Activity 1

4) There are 32 bits per word within a machine instruction

5) There are three (3) different machine formats. The 3 different formats are R format, J format and I format.

* For R format, using *add $t0, $zero, $a0*  would be considered R format. Another example would be *sub $s0, $a0, $t0.* **R format refers to when all instructions values being used are located within a register**
* For J format, using *j END* would be considered J format. Another example would be *jal LOOP.* **J format refers to whenever a jump function is being used**
* For I format, using *addi $s0, $t0, 10*  would be considered I format. Another example would be *lw $t0, 3.* **I format refers to whenever a function holds a immediate constant present**

6)

1. The type is I format. This instruction has four (4) fields in machine language. The field names are | Opcode | rs | rs | Immediate |
2. The value of opcode for this instruction is 0x08. Register **rs** is $s0, the value of this register is 0x0000. Register **rt**  is $zero, the value of this register is 0x0000. The value of **immediate** in hex is 0x008f

|  |  |  |  |
| --- | --- | --- | --- |
| opcode | rs | rt | immediate |
| 2010 | 0000 | 000 | 0019 |

7)

a) 0x0230402a

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| opcode | rs | rt | rd | shamt | funct |
| 000000 | 10101 | 10100 | 01000 | 00000 | 111110 |

b)The type is R format. It can be see that its r format because all of there being 6 bits all filled with 000000 as the opcode. There are 6 fields in this instruction type. The fields names are **opcode, rs, rt, rd, shamt and funct**.

c)The values are 0x15, 0x14, 0x8, 0x3E

d) This operation instruction type is set less than (SLT). This can be seen by the funct value being 42.

e)

8)

A) The format is I format.

B)

|  |  |  |  |
| --- | --- | --- | --- |
| opcode | rs | rd | Immediate |
| 0x5 | 0x8 | 0x0 | 0x1 |

c)The target LABEL is LESS. The value of it is 0x01

d) We do put this as the value of immediate because, in this case the offset is only 1 so it moves the PC only 1 word forward, but in other cases where the LABEL address is somewhere completely different it has to be taken into consideration.

E) The value of the immediate field is 1. This value is any constant number that is being added to the machine code to perform the call

f)

9)

A) This is a J format instruction

B) The opcode is 2

C) The label it jumps to is GREQ and the address for this label is **4194352**

D) We can use 26 bits. It can be squeezed because for J formats, majority of the available memory for the machine code is used for the address of the label whereas the opcode still only receives 6 bits to identify it.

E) **0x00400030 and**

**Assignment**

**Line #7**

|  |  |  |  |
| --- | --- | --- | --- |
| opcode | rs | rt | immediate |
| 001000 | 10000 | 00000 | 1111111111110001 |

Line #14

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| opcode | rs | rt | Rd | Shamt | funct |
| 000000 | 01000 | 00101 | 00000 | 00000 | 101010 |

Line #17

|  |  |  |  |
| --- | --- | --- | --- |
| Opcode | Rs | Rt | Imm |
| 000100 | 01000 | 00000 | 100110 |

Line #20

|  |  |
| --- | --- |
| Opcode | Address |
| 000010 | 0000 0000 0100 0000 0000 0001 1100 |