

# Computer Structure and Language

Hamid Sarbazi-Azad

Department of Computer Engineering  
Sharif University of Technology (SUT)  
Tehran, Iran



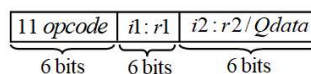
(c) Hamid Sarbazi-Azad

Computer Structure & Language -- Lecture#6: Answering Questions

2

یک کامپیوتر دو آدرس دارای حافظه‌ای به گنجایش  $2^{14}$  واحد آدرس پذیر ۶ بیتی، طول کلمه ۱۸ بیتی و ۳۲ ثبات همه‌منظوره R0 تا R31 می‌باشد. شیوه‌های نشاندهی ماشین شامل ثباتی (مستقیم و غیرمستقیم)، بلافاصله و مستقیم حافظه‌ای، و شیوه نمایش اعداد مکمل ۲ است. دستورات در سه قالب (طبق جداول زیر) کد میشوند.

## Format I



If *i1* then *opr1* = *M<sub>(r1)</sub>* else *opr1* = *r1*;  
If *i2* then *opr2* = *M<sub>(r2)</sub>* else *opr2* = *r2*;

Instruction	Opcode	Operation
<u>mov</u> <u>opr1,opr2</u>	0000	$opr1 \leftarrow (opr2);$
<u>add</u> <u>opr1,opr2</u>	0001	$opr1 \leftarrow (opr1) + (opr2);$
<u>sub</u> <u>opr1,opr2</u>	0010	$opr1 \leftarrow (opr1) - (opr2);$
<u>and</u> <u>opr1,opr2</u>	0011	$opr1 \leftarrow (opr1) \wedge (opr2);$
<u>or</u> <u>opr1,opr2</u>	0100	$opr1 \leftarrow (opr1) \vee (opr2);$
<u>xor</u> <u>opr1,opr2</u>	0101	$opr1 \leftarrow (opr1) \oplus (opr2);$
<u>swap</u> <u>opr1,opr2</u>	0110	$(opr1) \leftrightarrow (opr2);$
<u>addq</u> <u>opr1,Qdata</u>	0111	$opr1 \leftarrow (opr1) + Qdata;$
<u>subq</u> <u>opr1,Qdata</u>	1000	$opr1 \leftarrow (opr1) - Qdata;$
<u>movq</u> <u>opr1,Qdata</u>	1001	$opr1 \leftarrow Qdata;$

(c) Hamid Sarbazi-Azad      Computer Structure & Language -- Lecture#6: Answering Questions      3		
<div> <div>Format II</div> <div> <div>10</div> <div>opcode</div> <div>5 bits</div> </div> <div> <div>r</div> <div>5 bits</div> </div> <div> <div>address</div> <div>14 bits</div> </div> </div>		
<b>Instruction</b>	<b>Opcode</b>	<b>Operation</b>
<u>mov</u> r, address	000	$r \leftarrow (M_{address});$
<u>mov</u> address, r	001	$M_{address} \leftarrow (r);$
<u>jnz</u> r, address	010	if $(r) \neq 0$ then $PC \leftarrow address;$
<u>jz</u> r, address	011	if $(r) = 0$ then $PC \leftarrow address;$
<u>jneg</u> r, address	100	if $(r) < 0$ then $PC \leftarrow address;$
<u>jpos</u> r, address	101	if $(r) \geq 0$ then $PC \leftarrow address;$
loop r, address	110	$r \leftarrow (r) - 1;$ if $(r) \neq 0$ then $PC \leftarrow address;$
<u>jmp</u> r, address	111	$r \leftarrow (r) + 1;$ $PC \leftarrow address;$

<div> <div>Format III</div> <div> <div>0</div> <div>r</div> <div>5 bits</div> </div> <div> <div>data</div> <div>18 bits</div> </div> </div>		
<b>Instruction</b>	<b>Opcode</b>	<b>Operation</b>
<u>mov</u> r, #data	0	$r \leftarrow data;$

(c) Hamid Sarbazi-Azad      Computer Structure & Language -- Lecture#6: Answering Questions      4		
<p>۳- برنامه زیر چه می کند؟ مقادیر اولیه ثباتها مثبت است. (۳ نمره)</p> <p>۴- برنامه سوال ۳ را به کد ماشین ترجمه کنید. (۲ نمره)</p>		
	<b>org</b>	<b>0</b>
	<u>movq</u>	R1,0
	<u>add</u>	R1,R2
aaa:	<u>dw</u>	370FFh
	<u>jnz</u>	R3,3
sum:	<u>dw</u>	32105h
	<u>addq</u>	R3,1
	<u>mov</u>	R6,R4
	<u>sub</u>	R6,R5
array:	<u>dw</u>	2A600h,0DB80h,740h
	<u>mov</u>	R7,(R0)
	<u>addq</u>	(R7),-3
	<u>mov</u>	R5,#-2
	<b>end</b>	

(c) Hamid Sarbazi-Azad		Computer Structure & Language -- Lecture#6: Answering Questions		5
Address Machine Code		Assembly Code		Symbol Table
		<b>org</b>	0	Symbol    Address
0000	39040 [111001 000001 000000]	movq	R1,0	aaa        0006h
0003	31042 [110001 000001 000010]	add	R1,R2	sum        000Dh
0006	370FF [110111 000011 111111]	aaa: <b>dw</b>	370FFh	array       0019h
	== addq R3,-1			
0009	90C003 [10010 00011 00003]	jnz	R3,3	
000D	32105 [11 0010 0001 0000 0101]	sum: <b>dw</b>	32105h	
	== sub R4,R5			
0010	370C1 [110111 000011 000001]	addq	R3,1	
0013	30184 [110000 000110 000100]	mov	R6,R4	
0016	32185 [110010 000110 000101]	sub	R6,R5	
0019	2A600 0DB80 00740	array: <b>dw</b>	2A600h,0DB80h,740h	
0019	2A600 == jpos R6,000Dh (=sum)		[10 1010 0110 0000 0000]	
001C	0DB80 == jmp R0,001Dh		[00 1101 1011 1000 0000]	
001F	00740		[00 0000 0111 0100 0000]	
0022	301E0 [110000 000111 100000]	mov	R7,(R0)	
0025	379FD [110111 100111 111101]	addq	(R7),-3	
0028	17FFE [000101 11FFE]	mov	R5,#-2	
		<b>end</b>		
Format I		Format II		Format III
11 opcode	i1:r1	10 opcode	r	0 r
6 bits	6 bits	5 bits	5 bits	5 bits
i2:r2/Qdata		address		data
6 bits		14 bits		18 bits

**END OF SLIDES**