Processor Design and Implementation

Instruction set architecture

Microprocessor’s ISA includes the information needed to interact with the microprocessor, but not the details of the how the microprocessor itself is designed and implemented. It is essentially a programmer’s view of the microprocessor. It provides the details a programmer would need to know to write a program for the microprocessor, or the details a compiler would need in order to compile a program written in high-level programming language.

ISA includes

1. Instruction set (Assembly language instruction)
2. Programmer accessible registers and their use
3. Register sizes
4. Information necessary to interact with the memory
5. How the microprogramme react to interrupts in programmer point of view

ISA Design consideration

1. What should the ISA be able to do?
2. Tradeoffs in performance and size and cost
3. Instruction set have all of the instructions a program need to perform its required task
4. Instruction are orthogonal (Good instruction set minimizes the overlap between instructions)

Redundant instructions add to the cost provide no benefits to the user

1. Having few registers causes a program to make more references to memory thus reduces the performance
2. Backward compatibility
3. Floating-point data processing
4. Are interrupts needed
5. Are conditional instruction needed?

Example

