



[Courseware](#) [Course Info](#) [Discussion](#) [Wiki](#) [Progress](#) [Discussion Guidelines](#) [Resources](#) [Exploring Engineering](#)
[Syllabus](#) [How to Use Jade](#)

Help

DATA HAZARDS PART 1

	5:08 / 5:08	1.0x			
--	-------------	------	--	--	--

Download transcript .txt

Show Discussion

 New Post

DATA HAZARDS PART 2

	12:02 / 12:02	1.0x			
--	---------------	------	--	--	--

[Download transcript](#) .txt[Show Discussion](#)[New Post](#)

1. CHECK YOUR UNDERSTANDING (1/1 point)

Consider the following piece of LC-3 code:

ADD R1, R2, R3

AND R4, R1, R2

LDR R1, R4, #2

Which of the following statements describes the forwarding that would occur within the pipeline in order to properly execute this code? **[You must check ALL that apply in order to receive credit.]**

- ☒ The AND would be forwarded the result of R1 from the ADD. ✓
- ☐ The AND would be forwarded the result of R2 from the ADD.
- ☐ The LDR would be forwarded the result of R1 from the ADD.
- ☒ The LDR would be forwarded the result of R4 from the AND. ✓

EXPLANATION

Since the ADD writes R1 and the AND will read an old value in ID, R1 needs to be forwarded from the ADD to the AND.

The ADD instruction does not write R2, but reads it. Therefore, there is no need for forwarding.

The LDR writes R1. Therefore, it does not need to have R1 forwarded from the ADD. (Note that, fortunately, the write of R1 by the LDR to the register file occurs after the AND has been forwarded R1 from the ADD.)

Since the AND writes R4 and the LDR will read an old value in ID, R4 needs to be forwarded from the AND to the LDR.

[Hide Answer](#)*You have used 2 of 2 submissions*[Show Discussion](#)[New Post](#)

2. CHECK YOUR UNDERSTANDING (1/1 point)

Consider the following LC-3 code segment:

ADD R1, R2, R3

NOT R1, R1

AND R5, R1, R6

Help

Which of the following statements describes how the pipeline would handle forwarding in this case?

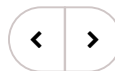
- ☐ R1 from the ADD would get forwarded to both the NOT and the AND
- ☒ R1 from the ADD would get forwarded to the NOT, and R1 from the NOT would get forwarded to the AND. ✓
- ☐ R1 from the NOT would get forwarded to both the ADD and the AND.
- ☐ No forwarding would occur in this case.

EXPLANATION

The intent in writing this program is for the result of the ADD to get inverted before being used by the AND. The second case fulfills this requirement. Note that when the AND is in EX that there are two possible forwarding cases involving R1: from the NOT in MEM and from the ADD in WB. The hardware must prioritize MEM to EX forwarding over WB to EX forwarding in order for the program to execute in the way that the programmer intended.

[Final Check](#)[Save](#)[Hide Answer](#)

You have used 1 of 2 submissions

[Show Discussion](#)[New Post](#)

EdX offers interactive online classes and MOOCs from the world's best universities. Online courses from MITx, HarvardX, BerkeleyX, UTx and many other universities. Topics include biology, business, chemistry, computer science, economics, finance, electronics, engineering, food and nutrition, history, humanities, law, literature, math, medicine, music, philosophy, physics, science, statistics and more. EdX is a non-profit online initiative created by founding partners Harvard and MIT.

About edX[About](#)[News](#)[Contact](#)[FAQ](#)[edX Blog](#)**Follow Us**[Facebook](#)[Twitter](#)[LinkedIn](#)[Google+](#)

EdX, Open edX, and the edX and Open edX logos are registered trademarks or trademarks of edX Inc.

[Terms of Service and Honor Code](#)

[Privacy Policy \(Revised 10/22/2014\)](#)



Help

[Jobs at edX](#)



[Meetup](#)

[Reddit](#)

[Youtube](#)