Chapter 2 Exercises

Op(31:26)								
28-26 / 31-29	0(000)	1(001)	2(010)	3(011)	4(100)	5(101)	6(110)	7(111)
0(000)	R-format	Bltz/gez	Jump	Jump &link	Branch eq	Branch ne	blez	bgtz
1(001)	Add immediate	Addiu	Set less than imm.	Set less than imm. Unsigned	andi	ori	xori	Load upper imm.
2(010)	TLB	FlPt						
3(011)								
4(100)	Load byte	Load half	Lwl	Load word	Load byte unsigned	Load half unsigned	lwr	
5(101)	Store byte	Store half	Swl	Store word			swr	
6(110)	Load linked word	lwcl						
7(111)	Store cond. word	swcl						

Op(31:26)=000000 (R-format), funct(5:0)								
2-0 / 5-3	0(000)	1(001)	2(010)	3(011)	4(100)	5(101)	6(110)	7(111)
0(000)	Shift left logical		Shift right logical	sra	sllv		srlv	srav
1(001)	jump register	jalr			Syscall	Break		
2(010)	mfhi	mthi	mflo	mtlo				
3(011)	Mult	Multu	div	Divu				
4(100)	Add	Addu	Subtract	Subu	And	Or	Xor	Not or (nor)
5(101)			Set l.t.	Set l.t. unsigned				
6(110)								
7(111)								

Figure 2.19 MIPS instruction encoding.

Name	Fields						Comments
Field size	6 bits	5 bits	5 bits	5 bits	5 bits	6 bits	All MIPS instructions are 32 bits long
R-format	op	rs	rt	rd	shamt	funct	Arithmetic instruction format
I-format	op	rs	rt	address/immediate			Transfer, branch, imm. format
J-format	op	target address					Jump instruction format

Figure 2.20 MIPS instruction formats.

Name	Register Number	Usage	Preserve on call?
$zero	0	constant 0 (hardware)	n.a.
$at	1	reserved for assembler	n.a.
$v0 - $v1	2-3	returned values	no
$a0 - $a3	4-7	arguments	yes
$t0 - $t7	8-15	temporaries	no
$s0 - $s7	16-23	saved values	yes
$t8 - $t9	24-25	temporaries	no
$gp	28	global pointer	yes
$sp	29	stack pointer	yes
$fp	30	frame pointer	yes
$ra	31	return addr (hardware)	yes

Review Question

1. Explain "Store-program concept " using less than one hundred English words

2. <p.98, 2.5, book example> Assume $t1 has the base of the array A and $s2 corresponds to h, compile the following assignment statement into MIPS instruction

A[300]= h+ A[300]

Ans:

2.1<2.2>The following problems explore translating from C to MIPS. Assume that the variables f, g, h, and i are given and could be considered 32-bit integers as declared in a C program.

a) f = g + h + i + j ;

b) f = g + (h + 5) ;

2.1.1 [5]<2.2> For the C statements above, what is the corresponding MIPS assembly code? Use a minimal number of MIPS assembly instructions.

2.1.3 [5J <2.2> If the variables f, g, h, i, j have values 1, 2, 3, 4, and 5, respectively, what is the end value of f?

The following problems deal with translating from MIPS to C. Assume that the variables g, h, I, and j are given and could be considered 32-bit integers as declared in a C program

a) add f, f, h

b) sub f, $0, f
   addi f, f, 1

2.2.4<2.2> For the MIPS statements above, what is a corresponding C statement?

2.2.5<2.2> If the variables f, g, h, and i have values, 1, 2, 3, and 4, respectively, what is the end value of f?

2.4. The following problems deal with translating from C to MIPS. Assume that the variables f, g, h, i, and j are assigned to registers $s0, $sl, $s2 , $s3, and$s4 , respectively. Assume that the base address of the arrays A and B are in registers $s6 and $s7, respectively.

a. f = g + h + B[4];

b. f = g – A[B[4]];

2.4.1 [10] <2.2,2.3> For the C statements above, what is the corresponding MIPS assembly code?

2.4.2 [5] <2.2, 2.3> For the C statements above, how many MIPS assembly instructions are needed to perform the C statement?

2.4.3 [5] <2.2, 2.3> For the C statements above, how many different registers are needed to carry out the C statement?

2.7

The following problems explore number conversions from signed and unsigned binary numbers to decimal numbers.

a. $1010\ 1101\ 0001\ 0000\ 0000\ 0000\ 0000\ 0010_{two}$

b. $1111\ 1111\ 1111\ 1111\ 1011\ 0011\ 0101\ 0011_{two}$

2.7.1 [5] <2.4> For the patterns above, what base 10 number does the binary number represent, assuming that it is a two complement integer?

2.7.2 [5] <2.4> For the patterns above, what base 10 number does the binary number represent, assuming that it is an unsigned integer?

2.7.3 [5] <2.4> For the patterns above, what hexadecimal number does it represent?

2.10. In the following problems, the data table contains bits that represent the opcode of an instruction. You will be asked to interpret the bits as MIPS instructions into assembly code and determine what format of MIPS instruction the bits represent.

a. $1010\ 1110\ 0000\ 1011\ 0000\ 0000\ 0000\ 0100_{two}$

b. $1000\ 1101\ 0000\ 1000\ 0000\ 0000\ 0100\ 0000_{two}$

2.10.1 [5] <2.5> For the binary entries above, what instruction do they represent?

2.10.2 [5] <2.5> What type (I-type, R-type,J-type) instruction do the binary entries above represent?

2.10.3 [5] <2.4, 2.5> If the binary entries above were data bits, what number would they represent in hexadecimal?

In the following problems, the data contains MIPS instructions. You will be asked to translate the entries into the bits of the opcode and determine what is the MIPS instruction format.

a. add $t0, $t0, $zero

b. lw $t1, 4($s3)

2.10.4 <2.4, 2.5> For the instructions above, show the hexadecimal representation of these instructions

2.10.5<2.5> What type (I-type, R-type) instruction do the instructions above represent?

2.10.6<2.5> What is the hexadecimal representation of the opcode, rs, and rt fields in this instructions? For R-type instructions, what is the hexadecimal representation of the rd and funct fields? For I-type Instructions, what is the hexadecimal representation of the immediate field?