

ISA - Instruction Set Architecture (III)

Types of Operations

- Depending on the hardware, the operations a computer supports are decided.
- Based on the number of supported operations, the number of bits needed for opcode (operation code) for an ISA is also decided. Depending on the operation and the architecture, the number of operands, where the operands are, and the way to access these operands can be decided. These together decide the instruction format of an ISA.
- Some architectures have multiple instruction formats, while others have a uniform format.

- Operations can be roughly grouped into four categories:
 - Data transfer: transfer data from one location to another
 - most fundamental type of machine instruction
 - e.g., LOAD, STORE, PUSH, POP, etc.
 - *Things* data transfer instruction *must specify*
 - the *location of the source and destination* operands
 - each location could be memory or a register (integer or floating-point registers)
 - the *length of data* to be transferred (e.g., a word or a halfword)
 - the *mode of addressing* for each operand (relative/absolute)
 - If the operand is an absolute value, it refers to the value at that particular location in main memory.
 - If the operand adopts relative addressing, the operand could be a value in 2's complement, which refers to a memory location ahead of or behind the current PC value.
 - If the indicated absolute/relative address could be found in cache, load the value from cache. Otherwise, look into the main memory.
 - Arithmetic/logical operations: perform functions in ALU
 - Some machines have a set of instructions to support type conversion to facilitate arithmetic operations. So, the machines can support more types of data in addition to the ones that the arithmetic operations support without losing data precision.
 - Transfer of control: update program counter to execute sequentially or branch
 - I/O: transfer information between registers, memory, and input/output devices