Quiz - Lecture 8

Due 21 Aug at 23:59 **Points** 5 **Questions** 4 **Available** 20 Aug at 9:10 - 21 Aug at 23:59 1 day
Time limit None **Allowed attempts** Unlimited

Instructions

These questions are intended to test your knowledge of CPU design in the Hack machine.

Take the quiz again

Attempt history

	Attempt	Time	Score
KEPT	Attempt 2	less than 1 minute	5 out of 5
LATEST	Attempt 2	less than 1 minute	5 out of 5
	Attempt 1	less than 1 minute	1 out of 5

Score for this attempt: **5** out of 5
Submitted 20 Aug at 14:14
This attempt took less than 1 minute.

Question 1

1 / 1 pts

Look at the following (incomplete) diagram of the Hack CPU. Look at the wire pointed to by the large red arrow.

Where does the signal on this wire come from and what action does this signal trigger?

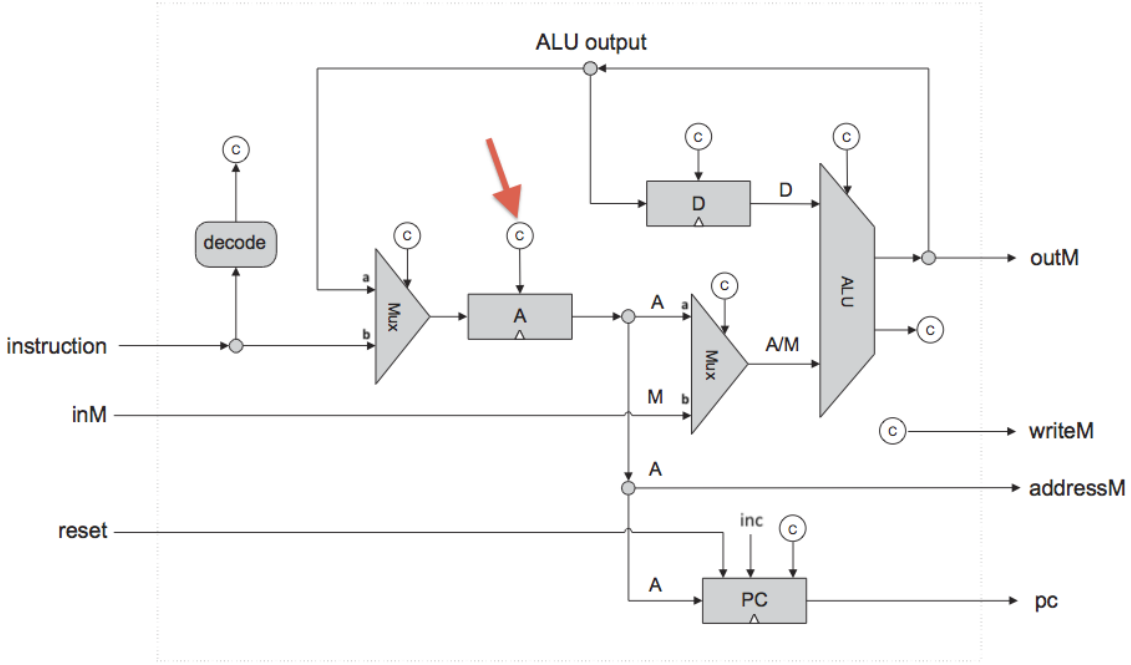

Figure 5.9 Proposed CPU implementation. The diagram shows only *data* and *address paths*, namely, wires that carry data and addresses from one place to another. The diagram does not show the CPU's *control logic*, except for inputs and outputs of control bits, labeled with a circled "c". Thus it should be viewed as an incomplete chip diagram.

Correct!

The wire comes from an "or" of the first destination bit (d1 - specifying a destination of A in a C-instruction) and the "not" of the leftmost bit of the instruction (i15 - specifying whether it is an A or a C instruction). If the signal coming out of the "or" is 1 then the A-register is loaded.

The wire comes from the instruction and it triggers something to happen to the A-register

The wire is the "not" of the left most value in the instruction register (i15 - specifying whether it is an A or a C instruction). This signal triggers the loading of the A-register.

The wire comes from the third destination (d3 - bit specifying that we are to load Memory). The A register will also get a copy of the output of the ALU under these circumstances. A 1 on the signal will trigger a loading of the A-register from a fetch from the memory location specified by the previous value in the A-register.

Question 2

1 / 1 pts

Look at the following (incomplete) diagram of the Hack CPU. Look at the wire pointed to by the large red arrow.

Where does the signal on this wire come from and what action does this signal trigger?

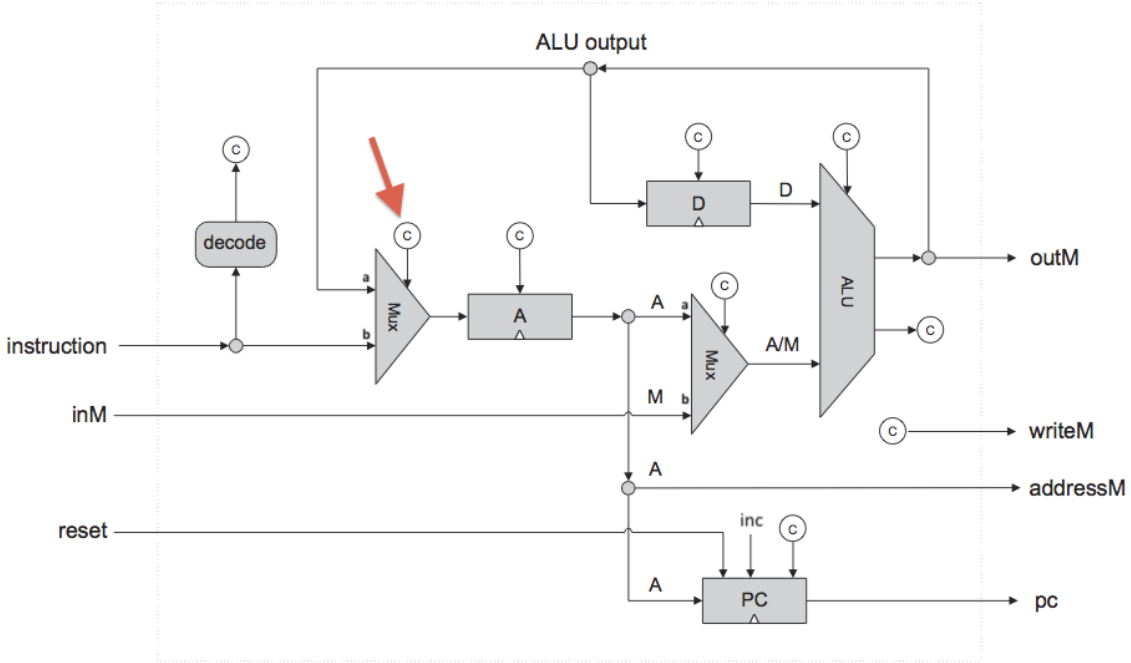

Figure 5.9 Proposed CPU implementation. The diagram shows only *data* and *address paths*, namely, wires that carry data and addresses from one place to another. The diagram does not show the CPU's *control logic*, except for inputs and outputs of control bits, labeled with a circled "c". Thus it should be viewed as an incomplete chip diagram.

This is a "not" of the leftmost destination bit of the C-instruction (d1). This allows the A-register to be loaded with the output of the ALU.

Correct!

This wire is the "not" of the left-most bit of the instruction (i15) because the ALU Output is connected to input **a** of the multiplexor. The signal allows an A-instruction to be routed to the A-register where it can be loaded.

The most significant bit of the result produced by the ALU. If the "not" of this bit is one then the Mux will select to load the output of the memory operation in to the A-register. Otherwise it will transmit the value of the program counter to the A register so the next instruction can be loaded.

The "a" bit of the C-instruction (the fourth bit from the left). This will trigger whether the value routed to A will come from the memory or the D-register.

Question 3

1 / 1 pts

Look at the following (incomplete) diagram of the Hack CPU. Look at the wire (actually, this is a bundle of wires) pointed to by the large red arrow.

Where does the signal on these wires come from and what action does this signal trigger?

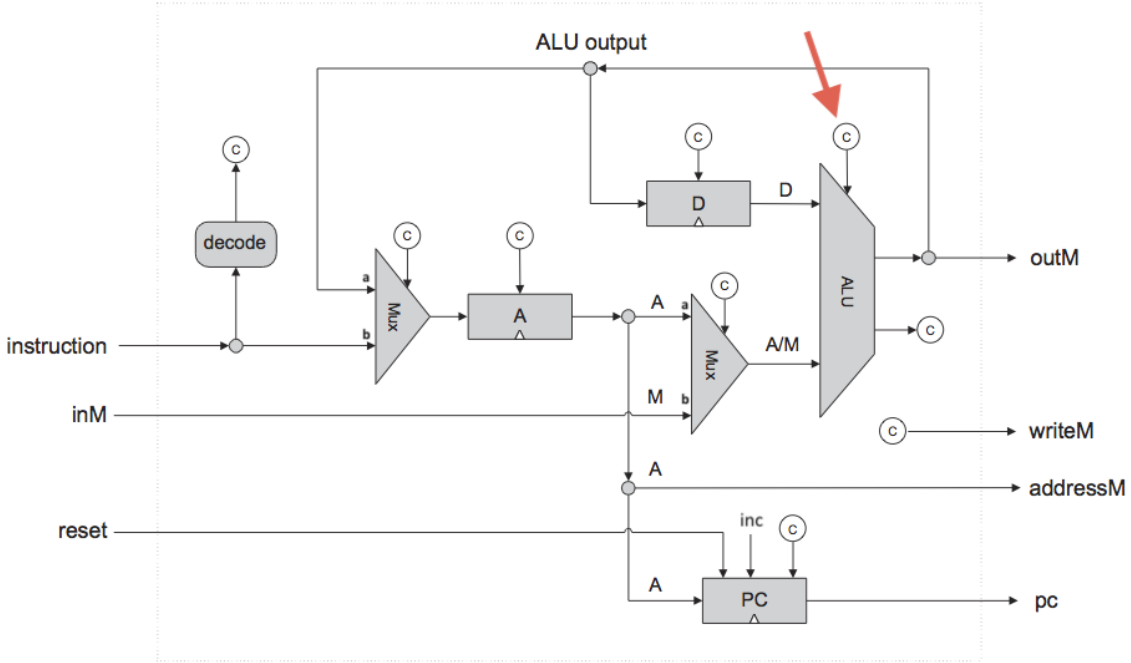

Figure 5.9 Proposed CPU implementation. The diagram shows only *data* and *address paths*, namely, wires that carry data and addresses from one place to another. The diagram does not show the CPU's *control logic*, except for inputs and outputs of control bits, labeled with a circled "c". Thus it should be viewed as an incomplete chip diagram.

The wires are the right most bits of the C-instruction (when we have an A instruction we either ignore the output of the ALU or "and" the input of these wires with one of the bits indicating that this is a C instruction) these bits determine if or how we jump. These bits feed into the logic of the ALU and make it produce the relevant bits for updating the PC to determine what instructions to execute next.

Correct!

These wires are c1 through to c6 wires of the instruction when we have a C-instruction (when we have an A instruction we either ignore the output of the ALU or "and" the input of these wires with one of the bits indicating that this is a C instruction). Their purpose is to determine the operations we perform on the ALU.

These wires are the rightmost 15 wires of an A-instruction and they are used to put values into the A-register from the ALU. If we have a C-instruction, the circuit is wired such that these wires are ignored.

These wires are c1 through to c6 wires of the instruction when we have a C-instruction (when we have an A instruction we either ignore the output of the ALU or "and" the input of these wires with one of the bits indicating that this is a C instruction). Their purpose is to determine where the result of the ALU goes.

Question 4

2 / 2 pts

Look at the following (incomplete) diagram of the Hack CPU. Look at the wire (and it is a single wire) pointed to by the large red arrow.

Where does the signal on this wire come from and what action does this signal trigger?

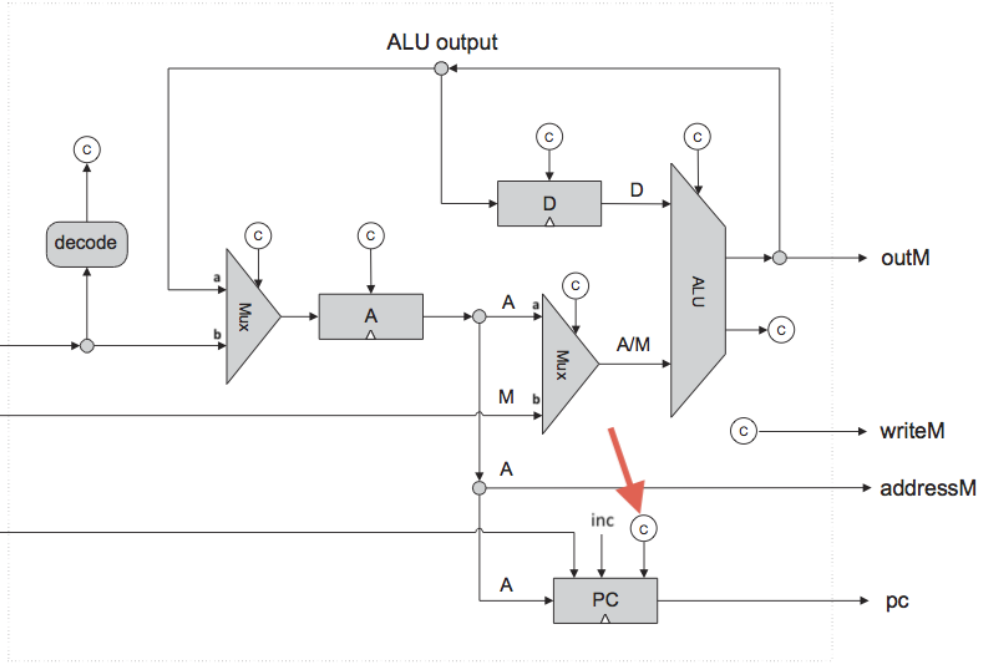

Figure 5.9 Proposed CPU implementation. The diagram shows only *data* and *address paths*, namely, wires that carry data and addresses from one place to another. The diagram does not show the CPU's *control logic*, except for inputs and outputs of control bits, labeled with a circled "c". Thus it should be viewed as an incomplete chip diagram.

This is the "z" wire that comes out of the ALU. This tells the PC register whether to increment this time or stay the same.

Correct!

The wire is load wire of the PC register. This tells the PC whether or not to load the value of the A-register into the PC which effectively performs a jump. This wire comes from some logic gates that combines some status signals from the ALU with the three right-most (jump) wires of the C-instruction.

This is the "ng" wire that comes out of the ALU. This tells the PC register whether to increment this time or stay the same.

This is "ignore reset" wire of the PC register. This wire tells the PC to ignore the signal that comes to the PC when the values of an A-instruction land in the A-register.

Quiz score: **5** out of 5