Homework4

yxu66

2.8 “Translate 0xabcdef12 into decimal.”

0xabcdef12(hex)
= 10*16^7 + 11*16^6 + 12*16^5 + 13*16^4 + 14*16^3 + 15*16^2 + 1*16^1 + 2*16^0
= 2882400018(decimal)

2.11 “For each MIPS instruction, show the value of the opcode (OP), source register (RS), and target register (RT) fields. For the I-type instructions, show the value of the immediate field, and for the R-type instructions, show the value of the destination register (RD) field.”

All value are decimal.
(1). addi $t0, $s6, 4
Format: I
R[rt] = R[rs] + SignExtImm

So OP = 8, RS = 22, RT = 8, immediate field = 4

(2). add $t1, $s6, $0
Format: R
R[rd] = R[rs] + R[rt]

So OP = 0, RD = 9, RS = 22, RT = 0

(3). sw $t1, 0($t0)
Format: I
M[R[rs]+SignExtImm] = R[rt]

So OP = 43, RS = 8, immediate field = 0, RT = 9

(4). lw $t0, 0($t0)
Format: I
R[rt] = M[R[rs]+SignExtImm]

So OP = 35, RT = 8, RS = 8, immediate field = 0

(5). add $s0, $t1, $t0
Format: R

R[rd] = R[rs] + R[rt]

So OP = 0, RD = 16, RS = 9, RT = 8

2.12.3 "For the contents of registers $s0 and $s1 as specified above, what is the value of $t0 for the following assembly code? sub $t0, $s0, $s1"

"Assume that registers $s0 and $s1 hold the values 0x80000000 and 0xD0000000, respectively."

$s0 has the value: 0x80000000(hex) = 8*16^7 = 2147483648(decimal)
$s1 has the value: 0xD0000000(hex) = 13*16^7 = 3489660928(decimal)
So $t0 = 2147483648 - 3489660928 = -1342177280(decimal) = -0x50000000(hex)

2.14 "Provide the type and assembly language instruction for the following binary value: 0000 0010 0001 0000 1000 0000 0010 0000two"

From 0000 0010 0001 0000 1000 0000 0010 0000two we can get :
000000(OP) 10000(RS) 10000(RT) 10000(RD) 00000(shamt) 100000(funct)
So: OP = 0, RS = 2^4=16, RT = 16, RD = 16, shamt = 0, FUNCT = 2^5 = 32
So it's format is R and according to the green sheet:
OP = 0 and FUNCT = 32(decimal) = 20(hex), that's add, R-type.
So assembly language instruction is: add $s0, $s0, $s0

2.15 "Provide the type and hexadecimal representation of following instruction: sw $t1, 32($t2)"

According to the green sheet: the type is I.
M[R[rs]+SignExtImm] = R[rt]

So OP = 43, RS = 10, SignExtImm = 32, RT = 9

Then we got: 101011 01010 01001 0000000000100000 (two) -> 1010 1101 0100 1001 0000 0000 0010 0000(hex) -> 0xad49 0020

2.16 "Provide the type, assembly language instruction, and

binary representation of instruction described by the following MIPS fields:

op=0, rs=3, rt=2, rd=3, shamt=0, funct=34”

According to the green sheet: funct = 34(decimal) = 22(hex)

So it’s sub: R[rd] = R[rs] - R[rt], type is R.

Assembly language instruction is: sub $v1, $v1, $v0

Binary representation: 000000 00011 00010 00011 00000 100010

2.17 “Provide the type, assembly language instruction, and binary representation of instruction described by the following MIPS fields:

op=0x23, rs=1, rt=2, const=0x4”

Acoording to the green sheet, it’s beq, I-type.

if(R[rs]==R[rt]) 4_{hex}

PC=PC+4+BranchAddr

OP = 35, const = 4 It’s lw, R-type.

lw $v0, 4($at)

100011 00001 00010 0000000000000100

2.21 “Provide a minimal set of MIPS instructions that may be used to implement the following pseudoinstruction:

not $t1, $t2 // bit-wise invert”

addi $t0, $t0, 1

sub $t1, $t0, $t2

1. Convert 1234 from decimal into binary

1234 = 2^10 + 2^7 + 2^6 + 2^4 + 2 = 10011010010

2. Convert 4321 from decimal into hexadecimal

4321%16 = 1

4321 / 16 = 270

270 % 16 = 14
270 / 16 = 16
16 % 16 = 0
16 / 16 = 1
1 % 16 = 1
1 / 16 = 0
So it's 0x10e1

3. Convert 0110 1011 from binary into decimal
2^6+2^5+2^3+2^1+2^0 = 107

4. Convert 0x1234 from hexadecimal into decimal
1*16^3 + 2*16^2 + 3*16^1 + 4*16^0 = 4660

5. In an 8-bit system find the two's complement representation of the decimal value -21.
21 = 0001 0101(binary in 8 bit)
~y + 1 = 0001 0101
~y = 0001 0100
y = 1110 1011

6. In a 16-bit system find the two's complement representation of the decimal value -21.
21 = 0000 0000 0001 0101(binary in 16 bit)
~y + 1 = 0000 0000 0001 0101
~y = 0000 0000 0001 0100
y =1111 1111 1110 1011