Instruction Set Architecture

Er. Anshu Singh

Assistant Professor

Computer Science & Engineering Department

University of Lucknow

Lucknow

Branch: BCA

Semester: 3

Subject: Computer Architecture

Disclaimer: The e-content is exclusively meant for academic purposes and for enhancing teaching and learning. Any other use for economic/commercial purpose is strictly prohibited. The users of the content shall not distribute, disseminate or share it with anyone else and its use is restricted to advancement of individual knowledge. The information provided in this e-content is developed from authentic references, to the best of my knowledge.

Instruction Set Architecture(ISA)

It is a collection of machine language instruction that a particular processor understand and execute.in other words set of assembly language mnemonics represents machine code of a particular computer.

Note: Instruction in a machine is dependent computer that is different processor have different instruction set.

Tow major parts of instruction

Opcode: An operation code filed termed as opcode that specify what operation to be performed. E.g. LOAD, STORE, ADD etc.

Operands: An address field of instruction tell on which data processing is to be performed. The operand can reside in memory or processor register or can be incorporated within the operand field of the instruction as an immediate constant.

Instruction set based classification of processor

Instruction set architecture is a set of processor design technique used to implement the instruction workflow on hardware

ISA basically tells how your processor is going to process your program instruction.

Classified into two ways

RISC(Reduced instruction set computer): RISC is a computer which only use simple instruction that can be divided into multiple instruction which perform low level operation with single clock cycle.

CISC(complex instruction set computer): CISC is a computer where single instruction can execute several low level operation(such load from memory , store into the memory and arithmetic operation) or capable of multistep operation or addressing modes within single instruction.

RISC & CISC architecture

Consider an example of multiplying two number:

A=A*B

CISC approach

CISC process would come prepared with a specific instruction (i.e. MULT). When this instruction executes.

- Load the two values into separate register.
- Multiplies the operands in the execution unit.
- Store the product in the appropriate register

Here you can see the entire task of multiplying two numbers can be completed with one instruction

MULT A, B---------- this is an assembly statement.

MULT is a complex instruction. It operates directly on the computer memory and does not require any programmer to explicitly call any loading or storing functions.

Advantage of CISC

- Complier has to do very little work to translate a high-level language statement into assembly language statement
- Length of the code is relatively short
- Very less RAM is required to store instructions
- The emphasis is put on building complex instruction directly into hardware

RISC approach

RISC use simple instruction that can be executed within one clock cycle. Thus here MULT command described in CISC would be divided into three separate commands.

1. LOAD--- moves data from memory bank to register
2. PROD – which calculate the product of two operands and locate within registers.
3. STORE --- which moves data form register to the memory.

Perform Multiplication of two number in RISC approach, programmer would need to code four lines of assembly

```
LOAD R1,A

LOAD R2,B

PROD  R1, R2

STORE A, R1
```

It might look less efficient way of completing the operation because there are more lines code more RAM is needed to store the assembly level instruction.

The complier must also have to do more work to convert high level language statements into assembly language code statements

Advantage of RISC

Each instruction requires only one clock cycle to execute, the entire program will be executed in approximately the same amount of time as the multi-cycle MULT command

RISC require less transistor of hardware space than complex instruction, RISC have more general purpose register because all the instruction execute in a uniform amount of time(i.e one clock cycle)

Difference between CISC and RISC architecture

CISC	RISC
Emphasis on hardware	Emphasis on software
Include multi- clock cycle	Single clock cycle
Complex instruction	Reduced instruction
Memory to memory: load and store incorporated in instruction	Register to register: load and store are independent instruction
Small code after conversion of program written in high level into assembly code	Large code after conversion of program written in high level into assembly code
Example: PDP-II, VAX , Motorola 6800, intel's x86 cpu	Example: DEC alpha, ARC, ARM, SPARC, IBM power instruction set

References:

1. John P Hayes "Computer Architecture and organization" McGraw Hill
2. Dezso Sima,Terence Fountain and Peter Kacsuk " Advanced Computer Architecture" Pearson Education
3. Kai Hwang " Advanced Computer Architecture" TMH
4. Linda Null, Julia Lobur- The Essentials of Computer Organization and Architecture, 2014, 4th Edition.