**Tutorial 7:** WRES1201 – Computer System Architecture

1. What are the typical elements of a machine instruction?

**opcode; source and destination operand references; next instruction reference**

1. What type of locations can hold source and destination operands?

**Registers and memory**

1. If the instruction contains four addresses, what might be the purpose of each address?

Two operands, one result, and the address of the next instruction

1. List and briefly explain five important instruction set design issues?

**Operation:** How many and which operations to provide, and how complex operations should be.

**Data types:** The various types of data upon which operations are performed.

**Instruction format:** Instruction length (in bits), number of addresses, size of various fields, and so on.

**Registers:** Number of CPU registers that can be referenced by instructions, and their use.

**Addressing:** The mode or modes by which the address of an operand is specified.

1. What is the difference between an arithmetic shift and logical shift?

With a **logical shift***,* the bits of a word are shifted left or right. On one end, the bit shifted out is lost. On the other end, a 0 is shifted in.

The **arithmetic shift** operation treats the data as a signed integer and does not shift the sign bit. On a right arithmetic shift, the sign bit is replicated into the bit position to its right. On a left arithmetic shift, a logical left shift is performed on all bits but the sign bit, which is retained.

1. Why are transfer of control instruction needed?

**executed repeatedly**; **decision making**; **breaking the task**

1. List all the instruction to perform X = (A + B × C)/(D – E × F) for each following machine:
   1. One-Address Machine
   2. Two-Addresses Machine
   3. Three-Addresses Machine

Instructions that can be use are:

|  |  |  |
| --- | --- | --- |
| One-Address | Two-Address | Three-Address |
| LOAD M  STORE M  ADD M  SUB M  MUL M  DIV M | MOVE (X ← Y)  ADD (X ← X + Y) SUB (X ← X - Y) MUL (X ← X × Y) DIV (X ← X / Y) | MOVE (X ← Y)  ADD (X ← X + Z) SUB (X ← X - Z) MUL (X ← X × Z) DIV (X ← X / Z) |

Assume that T1, T2, T3 and etc are temporary storage, and for 2 and 3 addresses machine, we have register R1, R2, R3, …… as temporary storage.

|  |  |  |
| --- | --- | --- |
| LOAD E  MUL F  STORE T  LOAD D  SUB T  STORE T  LOAD B  MUL C  ADD A  DIV T  STORE X | MOV R0, E  MUL R0, F  MOV R1, D  SUB R1, R0  MOV R0, B  MUL R0, C  ADD R0, A  DIV R0, R1  MOV X, R0 | MUL R0, E, F  SUB R0, D, R0  MUL R1, B, C  ADD R1, A, R1  DIV X, R0, R1 |

1. Given the original bit = 10101110. Find the bit after the following shifting.
   1. Logical right shift (4 bits)
   2. Logical left shift (4 bits)
   3. Arithmetic right shift (4 bits)
   4. Arithmetic left shift (4 bits)
   5. Right rotate (4 bits)
   6. Left rotate (4 bits)
2. 00001010
3. 11100000
4. 11111010
5. 11100000
6. 11101010
7. 11101010