CS147 - Lecture 12

Kaushik Patra
(kaushik.patra@sjsu.edu)

1

- Instruction Set Architecture (ISA)
- CISC & RISC

Reference Books:

1) *Chapter 12,13 of 'Computer Organization & Architecture' by Stallings.*
2) *Chapter 10 of 'Logic and Computer Design Fundamentals' by Mano & Kime.*

© 2014 - All rights are reserved by Kaushik Patra

Instruction Set Architecture ...

2

© 2014 - All rights are reserved by Kaushik Patra

Instruction Set – An Architecture !!!

- Processor architecture depends upon
  - Set of supported instructions
  - Data storage (register file and memory) characteristics.
- Designing instructions involves many common elements which dictates the final architecture of instruction set, which includes
  - List of instructions.
  - Data storage and access strategies.

3

- From the experience of putting together a processor for CS147DV instruction set we can say that architecture of a processor is very much dependent on the set of instructions and supported features. The other important aspect is the data (and instruction) storage strategies.
- Now the question is that how are these instructions are designed? Determining set of instruction is highly controversial area. There is no right and wrong answer. It depends on many factors including feature of the end product, compiler capabilities, efficiency of memory technology, target performance required from processor, etc.
- The outcome of designing an instruction set is the list of instruction, including its size, format, etc. and the data storage and access strategies. Like any other architecture subject, design (or architecture) involves some basic architectural components for designing instruction. Thus we usually refer this specific area as Instruction Set Architecture (ISA).

© 2014 - All rights are reserved by Kaushik Patra

Instruction Characteristics

- Elements of machine instruction
  - Operation code
  - Source operand references
  - Result operand references
  - Next instruction reference
- Data source and storage
  - Memory
  - Processor registers
  - Immediate
  - IO devices

4

- Essentially any instruction achieves some functionality in the target application. It has to have an unique ID so that one instruction can be uniquely identified from another. This ID is called operation code or OpCode. Each instruction, in general, involves multiple operands and one or more results. An instruction must contain reference to the operands and results. There has to an mechanism to identify next instruction to be executed. These are the most basic elements of an instruction.

- The references to the operands and results can be in the memory, or registers in processors, or embedded inside the instruction itself (immediate type). In some case, it is not also very unusual to refer data from the IO devices (keyboard, disk, etc).

© 2014 - All rights are reserved by Kaushik Patra

Instruction Characteristics

- Instruction representation
  - Fixed or variable length
  - Bit field definitions identifying the instruction elements
- Instruction Types
  - Data processing
  - Data storage
  - Data movement
  - Program Control

5

- An instruction can be represented in various way. The format defines which group of bits represents which information related to the instruction. Usually OpCode is represented with a fixed group of bits where processor can pick the code. Depending on the OpCode, rest of information are mapped inside the instruction format to individual groups of bits. Some ISA allows variable length of such groups and as result it allows variable length instructions. On the other hand, some ISA assumes a fixed length instruction and each group length is fixed. This way, the processor circuit is simplified, but some bits may remain unused.

- An ISA should, in general, supports instructions for data processing (arithmetic or logic operation), data storage (memory access), data movement (some ISA supports data to be read or written with an IO device), and program control (branch on some test conditions – to implement decision power in software program).

© 2014 - All rights are reserved by Kaushik Patra

Instruction Design

- Most disputed area in computer architecture study. The debated areas includes the following:
  - Operation repertoire
  - Data Types
  - Instruction formats
  - Registers to hold intermediate data
  - **Data organization in memory**
  - **Addressing modes of an operand**

6

- There is no hard rules for designing an optimal ISA. One ISA needs to address many aspect of the features that a processor going to offer to its user. One such aspect is the operation repertoire or the list of operations. It can be a very simple and basic set of operation or can be as complex as a high level operation. The other aspect is the data type the instruction handles. It may be integer, float, address, character, etc. ISA also defines number of registers to hold intermediate data. Another important aspect of ISA is the data organization in the memory in case of byte addressable memory (which is the most common case). The aspect of addressing mode defines how the data (for operands and result) are to be accessed (read or write). The data organization and addressing mode will be discussed in details.

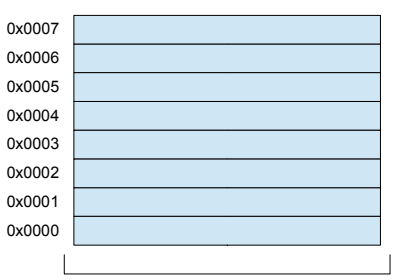
© 2014 - All rights are reserved by Kaushik Patra

Data Organization in Memory ...

7

© 2014 - All rights are reserved by Kaushik Patra

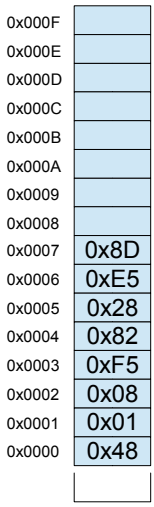
Byte Addressable Memory



For 32-bit access

M[0x0000] = 0x480108F5
M[0x0004] = 0x8228E58D

M[0x0001] = 0x0108F582



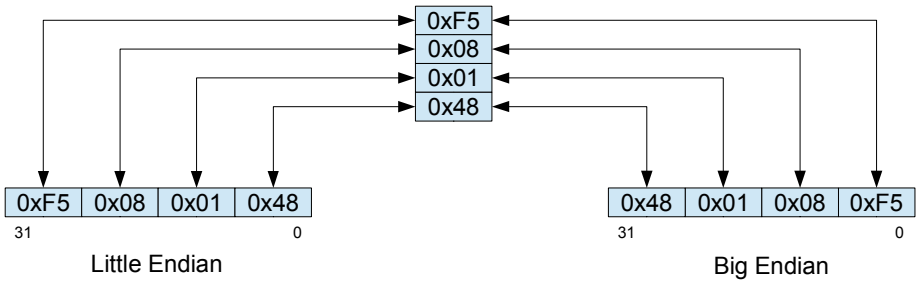
Next 32-bit data from 0x0000 will be 0x0004

8

- So far, we have considered 32-bit word accessible memory. This means each address points to a complete 32-bit word. However, this is not a very common case. We have picked up this model for our class project to make the circuit simple. In reality, the memory are byte addressable. This means each address points to a byte. Hence to access a complete 32-bit word consecutive 4 addresses are accessed to compose 32-bit information. Hence to read two successive words, we need to read addresses those are 4 address apart (address and adress+4, unlike the word addressable memory in which address+1 will give the next word).

© 2014 - All rights are reserved by Kaushik Patra

Endianness of Data in Memory



9

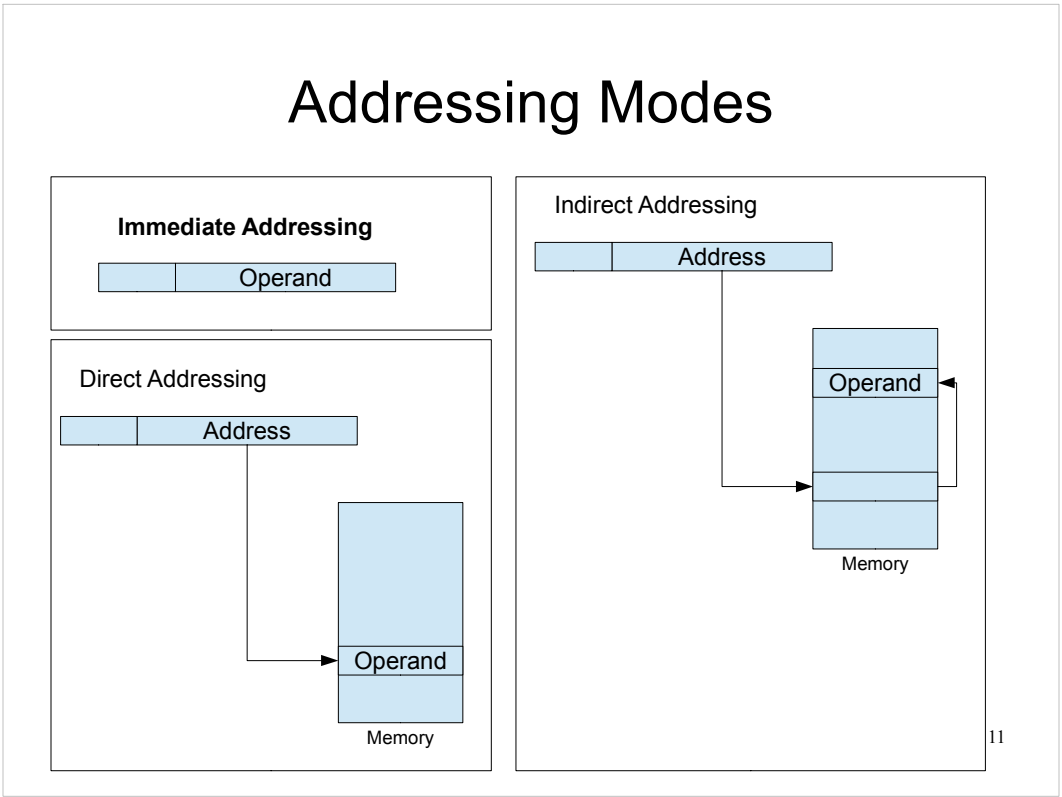
- Disagreement still exists on how to organize data in a byte addressable memory. There are two schools of thought – big endian and little endian. The big endian suggests that the lowest byte address will contain the most significant byte while the highest address will contain the least significant byte. The little endian concept is just the opposite.

- Each processor assume one of the endianness. ARM processor has a control bit which can be set by user to switch between big and little endian.

© 2014 - All rights are reserved by Kaushik Patra

Operand Addressing Modes ...

© 2014 - All rights are reserved by Kaushik Patra



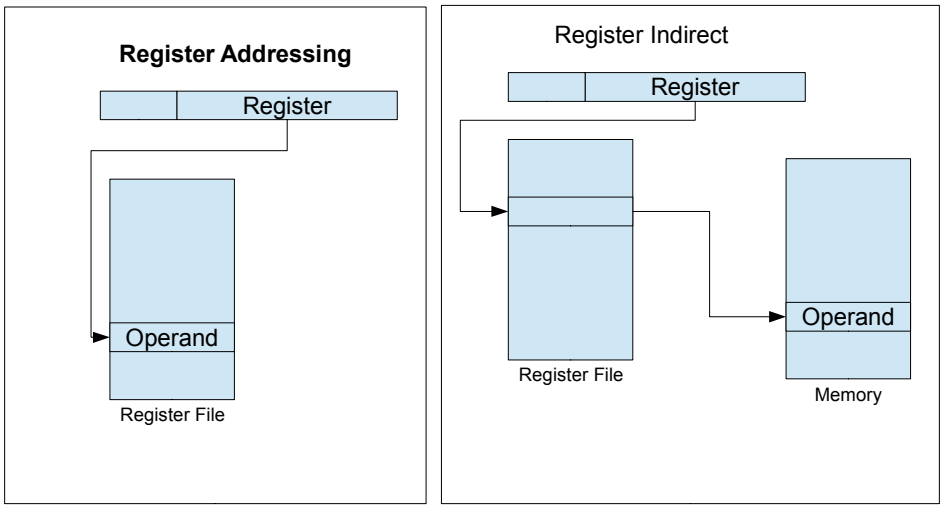
- The immediate addressing mode embeds the operands within in the instruction word. The data is directly taken out of the group of bits in the instruction word itself. The CS147DV ISA supports this.

- The direct addressing mode contains the address of the operand embedded in the instruction word. The operand is to be retrieved from the memory location before it can be used.

- The indirect addressing mode contains the address of the address of the operand embedded in the instruction word. Firs the operand address to be retrieved from the memory and then using the retrieved address, the operand is retrieved from the memory.

© 2014 - All rights are reserved by Kaushik Patra

Addressing Modes



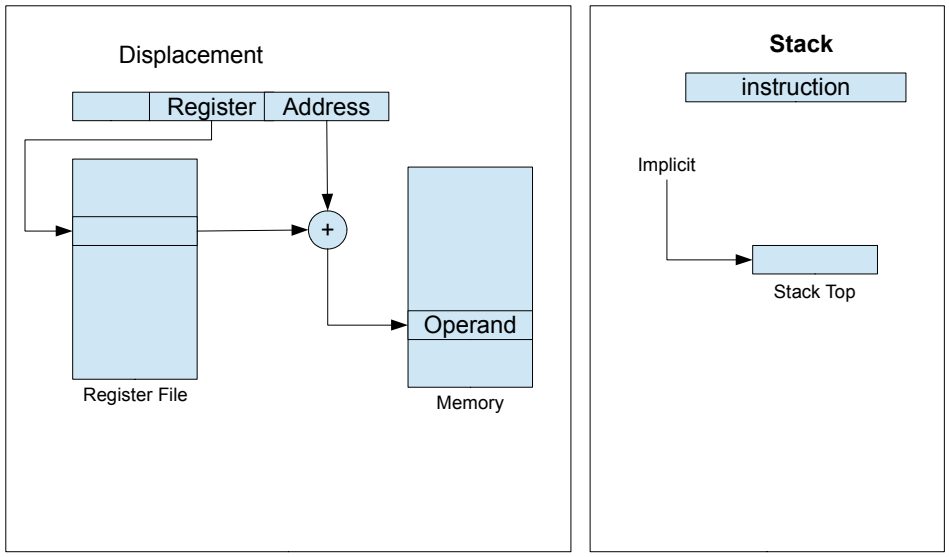
12

- The register addressing mode embeds the register ID (location) information in the instruction word. The register file is accessed at the specified ID to retrieve the operand. The CS147DV ISA supports this.

- The register indirect addressing mode embeds the register ID (location) in the information word. This register contains the address of the operand in the memory. The operand is retrieved from the memory using address as contained in the register.

© 2014 - All rights are reserved by Kaushik Patra

Addressing Modes



- The displacement addressing mode is a combination of register indirect addressing and direct addressing mode. In this case the operand address is computed by adding address in the register and the address embedded in the instruction word.

- The stack addressing mode assumes an implicit operand address at the stack pointer register (SP). CS147DV supports this addressing mode. The instruction word does not embed any information about the operand location.

© 2014 - All rights are reserved by Kaushik Patra

CISC & RISC ...

14

© 2014 - All rights are reserved by Kaushik Patra

Clash of Titans

- The continuing battle between two class of instruction set architecture with different philosophy
  - CISC (Complex Instruction Set Computer)
    - Reduce amount of memory usage and access.
    - Support complex assembly level language.
    - Give compatibility support.
    - Make compiler's job easier.
  - RISC (Reduced Instruction Set Computer)
    - Execute one instruction per clock.
    - Keep all the instruction same size.
    - Allow only load/store to access memory
    - Give complex functionality support through high level language.

15

- CISC or complex instruction set computer is relatively older school of thought. This philosophy was mostly evolved out of two facts which were true in early computing data. The first one is the lack of powerful compilers. The programmers used to program in assembly language, hence the primary objective was to make assembly programmers' life easier by adding huge set of complex instructions, so that size of the mnemonics is manageable within human skill. The other fact was constraint imposed by memory technology. In those days memory was not a cheap commodity as it is now. Therefore the natural choice to save memory by compacting instructions as much as possible (which leads to the variable length compact instruction set). X86 ISA is a CISC type instruction set which is supported in INTEL processor, AMD processor, ARM processor, and many others.

- RISC or reduced instruction set computer is fairly new ISA. With the advent of powerful compiler and cheaper memory, researches were conducted to design for a higher performance processor. It was observed that simpler and uniform instruction format leads to simpler processor circuit which will result in faster operation. For a RISC style design, it is assumed that one instruction will be completed within one clock cycle, unlike the CISC architecture. ORACLE's SPARC processors, MIPS, and many others supports RISC ISA.

© 2014 - All rights are reserved by Kaushik Patra

Characteristics of CISC

- Memory access is directly available to most type of instructions.
- Addressing modes are substantial in number.
- Variable length instruction format.
- Instruction includes elementary and complex operation.

16

- Many instruction access memory directly unlike RISC (where only lw and sw instruction access memory). A CISC ISA supports most of the addressing modes discussed earlier. Also to reduce memory footprint of program stored in memory, the instructions are made variable lengths so that no bit in the word is unused. Also for the same purpose of decreasing memory footprint, more complex instructions are supported (instructions which can be alternately written with multiple simple instructions).

© 2014 - All rights are reserved by Kaushik Patra

Characteristics of RISC

- Data manipulation instructions are strictly register-to-register. Only two instruction can access the memory.
- Addressing modes are limited in numbers.
- Equal length instruction format.
- Instruction includes only elementary operations.

17

- For a RISC style computers the data manipulations instructions are strictly register based or immediate. One load and store memory instructions are allowed to access memory. By principle, RISC allows only limited number of addressing mode (immediate and registers). It also assumes equal length of instruction, which means there are cases of unused bits in an instruction. A RISC instruction set only contains simple basic instruction. All the complex functionality are implemented by software (e.g. division using subtraction only).

© 2014 - All rights are reserved by Kaushik Patra

CISC vs RISC

- CISC to RISC is a paradigm shift in computer architecture.
  - Analogous to shift from assembly language to higher level language.
- RISC offers simplicity of circuit structure.
  - Results in better performance through pipelining.
- Memory is no longer a scares resource, hence RISC style does not suffer much from the side effect of consuming more memory for program code.

18

- CISC school of ISA is mostly inclined to assist assembly programmer where RISC is inclined to have faster processor through instruction simplification. RISC is a paradigm shift in architecture similar to paradigm shift in programming though introduction of high level language. Though RISC style ISA tends to give higher memory footprint for program storage, it is no longer an issue since memory technology now supports very large amount of storage space.

© 2014 - All rights are reserved by Kaushik Patra

CS147 - Lecture 12

Kaushik Patra
(kaushik.patra@sjsu.edu)

19

- Instruction Set Architecture (ISA)
- CISC & RISC

Reference Books:

1) *Chapter 12,13 of 'Computer Organization & Architecture' by Stallings.*
2) *Chapter 10 of 'Logic and Computer Design Fundamentals' by Mano & Kime.*

© 2014 - All rights are reserved by Kaushik Patra