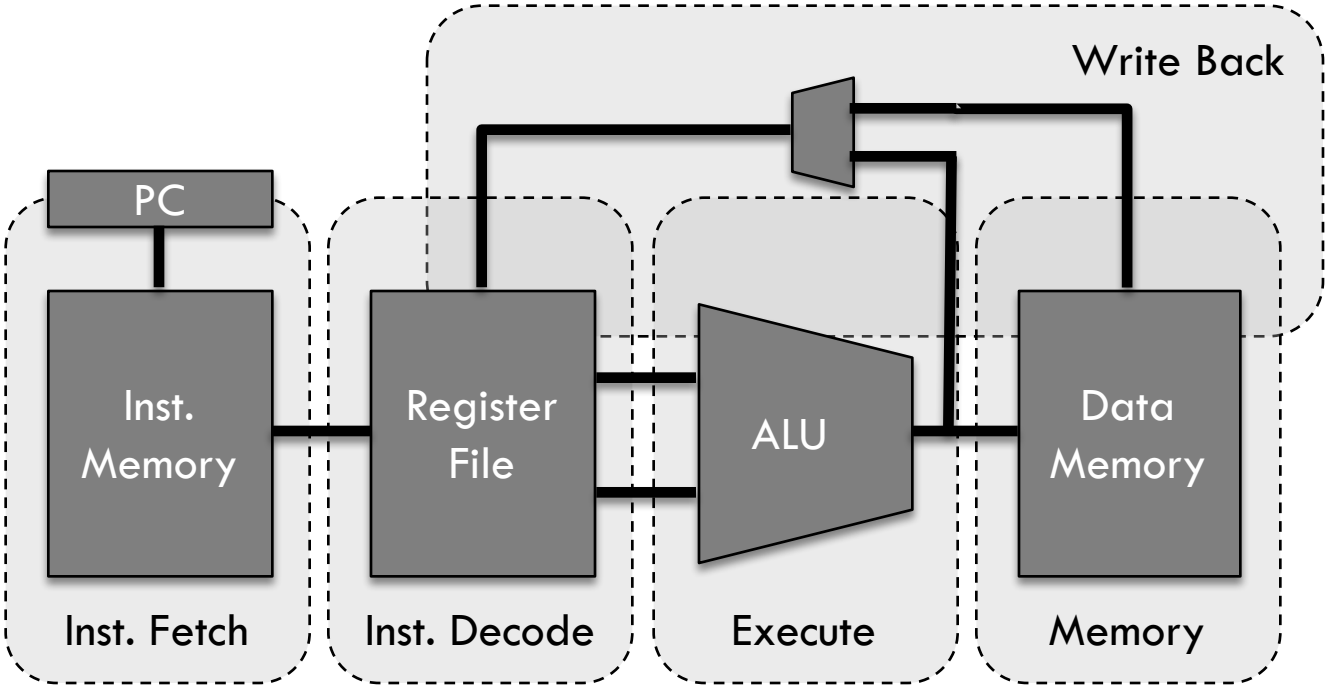
Single-cycle RISC Architecture

- Example program
  - CT=6ns; CPU Time = 5 x 1 x 6ns = 30ns



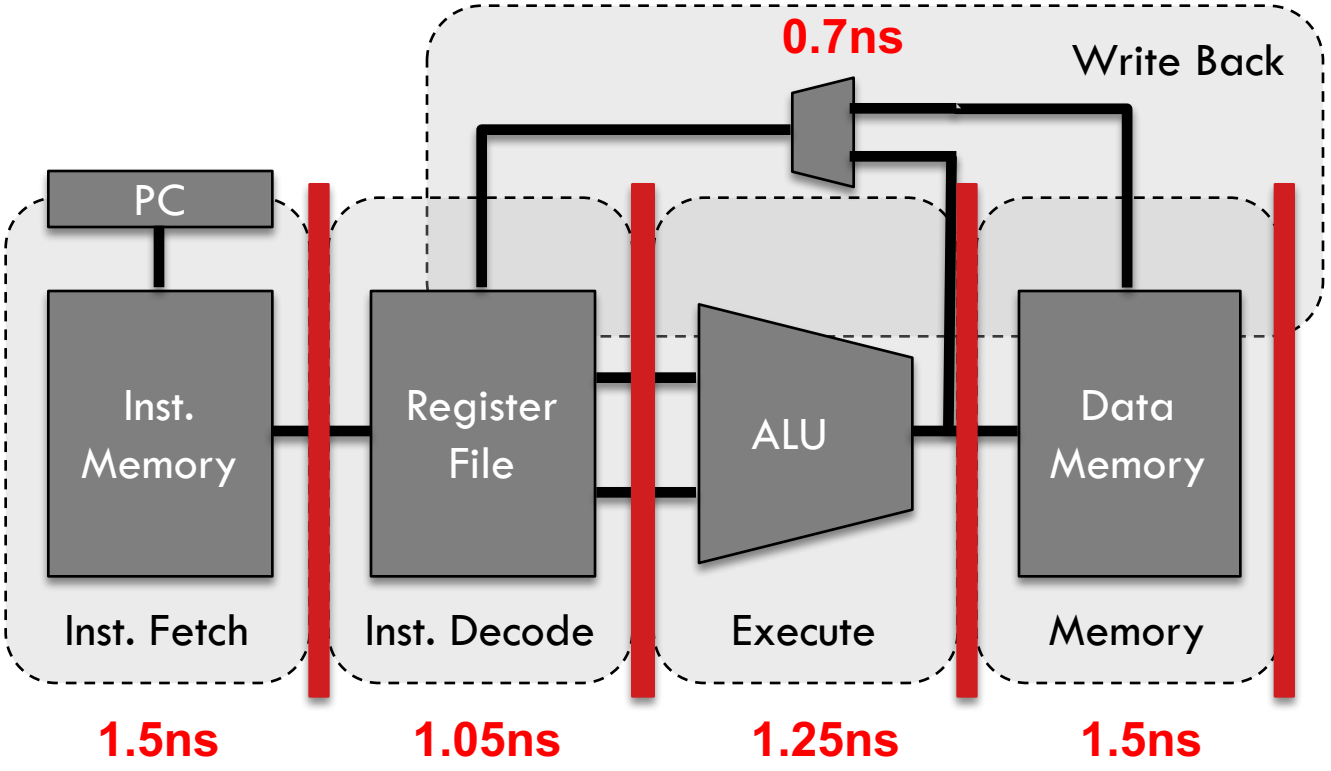
Reusing Idle Resources

- Each processing step finishes in a fraction of a cycle
  - Idle resources can be reused for processing next instructions



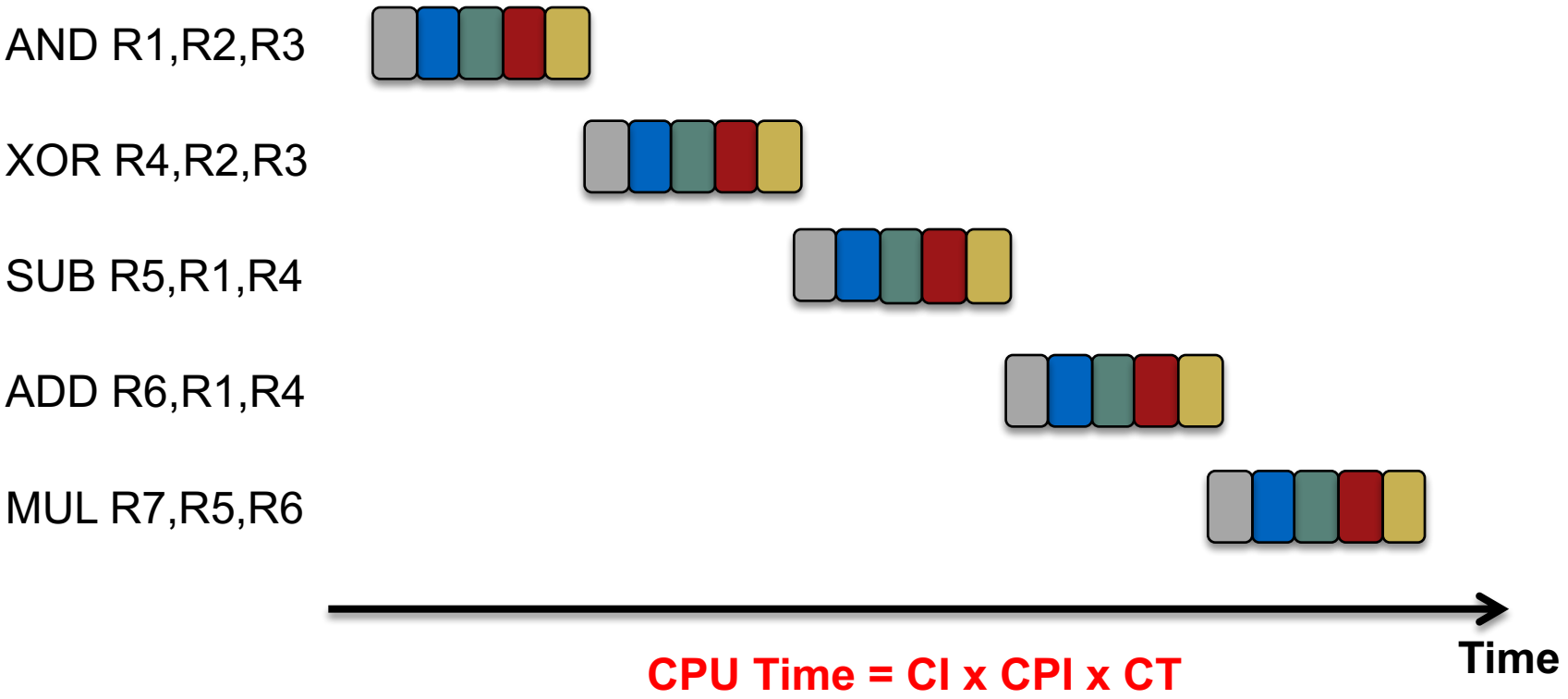
Pipelined Architecture

- Five stage pipeline
  - Critical path determines the cycle time



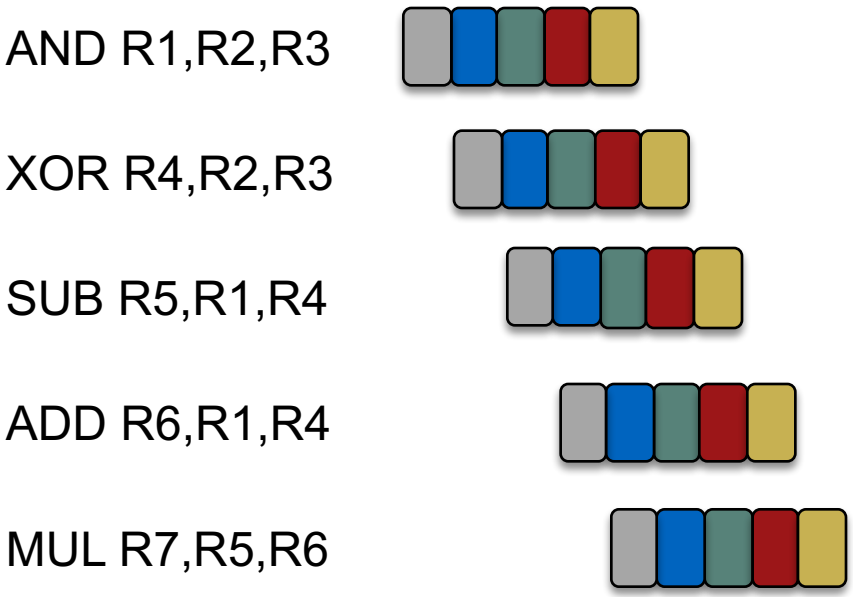
Pipelined Architecture

- Example program
  - CT=1.5ns; CPU Time = ?



Pipelined Architecture

- Example program
  - CT=1.5ns; CPU Time = *5 x 1 x 1.5 = 7.5*



What is the cost?

CT: original cycle time
P: no. pipeline stages
t: additional HW delay
New Time = CI x CPI x (t + CT/P)
Speedup = CT / (t + CT/P)

What about CPI?

CPU Time = CI x CPI x CT