Instruction Set

Machine Instruction Characteristics

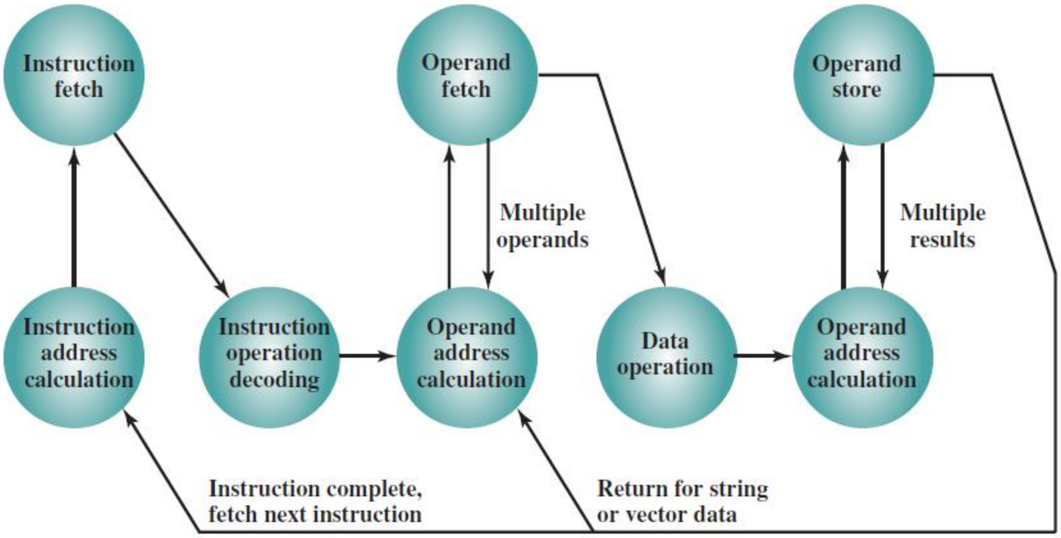
Operation of the processor is determined by the instruction it executes, referred to as Machine Instruction or Computer Instructions. Collection of Instruction that the processor can execute is referred to as processor's instruction set.

Element of Machine Instruction:

Each Instruction must contain information required by the processor for execution.

These elements are:
Operation Code: Specifies the operation to be performed.
Source Operand Reference: One or more source operand reference i.e. operands that are inputs for the operation.
Result Operand Reference: The result produce by the operation, stored at some specific location (s).
Next Instruction Reference: From where to fetch the next instruction once the current instruction is completed.

Source and Result operand reference can be in one of the three areas:
- Main or Virtual Memory
- CPU Registers
- I/O Devices
- Immediate


Instruction fetch
Operand fetch
Operand store
Multiple operands
Multiple results
Instruction address calculation
Instruction operation decoding
Operand address calculation
Data operation
Operand address calculation
Instruction complete, fetch next instruction
Return for string or vector data

Instruction Cycle State Diagram

From Designer's point of view:

Machine Instruction set provides the functional requirement for the processor.
Implementing the processor is implementing the machine instruction set

From Program's point of view:

Become aware of registers and memory structure, types of data directly supported by machine, functioning of ALU.

In most cases, the next instruction to be fetched immediately follows the current instruction, if a program counter is used, there is no need of explicit reference to next instruction.

Instruction Representation:

Instruction is represented by a sequence of bits.
Instruction is divided into fields, corresponding to elements of instruction.

OPCODE	Source Operand References 1, 2	Result Operand References 3	Next Instruction References 4

- Most instruction sets uses more than one instruction format
- Processor extract data from various instruction fields once it is in Instruction Register.
- In common practice, instructions are represented by symbolic representation called mnemonics.
- A machine language instruction express operations in a basic form involving movement of data to or from registers.

Instruction Types:

Machine Instruction set must be efficient to express any of the instructions from a high level language.

Data Processing: Arithmetic and Logic instruction.
Data Storage: Movement of data to and from register or memory location.
Data Movement: I/O instructions
Control: Test and Branch Instruction.

Number of Addresses:

- The Processor Architecture is traditionally described by describing the number of addresses contained in an instruction. This kind of description is not very common these days.
- Present day computer don't need all the FOUR addresses, explicitly.
- ONE, TWO or THREE address instruction may be used, the address of the next instruction is provided implicitly by the program counter.
- Most CPU designs involve a variety of Instruction Format.

Y= (A-B) / [C+(DxE)]

3 Address Format

```
SUB     Y, A, B         Y <- A – B
MPY     T, D, E         T <- D x E
ADD     T, T, C         T <- T + C
DIV     Y, Y, T         Y <- Y / T
```

2 Address Format

```
MOV     Y, A            Y <-  A
SUB     Y,  B           Y  <- B
MOV     T, D            T  <- D
MPY     T, E            T <- T x E
ADD     T, C            T <- T + C
DIV     Y, T            Y <- Y  / T
```

1 Address Format

```
LOAD    D               AC <- D
MPY     E               AC <- AC  x E
ADD     C               AC  <- AC + C
STORE   Y               Y <- AC
LOAD     A              AC <- A
SUB     B               AC <- AC –B
DIV     Y               AC <- AC / Y
STORE   Y               Y <- AC
```

In single address format / machine, many addresses are implied, It can be seen from the example.
One address is implied the other address is explicit.
The deficiency of 3rd Address is made up by adding one more instruction. Thus the overall program length has gone up.

Less Addresses Means:

- More Basic Instructions and less complex design of the CPU
- Instruction word is shorter
- More Instructions means longer program and longer execution time
- And the program design is also complex

Most computers employ a mixture of instruction formats in terms of number of explicit addresses.
Number of addresses per instruction is a basic design decision. (Zero Address Instructions are used with Stack)

Instruction Set Design:

The most interesting and most analyzed aspect of computer design is instruction set design. The design of an instruction set is very complex, since it affects so many aspects of the computer system:

The instruction defines many of the functions performed by the CPU and this has a significant effect on the implementation of the CPU. The instruction set is the programmer's means of controlling the CPU, Thus programmer's requirements must be considered in designing the instruction set.

The most fundamental design issues that still remains in dispute:

Operation Repertoire: How many and which operations to provide and how complex operation 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 processor registers, that can be referred by instruction and their use.
Addressing: The mode or modes by which the address of an operand is specified.

Finally, Instruction Set design is a complex subject. The instruction formats is a part of it and is again complex as it involves:

- Specification of total number of bits (Instruction Length).
- Number of Addresses in the instruction.
- Specification of the size of various field in the instruction word.

Types of Operands:

Machine instruction operates on data. The most important general categories of data are:
Addresses: (Addresses are also treated as data) unsigned integer)
Numbers: (Limited in terms of magnitude and precision)
Characters
Logical Data

Numbers:
Binary Integers
Binary Floating Point
Decimal (Binary Coded Decimal)

Characters:
ASCII or called IRA International Reference Alphabet.

Logical Data: When each bit of a word is treated separately.

Types of Operation:

The number of opcode varies from machine to machine. General type of operations found on all machines are:

Data Transfer: Move, Store, Load, Exchange, Clear, Set, Push, POP
Arithmetic: Add, Subtract, Multiply, Divide, Absolute, Negate, Increment, Decrement
Logical: AND, OR, NOT, Exclusive OR, Test, Compare, Shift, Rotate
Conversion: Translate, Convert
I/O : Input, Output, Start I/O, Test I/O
System Control:
Transfer of Control: Jump, JMP Condition, JMP Subroutine, Return, Execute Instruction, Skip, Skip Condition, Halt, Wait, No Operation

Data Transfer:
A Data Transfer instruction must specify:
i) Location of Source and Destination operands must be specified.
ii) Length of Data to be transferred.
iii) Addressing mode for each operand.

Data Transfer Operation occurs in following steps:
i) Calculate memory address, based on addressing mode.
ii) Determine whether the addresses item in in cache.
iii) If not issue a command to the memory module.

*****

Next Topic:

Addressing Modes to be discussed in Class