Andrew Nguyen

02/08/2021

CSE 140 – Computer Architecure

Homework 1

Exercise:

5.

32 bits

6.

3 types: J, I, R

J = j, jal

I = addi, andi

R = add, and

7.

a. I type, has 4 fields: opcode, rs, rt, immediate

b. value is 8 in hex, rs is 0 & 0 in hex, rt is 16 & in hex it is 0x10, 0x19

c. addi $s0, $zero, 25

Binary: 0010 0000 0001 0000 0000 0000 0001 1001

Hex: 0x20100019

8.

a. Machine cod in Hex: 0x0230402a

Binary: 0000 0010 0011 0000 0100 0000 0010 1010

b. R type

Can tell because end of hexcode being 2a and opcode is 000000

There are 6 fields: opcode, rs, rt, rd, shamt, funct

c. Binary: 000000 10001 10000 01000 00000 101010

opcode = 0x0, 0

rs = 0x11, 17 = $s1

rt = 0x10, 16 = $s0

rd = 0x8, 8 = $t0

shamt = 0x0, 0

funct = 0x2A, 42

d. R[rd] = (R[rs] < R[rt]) ? 1 : 0,

With the given opcode funct you it is shown the name of the code instruction to be referenced by the MISP sheet.

Mapping is: $t0 = ($s1 < $s0) ? 1 : 0

e. Final MIPS instruction: slt $t0, $s1, $s0

It is the same

9.

a. I type

b. 0x15000001 = 0001 0101 0000 0000 0000 0000 0000 0001 000101 01000 00000

0000000000000001

opcode = 0x5

rs = 0x8

rt = 0x0

c. The name of the target label = LESS

address of label in hex is: 0x0040001c

d. No. However it is needed for the operation if (R[rs] != R[rt]) PC = PC+4+BranchAddr.

The value of the branch address needs to be added to the PC.

e. The value of the immediate is 1 which is found by comparing the lines in between the given line and the target label line.

f. Machine Code is: 0x15000001

Binary: 0001 0101 0000 0000 0000 0000 0000 0001

Hex: 0x15000001

It is the same

10.

a. J type, has 2 fields: opcode and address

b. opcode instruction in hex: 0x2

c. The instruction jumps to GREQ and the address is: 0x00400030

d. You can use 26 bits in the address field.

We can squeeze the address by removing 4 bits from the start and 2 bits from the end because the first 4 bits are too big and the last 2 bits will always have a value of 00 so it can be removed

Binary: 0000 0000 0100 0000 0000 0000 0011 0000

e. Binary: 0000 1000 0001 0000 0000 0000 0000 1100

Hex: 0x0810000c

Yes it is the same

Assignment

0x0040000c

move $s1, $v0

R type

opcode: 000000, 0x0, 0

rs: 00000, 0x0, 0

rt: 00010, 0x2, 2

rd: 10001, 0x17, 11

shamt: 00000, 0x0, 0

funct: 100001, 0x21, 33

Binary: 000000 00000 00010 10001 00000 100001

Machine Code: 0x00028821

0x00400014

beq $t0, $zero, LEEQ

type: I

opcode: 000100, 0x4, 4

rs: 01000, 0x8, 8

rt: 00000, 0x0, 0

immediate: 0000000000000110, 0x0006, 6

Binary: 0001 0001 0000 0000 0000 0000 0000 0110

Machine Code: 0x11000006

0x0040002c

j END

type: J

opcode: 000010, 0x2, 2

address = 0x00400044 = 1000 0000 0000 0000 1000 100 to 0000 1000 0001 0000 0000 0000 0001 0001

Binary: 0000 1000 0001 0000 0000 0000 0001 0001

Machine Code: 0x08100011

0x00400034

la $a0, str2

type: I

opcode: 001111, 0xF, 15

rs: 00000, 0x0, 0

rt: 00001, 0x1, 1

immediate: 0001000000000001, 0x1001, 4097

Binary: 0011 1100 0000 0001 0001 0000 0000 0001

Machine Code: 0x3c011001