Courseware   **Course Info**   **Discussion**   **Wiki**   **Progress**   **Discussion Guidelines**   **Resources**   **Exploring Engineering**   **Syllabus**

How to Use Jade

Help

LC-3 FSM

0:00 / 1:36   1.0x

Download transcript | .txt

Show Discussion   🖉 New Post

1. CHECK YOUR UNDERSTANDING (1/1 point)

Which of the following operations are **not** performed within the LC-3 microarchitecture during Instruction Fetch for an ADD instruction?

- ◯ The value 1 is added to the PC.
- ◯ The incremented PC is loaded into the PC register.
- ◯ The PC is sent over the bus to the MAR register.
- ⦿ The IR register is loaded with the instruction in memory that immediately follows the ADD.   ✔

EXPLANATION

The first two statements are true since part of Instruction Fetch involves incrementing the PC and loading that value into the PC

register.

The third statement is true since the PC is loaded into MAR before the instruction is read from memory.

The last statement is false, since the IR would be loaded with the ADD instruction, not the instruction that follows the ADD. The next instruction will be fetched after the ADD has been processed, using the incremented PC.

| Final Check | **Save** | Hide Answer | *You have used 1 of 2 submissions* |

Show Discussion   ✎ New Post

2. CHECK YOUR UNDERSTANDING (1/1 point)

Which of the following statements **incorrectly** describes the operation of the LC-3 microarchitecture for an LDR instruction?

- ⦿ During Evaluate Address, two source operands, one for the base address and another for the data, are read from the register file. ✔
- ◯ During Evaluate Address, the base address is added to the sign extended offset and loaded into MAR.
- ◯ During Operand Fetch, data is loaded into MDR.
- ◯ During Store Result, data is passed from MDR to the register file.

EXPLANATION

During Evaluate Address, only a single source operand—for the base address—is read from the register file.

During Evaluate Address, the address is formed by adding the value in the base register to the offset, and the result is loaded into MAR.

During Operand Fetch, data is read from memory and loaded into MDR.

During Store Result, data is passed from the MDR to the register file to be stored.

| Final Check | **Save** | Hide Answer | *You have used 1 of 2 submissions* |

Show Discussion   ✎ New Post

3. CHECK YOUR UNDERSTANDING (1/1 point)

The LC-3 microarchitecture has a number of loadable registers whose load signals are asserted when a new value is to be placed into the register. Which of the following statements **incorrectly** describes the operation of the different load control signals?

○ LD.PC and LD.IR are both asserted during Instruction Fetch.

○ LD.MAR is asserted during Instruction Fetch, as well as during Evaluate Address for Load and Store instructions.

○ LD.REG is asserted during Store Result for both the ADD and LDR instructions.

● LD.MDR is asserted during Evaluate Address for an LDR instruction. ✔

EXPLANATION

During Instruction Fetch, the incremented PC is loaded into the PC register, and the instruction fetched from Memory is loaded into the IR register.

The MAR register is loaded during Instruction Fetch and also during Evaluate Address.

For both the ADD and LDR instructions, the register file is updated during Store Result by asserting LD.REG.

For an LDR instruction, the MDR register is loaded during Fetch Operands, not Evaluate Address.

Final Check   **Save**   Hide Answer   *You have used 1 of 2 submissions*

Show Discussion   New Post

---

edX

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.

© 2015 edX Inc.

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)

About edX

About

News

Contact

FAQ

edX Blog

Donate to edX

Jobs at edX

Follow Us

Facebook

Twitter

LinkedIn

Google+

Tumblr

Meetup

Reddit

Youtube

POWERED BY OPENedX