Extract: and s0 với 0xFF000000

Clear: and với 1 giữ nguyên, and với 0 = 0

Set: or, bit or với 1 = 1, or với 0 bằng chính nó

Xor với chính nó

Kiểm tra tổng với 1 trong 2 toán hạng có cùng dấu k? Cùng thì ko bị tràn, khác là tràn

Tính tổng xong, kiểm tra 2 toán hạng trùng dấu.

Kiểm tra 1 toán hạng với tổng trùng dấu k, dùng phép xor (trái dấu tràn số)

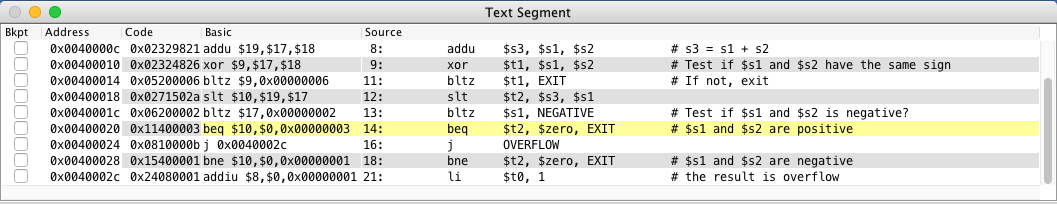
5. Sử dụng lệnh dịch để thực hiện phép nhân với luỹ thừa của 2

**Laboratory Exercise 4**

Arithmetic and Logical operation

**# Assignment 1:**

TH1: Cộng 2 số dương không tràn số****

****

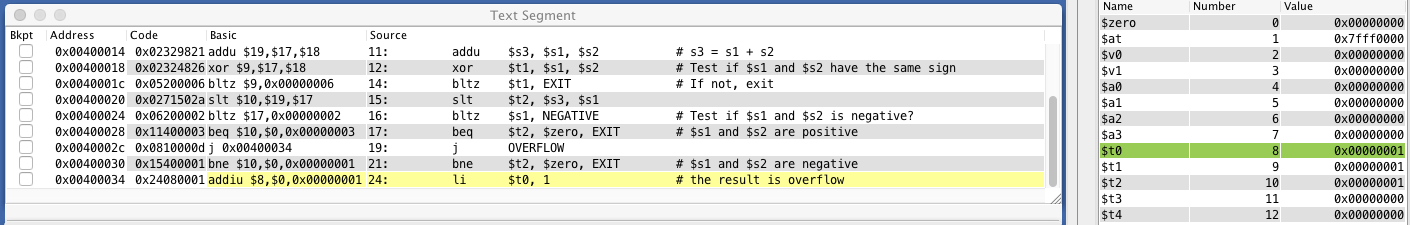
Chương trình sẽ thoát khi gặp lệnh beq, sum ($s3) = 18 (0x00000012)

TH2: Cộng 2 số dương tràn số



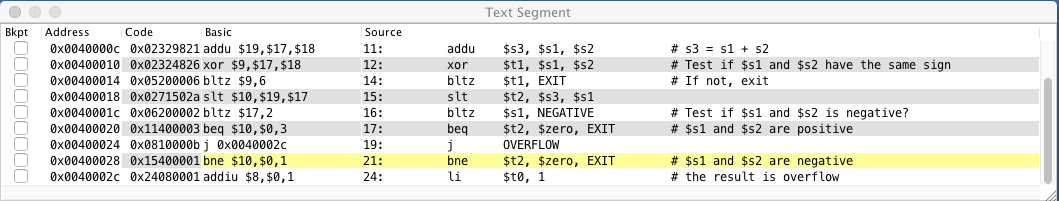
Giá trị sum thu được = -3



Kết quả thanh ghi $t0 (overflow) = 1

TH3: Cộng 2 số âm không tràn số





Chương trình sẽ thoát khi gặp lệnh bne, sum = -3

TH4: Cộng 2 số âm tràn số



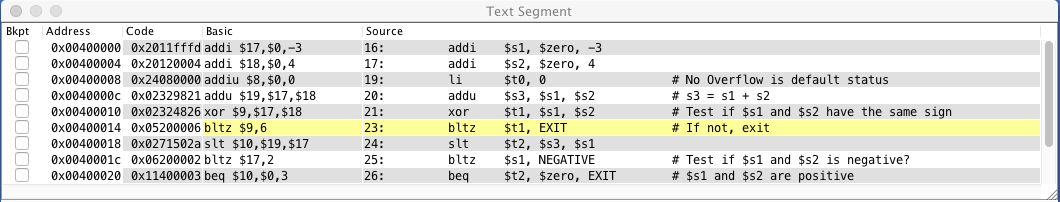
Giá trị sum thu được = 1



Giá trị thanh ghi $t0 = 1 (overflow)

TH5: Cộng 2 số trái dấu

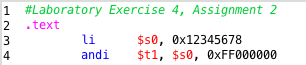




Chương trình bị thoát ngay khi gặp lệnh bltz, giá trị thanh ghi $s3 (sum) = 1 .

**# Assignment 2:**

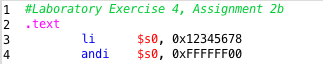
1. Extract MSB of $s0



MSB của $s0 được lưu vào thanh ghi $t1



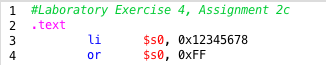
1. Clear LSB of $s0



LSB của $s0 bị reset về 0



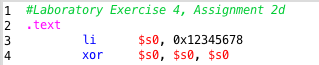
1. Set LSB of $s0 (bits 7 to 0 are set to 1)



LSB của $s0 được set = 1



1. Clear $s0 ($s0=0, must use logical instructions) : thực hiện phép xor $s0 với chính nó



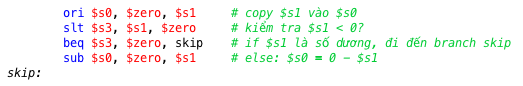
Giá trị $s0 bị clear về 0:



**# Assignment 3:**

1. **abs $s0, $s1 :** Lưu giá trị tuyệt đối của $s1 vào $s0

Tương đương:

****

Lúc này giá trị tuyệt đối thanh ghi $s1 được lưu trong thanh ghi $s0

1. **move $s0, $s1 :** Tăng giá trị của thanh ghi $s0 một lượng bằng giá trị thanh ghi $s1

Tương đương:



1. **not $s0, $s1 :** Thanh ghi $s0 bằng đảo bit thanh ghi $s1

Tương đương: Giá trị $s0 bằng nor $s1 và 0



1. **ble $s1, $s2, label** : Nếu $s1 <= $s2 thì đi đến branch “label”

Tương đương:

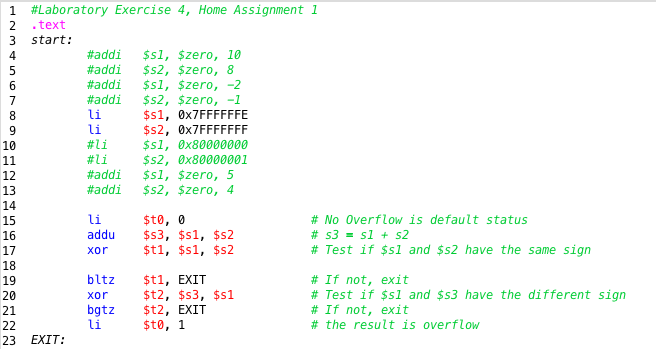
slt kiểm tra nếu $s2 < $s1 thì $t1 = 1

beq kiểm tra nếy $t1 = 0 thì đi đến branch label (tức là trường hợp $s1 <= $s2)

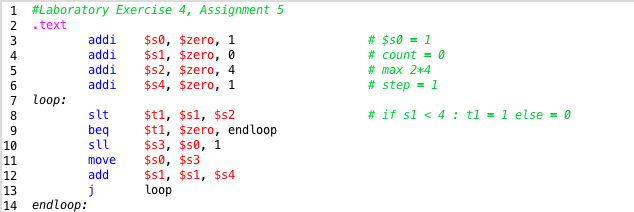
**# Assignment 4:**

Tính tổng xong, kiểm tra 2 toán hạng trùng dấu. (Trái dấu thì không tràn số, exit)

Kiểm tra 1 toán hạng với tổng trùng dấu k, dùng phép xor (trái dấu tràn số)

****

**# Assignment 5:**

****

Kết quả sẽ được lưu vào thanh ghi $s3

**Conclusion:**

sll $t1,$t2,5 (Shift Left Logical): Dịch sang trái giá trị của thanh ghi $t2 theo số bit (5) của immediate rồi lưu vào thanh ghi $t1

sllv $t1,$t2,$t3 (Shift Left Logical Variable): Dịch sang trái giá trị của thanh ghi $t2 theo số bit (giá trị của 5 bit thấp trong thanh ghi $t3) rồi lưu vào thanh ghi $t1

srl $t1,$t2,5 (Shift Right Logical): Dịch sang phải giá trị của thanh ghi $t2 theo số bit của immediate rồi lưu vào thanh ghi $t1

srlv $t1,$t2,$t3 (Shift Right Logical Variable): Dịch sang phải giá trị của thanh ghi $t2 theo số bit (giá trị của 5 bit thấp trong thanh ghi $t3) rồi lưu vào thanh ghi $t1