معماری کامپیوتری

نیم‌سال دوم ۰۳-۰۲

دانشکده‌ی مهندسی کامپیوتر
استاد: دکتر لاله ارشدی
تمرین تئوری چهارم
پاسخ‌دهنده: معین آعلی - ۴۰۱۱۰۵۵۶۱

پاسخ مسئله‌ی ۱.

$a : \text{۰}xAC\text{۶۲۰۰۱۴}$

$b : \text{۰}x\text{۰۰۸۲۰۸۲}A$

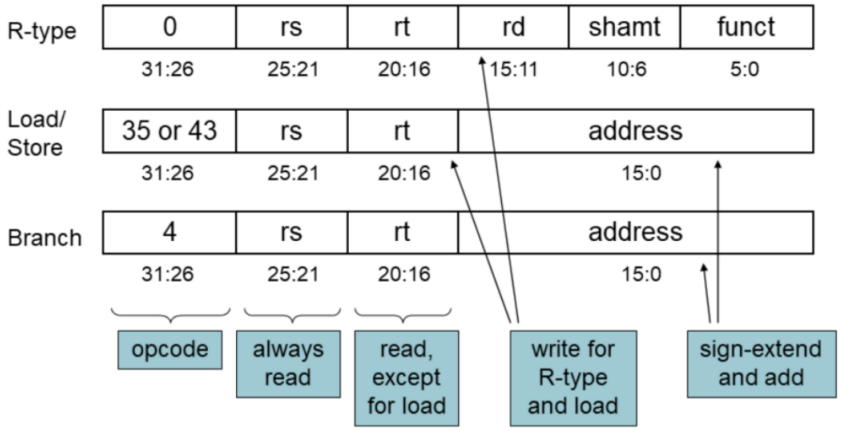
الف

در این باید کد اسمبلی معادل با هر دستور را بنویسیم. برای این کار ابتدا اعداد را به باینری تبدیل می‌کنیم:

$a = (\text{۱۰۱۰۱۱۰۰۰۱۱۰۰۰۱۰۰۰۰۰۰۰۰۰۰۱۰۱۰۰})_{\text{۲}}$

$b = (\text{۰۰۰۰۰۰۰۰۱۰۰۰۰۰۱۰۰۰۰۰۱۰۰۰۰۰۱۰۱۰۱۰})_{\text{۲}}$

حال از این تصویر استفاده کرده و دستورات را شناسایی می‌کنیم:



$opcode_a = a[\text{۳۱} : \text{۲۶}] = (\text{۱۰۱۰۱۱})_{\text{۲}} = (\text{۴۳})_{\text{۱۰}} \longrightarrow store$

$$\begin{cases} r_s = a[\text{۲۵} : \text{۲۱}] = (\text{۰۰۰۱۱})_{\text{۲}} = (\text{۳})_{\text{۱۰}} \longrightarrow v_{\text{۱}} \\ r_t = a[\text{۲۰} : \text{۱۶}] = (\text{۰۰۰۱۰})_{\text{۲}} = (\text{۲})_{\text{۱۰}} \longrightarrow v_{\text{۰}} \\ immediate = a[\text{۱۵} : \text{۰}] = (\text{۰۰۰۰۰۰۰۰۰۰۰۱۰۱۰۰})_{\text{۲}} = (\text{۲۰})_{\text{۱۰}} \end{cases} \longrightarrow sw \ \ \$v_{\text{۰}}, \text{۲۰}(\$v_{\text{۱}})$$

$opcode_b = b[\text{۳۱} : \text{۲۶}] = (\text{۰۰۰۰۰۰})_{\text{۲}} = (\text{۰})_{\text{۱۰}} \longrightarrow R - Type$

$$\begin{cases} r_s = b[\text{۲۵} : \text{۲۱}] = (\text{۰۰۱۰۰})_{\text{۲}} = (\text{۴})_{\text{۱۰}} \longrightarrow a_{\text{۰}} \\ r_t = b[\text{۲۰} : \text{۱۶}] = (\text{۰۰۰۱۰})_{\text{۲}} = (\text{۲})_{\text{۱۰}} \longrightarrow v_{\text{۰}} \\ r_d = b[\text{۱۵} : \text{۱۱}] = (\text{۰۰۰۰۱})_{\text{۲}} = (\text{۱})_{\text{۱۰}} \longrightarrow a_t \\ function = b[\text{۵} : \text{۰}] = (\text{۱۰۱۰۱۰})_{\text{۲}} \longrightarrow set\ on\ less\ than \end{cases} \longrightarrow slt \ \ \$a_t \ , \ \$a_{\text{۰}} \ , \ \$v_{\text{۰}}$$

ب

خروجی sign extend دستور a ، ۱۶ بیت اول دستور را به یک عدد ۳۲ بیتی تبدیل می‌کند با حفظ علامت:

$$output = (\text{۰۰۰۰۰۰۰۰۰۰۰۰۰۰۰۰۰۰۰۰۰۰۰۰۱۰۱۰۰})_{۲} = (۱۴)_{۱۶}$$

خروجی واحد shift left که توسط دستور b اجرای می‌شود، خروجی واحد sign extend را می‌گیرد و ۲ بیت به چپ شیفت می‌دهد.

$$output = (\text{۰۱۰۰۰۰۰۱۰۰۰۰۰۱۰۰۰۰۰۱۰۱۰۱۰۰۰})_{۲} = (۲۰۸۲۰A۸)_{۱۶}$$

ج

در این بخش باید مقادیر واحد کنترل ALU را به ازای هر دستور محسابه کنیم. از تصویر زیر کمک می‌گیریم:

opcode	ALUOp	Operation	funct	ALU function	ALU control
lw	00	load word	XXXXXX	add	0010
sw	00	store word	XXXXXX	add	0010
beq	01	branch equal	XXXXXX	subtract	0110
R-type	10	add	100000	add	0010
		subtract	100010	subtract	0110
		AND	100100	AND	0000
		OR	100101	OR	0001
		set-on-less-than	101010	set-on-less-than	0111

$$a \rightarrow lw \rightarrow \begin{cases} ALUOp = ۰۰ \\ funct = a[۵:۰] = \times \end{cases}$$

$$b \rightarrow RType \rightarrow \begin{cases} ALUOp = ۱۰ \\ funct = b[۵:۰] = ۱۰۱۰۱۰ \end{cases}$$

د

با توجه به این‌که هیچ کدام از دستورات مربوط به دستورات jump و یا branch نیستند، پس در هر دو حالت مقدار برابر است با:

$$PC \leftarrow PC + ۴$$

در شکل زیر مسیرهایی که از آن‌ها این مقادیر به‌دست می‌آیند را می‌بینید: