Assembly Language and Computer Architecture Lab

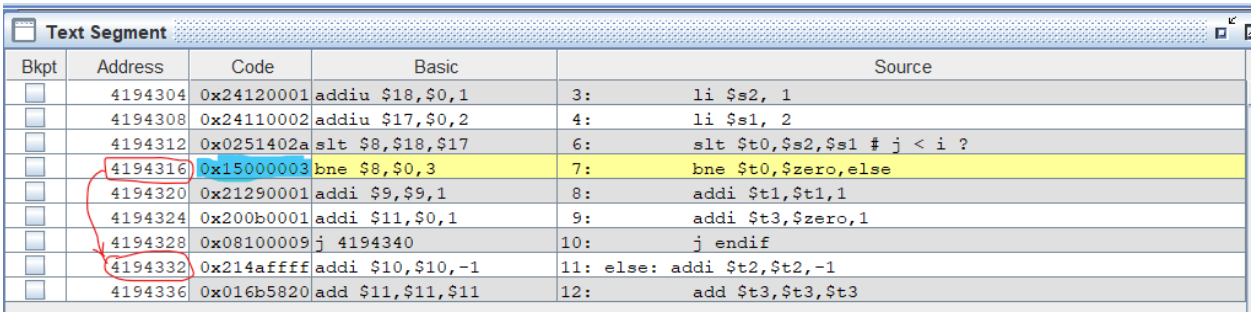
Week 3

Assignment 4

- Program Counter (PC) is the register that is affected by jump/branch instructions.

- Below are explainations of calculating target addresses:

Branch instruction (bne, beq)



Text Segment

Bkpt	Address	Code	Basic	Source
	4194304	0x24120001	addiu $18,$0,1	3: li $s2, 1
	4194308	0x24110002	addiu $17,$0,2	4: li $s1, 2
	4194312	0x0251402a	slt $8,$18,$17	6: slt $t0,$s2,$s1 # j < i ?
	4194316	0x15000003	bne $8,$0,3	7: bne $t0,$zero,else
	4194320	0x21290001	addi $9,$9,1	8: addi $t1,$t1,1
	4194324	0x200b0001	addi $11,$0,1	9: addi $t3,$zero,1
	4194328	0x08100009	j 4194340	10: j endif
	4194332	0x214affff	addi $10,$10,-1	11: else: addi $t2,$t2,-1
	4194336	0x016b5820	add $11,$11,$11	12: add $t3,$t3,$t3

The **bne** instruction has machine code **0x15000003** and address **4194316**.

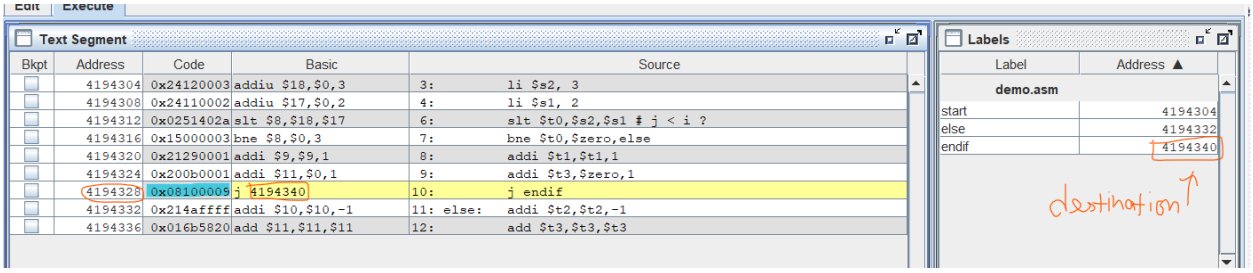
Let analyse its machine code:

- ➔ Convert to binary: **0001 0101 0000 0000 [0000 0000 0000 0011]** (the last 16 bits are *jump addr).*
- ➔ Get *jump addr* and left shift 2 bits: **0000 0000 0000 0011 00**
- ➔ Sign-extended to 32 bits: **00 0000 0000 0000 0000 0000 0000 0011 00**
- ➔ Add $4_{(10)}$ for the next instruction, we then get:

  0000 0000 0000 0000 0000 0000 0001 0000$_{(2)}$ = 16$_{(10)}$

- ➔ Add **16** to the address in base 10, we have the next instruction: **4194316 + 16 = 4194332**

Jump instruction (j)



The **j** instruction has machine code **0x08100009** and address **4194328**.

Let analyse its machine code:

- ➔ Convert to binary: **0000 10[00 0001 0000 0000 0000 0000 1001]** (the last 26 bits are *jump addr).*
- ➔ Get *jump addr* and left shift 2 bits: **00 0001 0000 0000 0000 0000 1001 00 = $12_{(10)}$**
- ➔ Current address: **$4194328_{(10)}$ = 0000 0000 0100 0000 0000 0000 0001 $1000_{(2)}$**
- ➔ Get first 4 bits of current address to get 32 bits address of destination:

  0000 00 0001 0000 0000 0000 0000 1001 00 = $4194340_{(10)}$