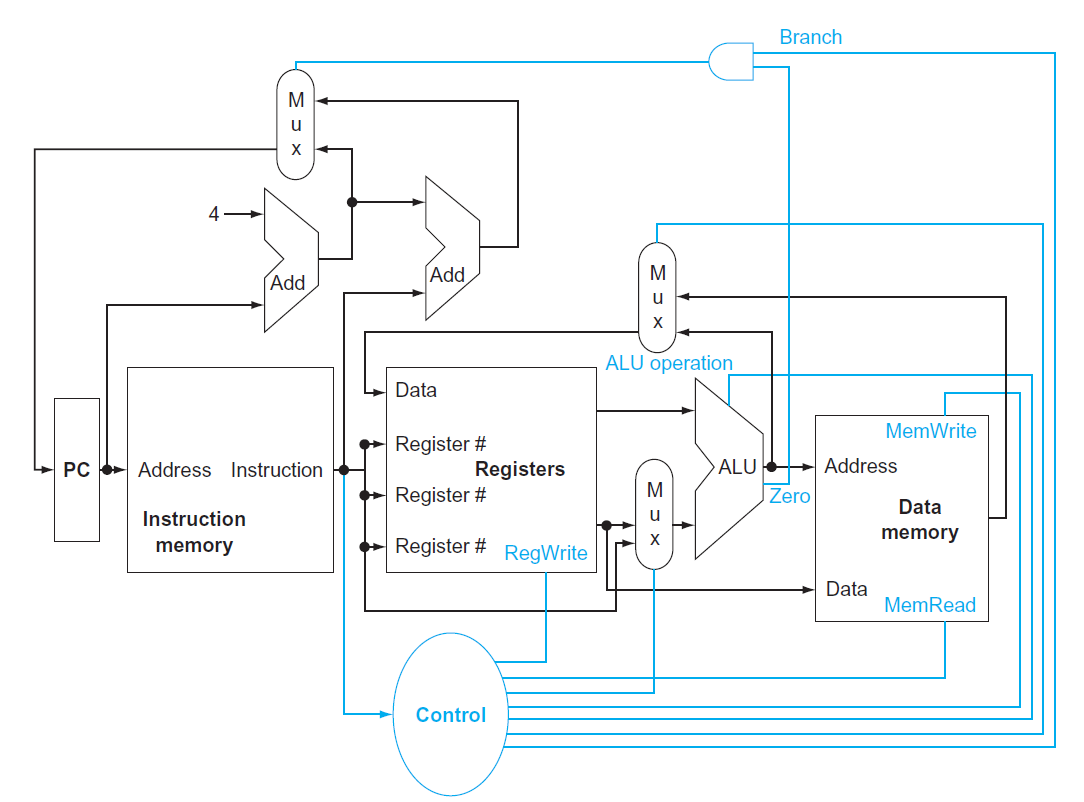
**4.1** Consider the following instruction:

Instruction: AND Rd,Rs,Rt

Interpretation: Reg[Rd] = Reg[Rs] AND Reg[Rt]



1

C

1

1

0

0

0

B

A

**4.1.1** [5] <§4.1> What are the values of control signals generated by the control in

Figure 4.2 for the above instruction?

**Mux A: 1**

**Mux B: 1**

**Mux C: 0**

**RegWrite: 1**

**ALU operation: 0000 (and)**

**Branch: 0**

**Zero: X**

**MemWrite: 0**

**MemRead: 0**

**4.1.2** [5] <§4.1> Which resources (blocks) perform a useful function for this

instruction?

**PC, Instruction memory, Registers, the first add unit (PC+4), ALU, Mux.**

**4.1.3** [10] <§4.1> Which resources (blocks) produce outputs, but their outputs

are not used for this instruction? Which resources produce no outputs for this

instruction?

**Memory unit and the second add unit produce outputs but their outputs**

**are not used for this instruction.**

**All resources produce outputs.**

**4.2** The basic single-cycle MIPS implementation in Figure 4.2 can only implement

some instructions. New instructions can be added to an existing Instruction Set

Architecture (ISA), but the decision whether or not to do that depends, among

other things, on the cost and complexity the proposed addition introduces into the

processor datapath and control. The first three problems in this exercise refer to the

new instruction:

Instruction: LWI Rt,Rd(Rs)

Interpretation: Reg[Rt] = Mem[Reg[Rd]+Reg[Rs]]

**4.2.1** [10] <§4.1> Which existing blocks (if any) can be used for this instruction?

**PC, Instruction memory, Registers, the first add unit (PC+4), ALU, Mux.**

**4.2.2** [10] <§4.1> Which new functional blocks (if any) do we need for this

instruction?

**No new blocks are needed.**

**4.2.3** [10] <§4.1> What new signals do we need (if any) from the control unit to

support this instruction?

**No new signals are needed.**

**4.4** Problems in this exercise assume that logic blocks needed to implement a

processor’s datapath have the following latencies:

**4.4.1** [10] <§4.3> If the only thing we need to do in a processor is fetch consecutive

instructions (Figure 4.6), what would the cycle time be?

**200 ps**

**4.4.2** [10] <§4.3> Consider a datapath similar to the one in Figure 4.11, but for a

processor that only has one type of instruction: unconditional PC-relative branch.

What would the cycle time be for this datapath?

**200 + 15 + 10 + 70 + 20 = 315 ps**

**4.4.3** [10] <§4.3> Repeat 4.4.2, but this time we need to support only conditional

PC-relative branches.

**200 + 90 + 20 + 90 + 20 = 420 ps**

**Maximum (420, 315) = 420 ps**

The remaining three problems in this exercise refer to the datapath element Shift -

left -2:

**4.4.4** [10] <§4.3> Which kinds of instructions require this resource?

**Branch and jump instructions**

**4.4.5** [20] <§4.3> For which kinds of instructions (if any) is this resource on the

critical path?

**Unconditional branch and jump instructions.**

**4.4.6** [10] <§4.3> Assuming that we only support beq and add instructions,

discuss how changes in the given latency of this resource affect the cycle time of the

processor. Assume that the latencies of other resources do not change.

**The add instruction doesn’t use shift left 2. beq uses shift left 2 in the path that takes 315 ps to finish, but the cycle time is 420 ps. This means that increasing latency from 0 to 105 ps or any decreasing latency for shift left 2 will not affect the cycle time of the processor. But if the latency increases more than 105 ps, it will cause longer cycle time of the processor.**