University of Babylon, College of science for women
Dept. of Computer science

Computer Architecture

Second year

Dr. Salah Al-Obaidi

Lecture #1: **Introduction**

Spring 2024

Contents

1. Introduction

Computer architecture is concerned with all aspects of the design and organization of the central processing unit (CPU) and the integration of the CPU into the computer system itself. The architecture extends upward into computer software because a processor's architecture must cooperate with the operating system and system software. It is impossible to design an operating system well without a knowledge of the underlying architecture. Moreover, the computer designer has to have an intimate understanding of software to implement the optimum architecture.

Computer architecture is concerned with the structure and behavior of the computer as seen by the user. It includes the *information formats*, *the instruction set*, and *techniques for addressing memory*. Computer architecture is a key component of computer engineering and the practicing computer engineer should have a practical understanding of this topic.

The term architecture implies structure and therefore computer architecture tells us something about the way in which the elements of a computer relate to each other. Computer architecture is generally thought of as the programmer's view of a computer; that is, the idealized or abstract view of a computer. The implementation or organization of the computer is hidden from the programmer, although it would be wrong to divorce entirely architecture and implementation because each exerts a powerful influence on the other.

The computer architecture curriculum has to achieve multiple objectives. It must provide an overview of computer architecture and teach students the operation of a typical von Neumann machine. It must highlight the important issues facing today's designers and give students the tools they will need to carry out research. Ideally, it should reinforce topics that are common to other areas of computer science; for example, teaching register indirect addressing reinforces the concept of pointers in C.

1.1 Computer Organization and Computer Architecture

In describing computers, a distinction is often made between *computer architecture* and *computer organization*. **Computer architecture** refers to those attributes of a system that have a direct impact on the logical execution of a program (see Figure 1.1). Examples of these attributes are:

- the instruction set
- the number of bits used to represent various data types (e.g., numbers, characters)
- I/O mechanisms
- memory addressing techniques

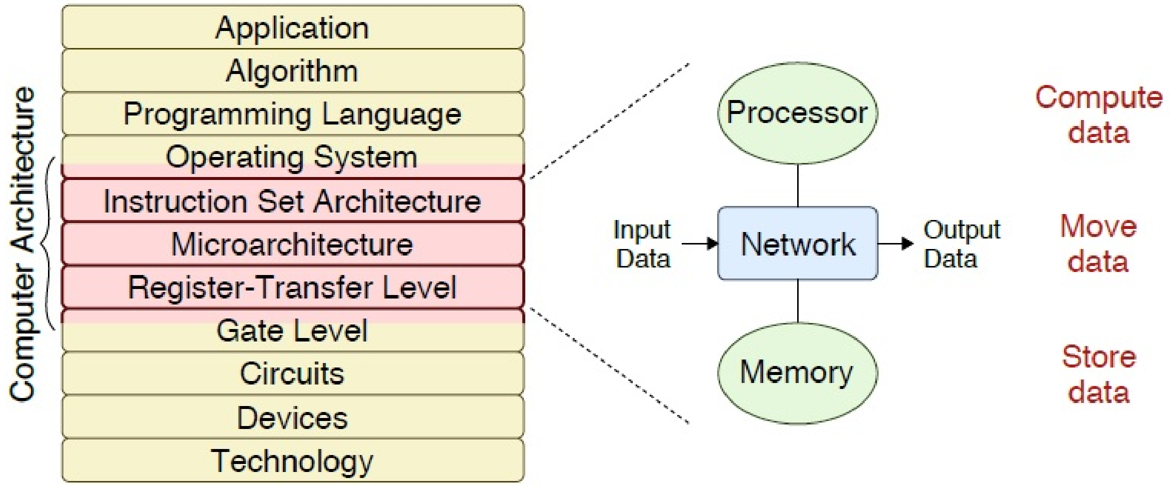

Figure 1.1: Computer Architecture for a programmer.

A term that is often used interchangeably with computer architecture is **instruction set architecture (ISA)**. The ISA defines *instruction formats*, *instruction opcodes*, *registers*, *instruction and data memory*; *the effect of executed instructions on the registers and memory*; and *an algorithm for controlling instruction execution.*

Computer Organization refers to the operational units and their interconnections that realize the architectural specifications. Examples are things that are transparent to the programmer:

- control signals
- interfaces between computer and peripherals
- the memory technology being used

So, for example, the fact that a multiply instruction is available is a computer architecture issue. How that multiply is implemented is a computer organization issue.

Architecture is those attributes visible to the programmer:

- Instruction set, number of bits used for data representation, I/O mechanisms, addressing techniques.
- e.g. Is there a multiply instruction?

Organization is how features are implemented

- Control signals, interfaces, memory technology.
- e.g. Is there a hardware multiply unit or is it done by repeated addition?

The organizational decision may be based on the anticipated frequency of use of the multiply instruction, the relative speed of the two approaches, and the cost and physical size of a special multiply unit.

1.2 Defining Computer Architecture

The task the computer designer faces is a complex one: Determine what attributes are important for a new computer, then design a computer to maximize performance while staying within cost, power, and availability constraints. This task has many aspects, including *instruction set design*, *functional organization*, *logic design*, and *implementation*. The implementation may encompass integrated circuit design, packaging, power, and cooling. Optimizing the design requires familiarity with a very wide range of technologies, from compilers and operating systems to logic design and packaging.

In the past, the term computer architecture often referred only to instruction set design. Other aspects of computer design were called implementation, often insinuating that implementation is uninteresting or less challenging.

However, the architect's or designer's job is much more than instruction set design, and the technical hurdles in the other aspects of the project are likely more challenging than those encountered in instruction set design.

1.3 Structure and Function

A computer is a complex system; contemporary computers contain millions of elementary electronic components. How, then, can one clearly describe them? The key is to recognize the hierarchical nature of most complex systems, including the computer. A hierarchical system is a set of interrelated subsystems, each of the latter, in turn, hierarchical in structure until we reach some lowest level of the elementary subsystem.

The hierarchical nature of complex systems is essential to both their design and their description. The designer need only deal with a particular level of the system at a time. At each level, the system consists of a set of components and their interrelationships. The behavior at each level depends only on a simplified, abstracted characterization of the system at the next lower level. At each level, the designer is concerned with structure and function:

- Structure: The way in which the components are interrelated.
- Function: The operation of each individual component as part of the structure.

Function

In general terms, there are only four basic functions that a computer can perform:

- **Data processing:** Data may take a wide variety of forms, and the range of processing requirements is broad. However, we shall see that there are only a few fundamental methods or types of data processing.
- **Data storage:** Even if the computer is processing data on the fly (i.e., data come in and get processed, and the results go out immediately), the computer must temporarily store at least those pieces of data that are being worked on at any given moment. Thus, there is at least a short-term data storage function. Equally important, the computer performs a long-term data storage function. Files of data are stored on the computer for subsequent retrieval and update.
- **Data movement:** The computer's operating environment consists of devices that serve as either sources or destinations of data. When data are received from or delivered to a device that is directly connected to the computer, the process is known as input– output (I/O), and the device is referred to as a peripheral. When data are moved over

longer distances, to or from a remote device, the process is known as data communications.

- **Control:** Within the computer, a control unit manages the computer's resources and orchestrates the performance of its functional parts in response to instructions.

Structure

We now look in a general way at the internal structure of a computer. We begin with a traditional computer with a single processor that employs a microprogrammed control unit, then examine a typical multicore structure.

SIMPLE SINGLE-PROCESSOR COMPUTER: Figure 1.2 provides a hierarchical view of the internal structure of a traditional single-processor computer. There are four main structural components:

- **Central processing unit (CPU):** Controls the operation of the computer and performs its data processing functions; often simply referred to as processor.
- **Main memory:** Stores data.
- **I/O:** Moves data between the computer and its external environment.
- **System interconnection:** Some mechanism that provides for communication among CPU, main memory, and I/O. A common example of system interconnection is by means of a system bus, consisting of a number of conducting wires to which all the other components attach.

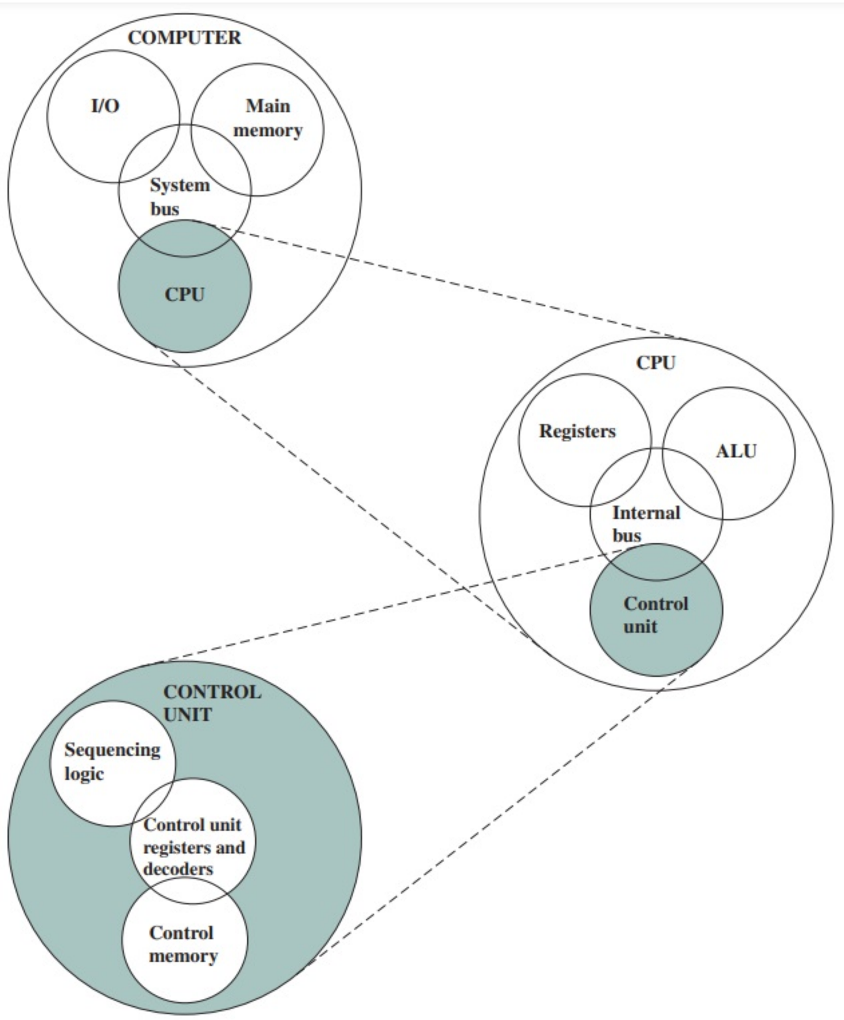

Figure 1.2: The Computer: Top-Level Structure

There may be one or more of each of the aforementioned components. Traditionally, there has been just a single processor. In recent years, there has been increasing use of multiple processors in a single computer.

However, for our purposes, the most interesting and in some ways the most complex component is the CPU. Its major structural components are as follows:

- **Control unit:** Controls the operation of the CPU and hence the computer.

- **Arithmetic and logic unit (ALU):** Performs the computer's data processing functions.
- **Registers:** Provides storage internal to the CPU.
- **CPU interconnection:** Some mechanism that provides for communication among the control unit, ALU, and registers.

MULTICORE COMPUTER STRUCTURE: contemporary computers generally have multiple processors. When these processors all reside on a single chip, the term **multicore computer** is used, and each processing unit (consisting of a control unit, ALU, registers, and perhaps cache) is called a **core**. To clarify the terminology, the following definitions will be used.

- **Central processing unit (CPU):** That portion of a computer that fetches and executes instructions. It consists of an ALU, a control unit, and registers. In a system with a single processing unit, it is often simply referred to as a processor.
- **Core:** An individual processing unit on a processor chip. A core may be equivalent in functionality to a CPU on a single-CPU system. Other specialized processing units, such as one optimized for vector and matrix operations, are also referred to as cores.

- **Processor:** A physical piece of silicon containing one or more cores. The processor is the computer component that interprets and executes instructions. If a processor contains multiple cores, it is referred to as a **multicore processor**.

Another prominent feature of contemporary computers is the use of multiple layers of memory, called cache memory, between the processor and main memory. We simply note that a cache memory is smaller and faster than main memory and is used to speed up memory access, by placing in the cache data from main memory, that is likely to be used in the near future. A greater performance improvement may be obtained by using multiple levels of cache, with level 1 (L1) closest to the core and additional levels (L2, L3, and so on) progressively farther from the core.

Figure 1.3 is a simplified view of the principal components of a typical multicore computer.

Next, we zoom in on the structure of a single core, which occupies a portion of the processor chip. In general terms, the functional elements of a core are:

- **Instruction logic:** This includes the tasks involved in fetching instructions, and decoding each instruction to determine the instruction operation and the memory locations of any operands.

- **Arithmetic and logic unit (ALU):** Performs the operation specified by an instruction.

- **Load/store logic:** Manages the transfer of data to and from main memory via cache

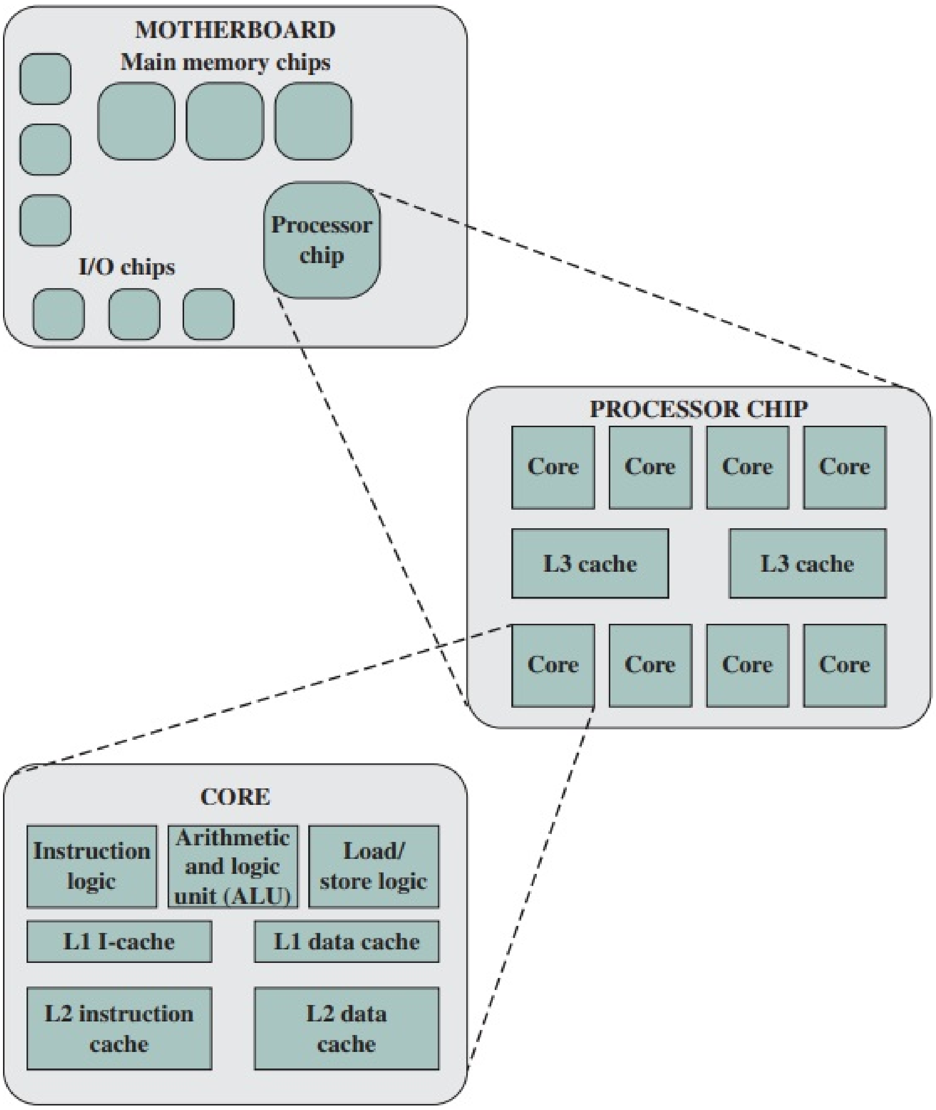

Figure 1.3: Simplified View of Major Elements of a Multicore Computer

The core also contains an L1 cache, split between an instruction cache (I-cache) that is used for the transfer of instructions to and from main memory, and an L1 data cache, for the transfer of operands and results. Typically, today's processor chips also include an L2 cache as part of the core. In many cases, this cache is also split between instruction and data caches, although a combined, single L2 cache is also used.

Figure 1.4 is a photograph of the processor chip for the IBM zEnterprise EC12 mainframe computer. This chip has 2.75 billion transistors. The superimposed labels indicate how the silicon real estate of the chip is allocated. We see that this chip has six cores, or processors. In addition, there are two large areas labeled L3 cache, which are shared by all six processors. The L3 control logic controls traffic between the L3 cache and the cores and between the L3 cache and the external environment. Additionally, there is storage control (SC) logic between the cores and the L3 cache. The memory controller (MC) function controls access to memory external to the chip. The GX I/O bus controls the interface to the channel adapters accessing the I/O.

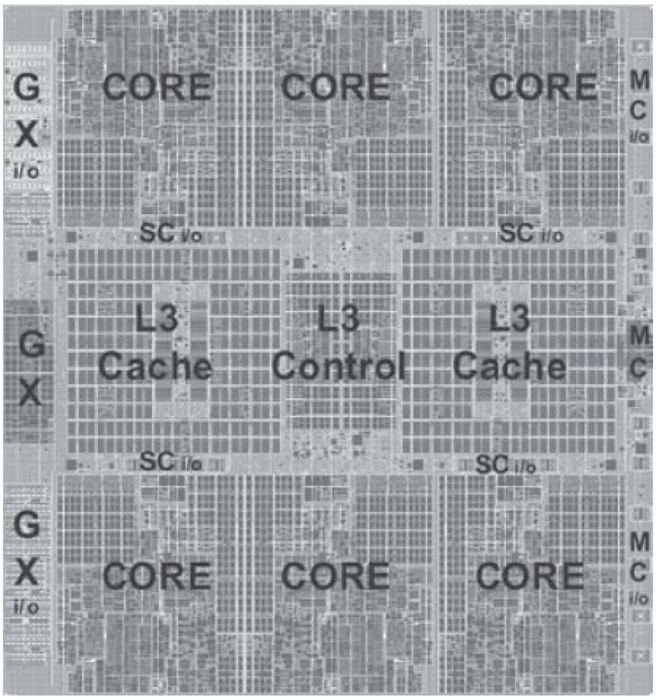

Figure 1.4: zEnterprise EC12 Processor Unit (PU) chip diagram