

BÁO CÁO THỰC HÀNH KIẾN TRÚC MÁY TÍNH (IT3280) TUẦN 2

Assignment 1:

C:\Users\Asus\Desktop\ktmt\mips1.asm - MARS 4.5

File Edit Run Settings Tools Help

Run speed at max (no interaction)

Edit **Execute**

Text Segment

Offset	Address	Code	Basic	Source
0x00400000	0x20103007	addi \$t6,\$0,0x000003...	3: addi \$a0,\$zero,0x3007	\$a0 = 0 + 0x3007 = 0x3007 ;I-...
0x00400004	0x00000020	add \$t6,\$0,\$0	4: add \$a0,\$zero,\$0	\$a0 = 0 + 0 = 0 ;R-...

Data Segment

Address	Value (+0)	Value (+4)	Value (+8)	Value (+c)	Value (+10)	Value (+14)	Value (+18)	Value (+1c)
0x10010000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x10010020	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x10010040	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x10010060	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x10010080	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x100100a0	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x100100c0	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x100100e0	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x10010100	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x10010120	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000

0x10010000 (.data) ☒ Hexadecimal Addresses ☒ Hexadecimal Values ☐ ASCII

Registers Coproc 1 Coproc 0

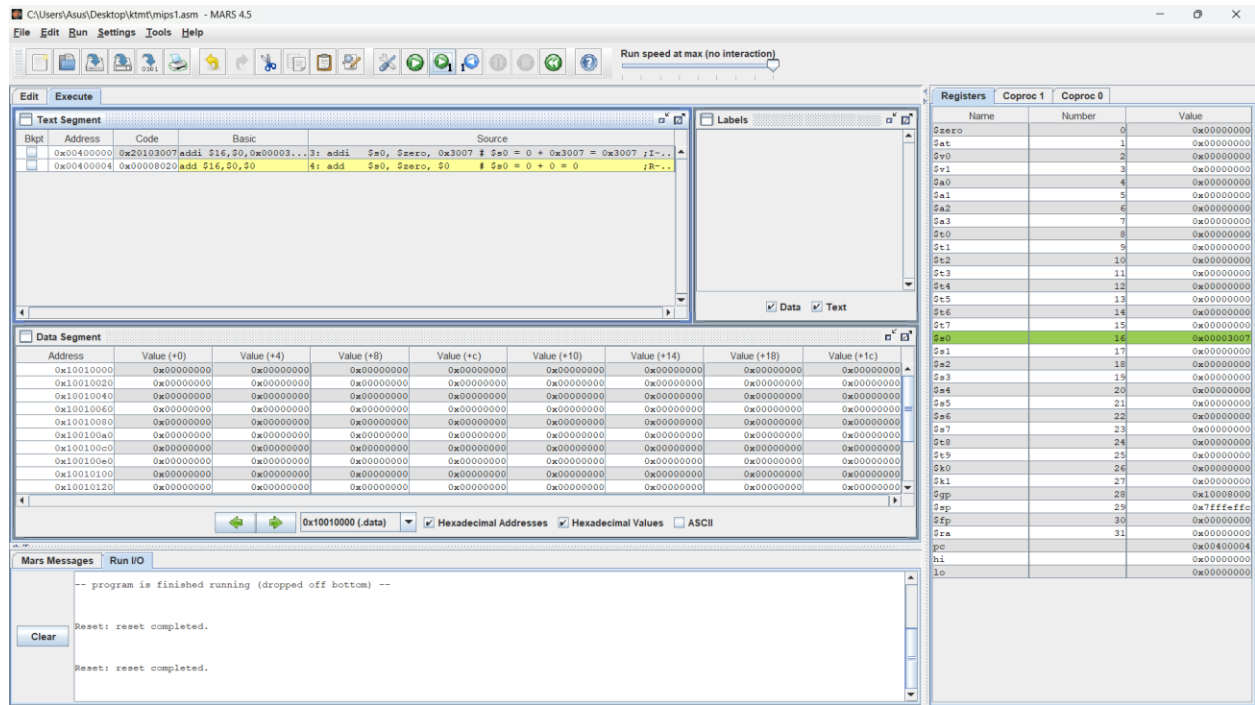
Name	Number	Value
\$zero	0	0x00000000
\$at	1	0x00000000
\$v0	2	0x00000000
\$v1	3	0x00000000
\$a0	4	0x00000000
\$a1	5	0x00000000
\$a2	6	0x00000000
\$a3	7	0x00000000
\$t0	8	0x00000000
\$t1	9	0x00000000
\$t2	10	0x00000000
\$t3	11	0x00000000
\$t4	12	0x00000000
\$t5	13	0x00000000
\$t6	14	0x00000000
\$t7	15	0x00000000
\$a0	16	0x00000000
\$a1	17	0x00000000
\$a2	18	0x00000000
\$a3	19	0x00000000
\$a4	20	0x00000000
\$a5	21	0x00000000
\$a6	22	0x00000000
\$a7	23	0x00000000
\$t8	24	0x00000000
\$t9	25	0x00000000
\$k0	26	0x00000000
\$k1	27	0x00000000
\$gp	28	0x10000000
\$sp	29	0x7ffffeffc
\$fp	30	0x00000000
\$ra	31	0x00000000
\$pc		0x00400000
\$hi		0x00000000
\$lo		0x00000000

Mars Messages Run I/O

-- program is finished running (dropped off bottom) --

Reset: reset completed.

Reset: reset completed.



Trạng thái	\$pc	\$s0
Ban đầu	0x00400000	0x00000000
Gđ1	0x00400004	0x00003007
Gđ2	0x00400008	0x00000000

- **Gđ1:**

Op: 8 => 001000

Rs: 0 => 00000

Rt: 16 => 10000

Immi: 0x3007 => 0110 0000 0000 0111

Code: 0010 0000 0001 0000 0110 0000 0000 0111

Mã hex: 0x20103007.

- **Gđ2:**

Op: 0 => 000000

Rs: 0 => 00000

Rt: 0 => 00000

Rd: 16 => 01000

Shamt: 0 => 00000

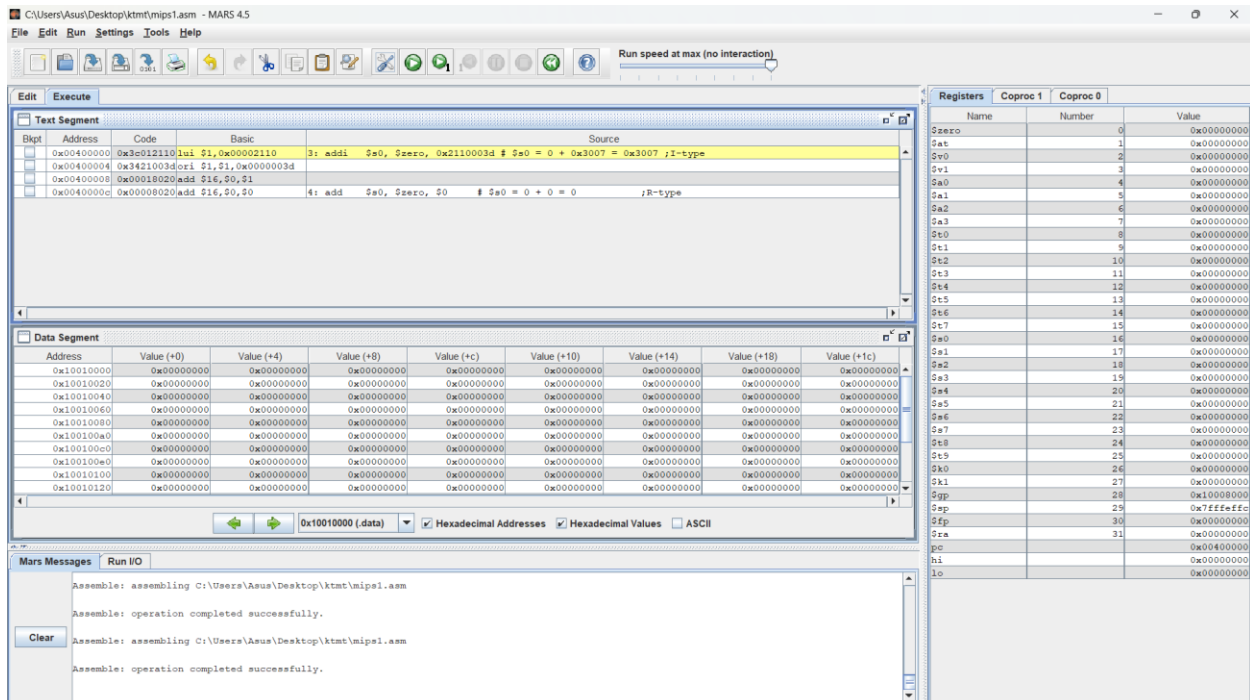
Funt: 32 => 100000

Code: 0000 0000 0000 0000 0100 0000 0010 0000

Mã hex: 0x00008020

⇒ Cửa sổ textsegment trùng với mã máy.

- Sửa lại lệnh addi: `addi $s0, $zero, 0x2110003d`



⇒ Nhận xét: Chương trình không biên dịch lệnh addi, thay vào đó là hai lệnh lui và ori do 0x2110003d là số 32 bit, mà addi có instr. Format I nên chỉ truyền được tối đa 16 bit (từ $2^{(-15)}$ đến $2^{(15)} - 1$). Do đó chương trình phải truyền 16bit cao vào lui và 16bit thấp vào ori để chương trình có thể chạy được.

Assignment 2:

☐ Text Segment

Bkpt	Address	Code	Basic	Source
<input type="checkbox"/>	0x00400000	0x3e102110	lui \$16,0x00002110	2: lui \$s0,0x2110 #put upper half of pattern in \$s0
<input type="checkbox"/>	0x00400000	0x3e10003d	ori \$16,\$16,0x0000003d	3: ori \$s0,\$s0,0x003d #put lower half of pattern in \$s0

Name	Number	Value
\$v0	2	0x00000000
\$v1	3	0x00000000
\$a0	4	0x00000000
\$a1	5	0x00000000
\$a2	6	0x00000000
\$a3	7	0x00000000
\$t0	8	0x00000000
\$t1	9	0x00000000
\$t2	10	0x00000000
\$t3	11	0x00000000
\$t4	12	0x00000000
\$t5	13	0x00000000
\$t6	14	0x00000000
\$t7	15	0x00000000
\$s0	16	0x00000000
\$a1	17	0x00000000
\$a2	18	0x00000000
\$a3	19	0x00000000
\$s4	20	0x00000000
\$s5	21	0x00000000
\$s6	22	0x00000000
\$a7	23	0x00000000
\$t8	24	0x00000000
\$t9	25	0x00000000
\$k0	26	0x00000000
\$k1	27	0x00000000
\$sp	28	0x00000000
\$sp	29	0xffffffff
\$fp	30	0x00000000
\$ra	31	0x00000000
pc		0x00400000
hi		0x00000000

☐ Data Segment

Address	Value (+0)	Value (+4)	Value (+8)	Value (+c)	Value (+10)	Value (+14)	Value (+18)	Valu
0x10010000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0xc
0x10010020	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0xc
0x10010040	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0xc
0x10010060	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0xc
0x10010080	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0xc
0x100100a0	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0xc
0x100100c0	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0xc

☒ Hexadecimal Addresses
 ☒ Hexadecimal Values
 ☐ ASCII

Mars Messages Run I/O

Reset: reset completed.

Trạng thái	\$s0	pc
Ban đầu	0x00000000	0x00400000
Chạy lui	0x21000000	0x00400004
Chạy ori	0x2100003d	0x00400008

- Chuyển sang vùng lệnh .text:

- 0x2 là 16 bit nhưng do li không phải là lệnh thực hiện trực tiếp lên hệ thống nên phải đổi sang câu lệnh khác (addiu) để có thể thực hiện.

Assignment 4:

- Sự thay đổi của các thanh ghi:

Trạng thái	\$t1	\$t2	\$s0	pc
Ban đầu	0x00000000	0x00000000	0x00000000	0x00400000
Chạy addi 1	0x00000005	0x00000000	0x00000000	0x00400004
Chạy addi 2	0x00000005	0xffffffff	0x00000000	0x00400008
Chạy add 1	0x00000005	0xffffffff	0x0000000a	0x0040000c
Chạy add 2	0x00000005	0xffffffff	0x00000009	0x00400010

Nhận xét:

- Ta khai báo $X = 5$ và $Y = -1$, sau khi chạy addi 1 và addi 2 thì ta đã gán $t1 = 5$ và $t2 = -1$, sau khi chạy add 1 thì ta thực hiện phép toán $2X = 10$ và gán vào thanh ghi s0, cuối cùng chạy add 2 là phép toán $2X + Y = 10 - 1 = 9$ và gán vào thanh ghi s0, lúc này là $0x00000009 = 9$, đúng với kết quả của phép tính.
- Ở lệnh addi ta có dạng là addi \$rt, \$rs, immi tương ứng với format I. Ta có 0x2 tương ứng với opcode là 2 là lệnh addi còn vị trí thứ 3 là 9 tương ứng với rt còn vị trí thứ 4 là 0 tương ứng với rs còn 4 bit cuối là 0005 tương ứng với immi.

- Xét: addi \$t2, \$zero, -1 (addi \$10, \$0, 0xffffffff)

Op: 8 => 001000

Rs: 0 => 00000

Rt: 10 => 01010

Imm: 0xffffffff => 1000 0000 0000 0000

Code: 0010 0000 0000 1010 1000 0000 0000 0000

Mã Hex: 0x200affff

➔ Lệnh addi phù hợp với format I.

- Lệnh add thứ nhất: add \$s0, \$t1, \$t1 (add \$16, \$9, \$9)

Opcode : 0 => 000000

Rd : 16 => 10000

Rs : 9 => 10001

Rt : 9 => 10001

Shamt : 0 => 00000

Funt : 32 => 100000

Code: 0000 0001 0010 1001 1000 0000 0010 0000

Mã hex: 0x01298020

- Lệnh add thứ hai: add \$s0, \$s0, \$t2 (add \$16, \$16, \$10)

Opcode : 0 => 000000

Rd : 16 => 10000

Rs : 16 => 10000

Rt : 10 => 01010

Shamt : 0 => 00000

Funt : 32 => 100000

Code: 0000 0010 0000 1010 1000 0000 0010 0000

Mã hex: 0x020a8020

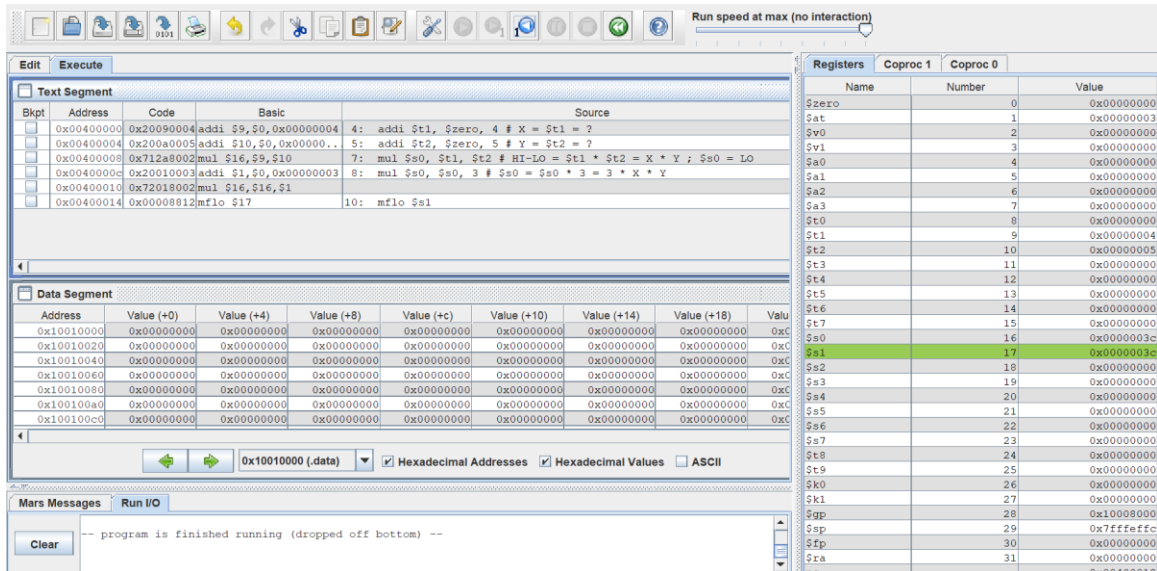
➔ Lệnh add phù hợp với format R.

Assignment 5:

Text Segment				
Bkpt	Address	Code	Basic	Source
<input type="checkbox"/>	0x00400000	0x20090004	addi \$9,\$0,0x00000004	4: addi \$t1, \$zero, 4 # X = \$t1 = ?
<input type="checkbox"/>	0x00400004	0x200a0005	addi \$10,\$0,0x00000005	5: addi \$t2, \$zero, 5 # Y = \$t2 = ?
<input type="checkbox"/>	0x00400008	0x712a8002	mul \$16,\$9,\$10	7: mul \$s0, \$t1, \$t2 # HI-LO = \$t1 * \$t2 = X * Y ; \$s0 = LO
<input type="checkbox"/>	0x0040000c	0x20010003	addi \$1,\$0,0x00000003	8: mul \$s0, \$s0, 3 # \$s0 = \$s0 * 3 = 3 * X * Y
<input type="checkbox"/>	0x00400010	0x72018002	mul \$16,\$16,\$1	
<input type="checkbox"/>	0x00400014	0x00008812	mflo \$17	10: mflo \$s1

Nhận xét: lệnh mul ở dòng 8 bị thay thành lệnh addi do lệnh mul là format R, mà khi ta gõ là format I => Chương trình phải thêm 1 lệnh khác (addi) để chuyển 0x3 thành 1 biến rồi mới thực hiện

- Sau khi chạy xong chương trình:



- Ta có:

Trạng thái	hi	lo
Ban đầu	0x00000000	0x00000000
Chạy add 1	0x00000000	0x00000000
Chạy add 2	0x00000000	0x00000000
Chạy mul 1	0x00000000	0x00000014
Chạy mul 2	0x00000000	0x0000003c

Nhận xét: phép tính $3 * X * Y = 60 = 3c \Rightarrow$ Kết quả cho ra đúng với phép tính.

Assignment 6:

Bkpt	Address	Code	Basic	Source
	0x00400000	0x3c011001	lui \$1, 0x00001001	6: la \$t8, X
	0x00400004	0x34380000	ori \$24, \$1, 0x00000000	
	0x00400008	0x3c011001	lui \$1, 0x00001001	7: la \$t9, Y
	0x0040000c	0x34390004	ori \$25, \$1, 0x00000004	

Nhận xét: Lệnh la chỉ lưu được địa chỉ 16 bit mà địa chỉ của biến là 32 \Rightarrow la sẽ thực hiện 2 câu lệnh lui và ori để lưu địa chỉ của biến.

Edit Execute				
Text Segment				
Bkpt	Address	Code	Basic	Source
<input type="checkbox"/>	0x00400000	0x3c011001	lui \$1,0x00001001	6: la \$t8, X
<input type="checkbox"/>	0x00400004	0x34380000	ori \$24,\$1,0x00000000	
<input type="checkbox"/>	0x00400008	0x3c011001	lui \$1,0x00001001	7: la \$t9, Y
<input type="checkbox"/>	0x0040000c	0x34390004	ori \$25,\$1,0x00000004	
<input type="checkbox"/>	0x00400010	0x8f090000	lw \$9,0x00000000(\$24)	8: lw \$t1, 0(\$t8)
<input type="checkbox"/>	0x00400014	0x8f2a0000	lw \$10,0x00000000(\$24)	9: lw \$t2, 0(\$t9)
<input type="checkbox"/>	0x00400018	0x01298020	add \$16,\$9,\$9	10: add \$s0, \$t1, \$t1
<input type="checkbox"/>	0x0040001c	0x020a8020	add \$16,\$16,\$10	11: add \$s0, \$s0, \$t2
<input type="checkbox"/>	0x00400020	0x3c011001	lui \$1,0x00001001	12: la \$t7, Z
<input type="checkbox"/>	0x00400024	0x342f0008	ori \$15,\$1,0x00000008	
<input type="checkbox"/>	0x00400028	0xadf00000	sw \$16,0x00000000(\$24)	13: sw \$s0, 0(\$t7)

Label	Address
week2.6	
X	0x10010000
Y	0x10010004
Z	0x10010008

Edit Execute				
Text Segment				
Bkpt	Address	Code	Basic	Source
<input type="checkbox"/>	0x00400000	0x3c011001	lui \$1,0x00001001	6: la \$t8, X
<input type="checkbox"/>	0x00400004	0x34380000	ori \$24,\$1,0x00000000	
<input type="checkbox"/>	0x00400008	0x3c011001	lui \$1,0x00001001	7: la \$t9, Y
<input type="checkbox"/>	0x0040000c	0x34390004	ori \$25,\$1,0x00000004	
<input type="checkbox"/>	0x00400010	0x8f090000	lw \$9,0x00000000(\$24)	8: lw \$t1, 0(\$t8)
<input type="checkbox"/>	0x00400014	0x8f2a0000	lw \$10,0x00000000(\$24)	9: lw \$t2, 0(\$t9)
<input type="checkbox"/>	0x00400018	0x01298020	add \$16,\$9,\$9	10: add \$s0, \$t1, \$t1
<input type="checkbox"/>	0x0040001c	0x020a8020	add \$16,\$16,\$10	11: add \$s0, \$s0, \$t2
<input type="checkbox"/>	0x00400020	0x3c011001	lui \$1,0x00001001	12: la \$t7, Z
<input type="checkbox"/>	0x00400024	0x342f0008	ori \$15,\$1,0x00000008	
<input type="checkbox"/>	0x00400028	0xadf00000	sw \$16,0x00000000(\$24)	13: sw \$s0, 0(\$t7)

Label	Address
week2.6	
X	0x10010000
Y	0x10010004
Z	0x10010008

- Sau 3 lần chạy la thì ta thấy địa chỉ X, Y, Z bằng hằng số:

Data Segment	
Address	Value (+0)
0x10010000	0x00000005
0x10010020	0x00000000
0x10010040	0x00000000
0x10010060	0x00000000

Biến X

Data Segment				
Address	Value (+0)	Value (+4)	Value (+8)	
0x10010000	0x00000005	0xffffffff	0x00000009	
0x10010020	0x00000000	0x00000000	0x00000000	
0x10010040	0x00000000	0x00000000	0x00000000	
0x10010060	0x00000000	0x00000000	0x00000000	
0x10010080	0x00000000	0x00000000	0x00000000	
0x100100a0	0x00000000	0x00000000	0x00000000	
0x100100c0	0x00000000	0x00000000	0x00000000	
0x100100e0	0x00000000	0x00000000	0x00000000	
0x10010100	0x00000000	0x00000000	0x00000000	
0x10010120	0x00000000	0x00000000	0x00000000	
0x10010140	0x00000000	0x00000000	0x00000000	
0x10010160	0x00000000	0x00000000	0x00000000	
0x10010180	0x00000000	0x00000000	0x00000000	
0x100101a0	0x00000000	0x00000000	0x00000000	

Biến Y

Address	Value (+0)	Value (+4)	Value (+8)	Value (+c)
0x10010000	0x00000005	0xffffffff	0x00000009	0x00000000
0x10010020	0x00000000	0x00000000	0x00000000	0x00000000
0x10010040	0x00000000	0x00000000	0x00000000	0x00000000
0x10010060	0x00000000	0x00000000	0x00000000	0x00000000
0x10010080	0x00000000	0x00000000	0x00000000	0x00000000
0x100100a0	0x00000000	0x00000000	0x00000000	0x00000000
0x100100c0	0x00000000	0x00000000	0x00000000	0x00000000

Biến Z

- Ta có sự thay đổi giá trị của thanh ghi:

Trạng thái	\$s0	\$t1	\$t2	\$t7	\$t8	\$t9
Ban đầu	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
Chạy la 1					0x10010000	
Chạy la 2						0x10010004
Chạy lw 1		0x00000005				
Chạy lw 2			0xffffffff			

Chạy add 1	0x0000000a					
Chạy add 2	0x00000009					
Chạy la				0x10010008		
Chạy sw						

➤ Nhận xét:

- lw: dùng để lưu giá trị trên địa chỉ vào địa chỉ còn lại.
- sw: dùng để lưu giá trị trên thanh ghi vào 1 địa chỉ.