



< Previous



Next >

## Beta instruction sequences

Bookmark this page

Lab due Jan 11, 2016 21:59 -02    Past due

Beta instruction sequences

0.0/5.0 points (graded)

- [Summary of Instruction Formats \(PDF\)](#)
- [Beta Documentation \(PDF\)](#)

For each of the Beta instruction sequences shown below, indicate the values of the specified registers after the sequence has been executed by an unpipelined Beta. Consider each sequence separately and assume that execution begins at location 0 and halts when the HALT() instruction is about to be executed. Also assume that all registers have been initialized to 0 before execution begins.

Remember that even though the Beta reads and writes 32-bit words from memory, all addresses are *byte addresses*, i.e., the addresses of successive words in memory differ by 4.

You can find detailed descriptions of each Beta instruction in the "Beta Documentation" handout -- see link above.

**Hint:** You can enter answers in hex by specifying a "0x" prefix, *e.g.*, 16 could be entered as "0x10". Usually one would enter addresses, values in memory, etc. using hex. You can also use a "0b" prefix to enter a binary value.

(A)

```
. = 0
AND(r31, r31, r0)
CMPEQ(r31, r31, r1)
ADD(r1, r1, r2)
OR(r2, r1, r3)
SHL(r2, r3, r4)
HALT()
```

Value left in R0?	<input type="text" value="0x000"/>	Answer: 0
Value left in R1?	<input type="text" value="0x001"/>	Answer: 1
Value left in R2?	<input type="text" value="0x002"/>	Answer: 2
Value left in R3?	<input type="text" value="0x003"/>	Answer: 3
Value left in R4?	<input type="text" value="0x010"/>	Answer: 0b10000
Address of 32-bit memory location containing OR instruction?		<input type="text" value="0x00C"/> Answer: 0xC

(B)

```
. = 0
ADDC(r31, N, r0)
LD(r0, 8, r1)
SRAC(r1, 4, r2)
ST(r2, 4, r0)
HALT()

. = 0x2000
N: LONG(0x12345678)
   LONG(0xDEADBEEF)
   LONG(0xEDEDEDED)
   LONG(0x00000004)
```

Value left in R0?	<input type="text"/>	Answer: 0x2000
Value left in R1?	<input type="text"/>	Answer: 0xEDEDEDED
Value left in R2?	<input type="text"/>	Answer: 0xEDEDEDED

 Calculator

value left in R2?  Answer: 0xFEFEFEFE

Address of 32-bit memory location written by ST?  Answer: 0x2004

Value found in 32-bit memory location with address 0?  Answer: 0xC01F2000

(C)

```
. = 0
LD(r31, X, r0)
CMPL(r0, r31, r1)
BNE(r1, L1, r31)
ADDC(r31, 17, r2)
BEQ(r31, L2, r31)
L1: SUB(r31, r0, r2)
L2: XORC(r2, 0xFFFF, r2)    // be careful here!
    HALT()

. = 0x1CE8
X: LONG(0x87654321)
```

Value left in R0?  Answer: 0x87654321

Value left in R1?  Answer: 1

Value left in R2?  Answer: 0x87654320

Value assembler assigns to L1?  Answer: 0x14

Value found in 32-bit memory location with address 8?  Answer: 0x77E10002

(D)

```
. = 0
ADDC(r31, 0, r0)
LD(r31, N, r1)
BEQ(r31, L3, r31)
L1: ANDC(r1, 1, r2)
    BEQ(r2, L2, r31)
    ADDC(r0, 1, r0)
L2: SHRC(r1, 1, r1)
L3: BNE(r1, L1, r31)
    HALT()

. = 0x2468
N: LONG(0x8F2E3D4C)
```

Value left in R0?  Answer: 17

Value left in R1?  Answer: 0

Number of times instruction labeled L2 is executed?  Answer: 32

Suppose that the instructions above were relocated so that the first instruction were at location 0x100 instead of location 0. Assuming we then started execution at location 0x100 and we wanted the instructions to perform the same computation, which instruction encodings should be changed when relocating the program?

- ☒ no instructions need to be changed
- ☐ BNE and BEQ instructions would need to be changed
- ☐ LD instructions would need to be changed

- ☐ BNE, BEQ and LD instructions would need to be changed
- ☐ all instructions need to be changed

(E)

```
. = 0
BEQ(r31, L1, r0)
ADDC(r0, 0, r0)
L1: LD(r0, 0, r1)
HALT()
```

Value left in R0?  Answer: 4

Value left in R1?  Answer: 0xC0000000

Submit

Answers are displayed within the problem

Discussion

Hide Discussion

Topic: Assignment 1 (due Jan 11) / Beta instruction sequences

Add a Post

Show all posts by recent activity

☒ BSim bug for part (E)?

The value BSim leaves in **r0** after **BEQ (r31, L1, r0)** in part (E) is different than the one that the grader considers correct. The B...

3

☒ Am i doing something wrong if i get a literal as a fraction?

When i divide the offsets by 4 i get a fraction, is there some problem with my calculation or is there some way to deal with these fr...

3

R2 in section C

Is there a problem with the grader? No matter what I fill there for R2 (with and without "0x", with and without "0b"), I always got the...

3

Clarification required on one question in Section B

What is meant by the question "Address of 32-bit memory location written by ST?" in section B? Does it asked about the address w...

3

About question 4 of Lab1-c

For the question "Value assembler assigns to L1?", it should be compute the 32bit value of "SUB(r31, r0, r2)", am I right? If so, what...

5

Where is N declared?

How does the assembler assign a value to N in the line ADDC(r31, N, r0) I don't see it declared anywhere. Have I missed a crucial n

3

< Previous

Next >



[edX for Business](#)  
[Open edX](#)  
[Careers](#)  
[News](#)

---

# Legal

[Terms of Service & Honor Code](#)  
[Privacy Policy](#)  
[Accessibility Policy](#)  
[Trademark Policy](#)  
[Sitemap](#)  
[Cookie Policy](#)  
[Your Privacy Choices](#)

---

# Connect

[Idea Hub](#)  
[Contact Us](#)  
[Help Center](#)  
[Security](#)  
[Media Kit](#)



© 2024 edX LLC. All rights reserved.  
深圳市恒宇博科技有限公司 [粤ICP备17044299号-2](#)