Chapter 1: Introduction
Chapter Contents

1.1 Overview
1.2 A Brief History
1.3 The Von Neumann Model
1.4 The System Bus Model
1.5 Levels of Machines
1.6 Upward Compatibility
1.7 The Levels
1.8 A Typical Computer System
1.9 Organization of the Book
1.10 Case Study: What Happened to Supercomputers
Some Definitions

• *Computer architecture* deals with the functional behavior of a computer system as viewed by a programmer (like the size of a data type – 32 bits to an integer).

• *Computer organization* deals with structural relationships that are not visible to the programmer (like clock frequency or the size of the physical memory).

• There is a concept of *levels* in computer architecture. The basic idea is that there are many levels at which a computer can be considered, from the highest level, where the user is running programs, to the lowest level, consisting of transistors and wires.
Pascal’s Calculating Machine

- Performs basic arithmetic operations (early to mid 1600’s). Does not have what may be considered the basic parts of a computer.
- It would not be until the 1800’s until Babbage put the concepts of mechanical control and mechanical calculation together into a machine that has the basic parts of a digital computer.

(Source: IBM Archives photograph.)
The von Neumann Model

- The von Neumann model consists of five major components: (1) input unit; (2) output unit; (3) arithmetic logic unit; (4) memory unit; (5) control unit.
The System Bus Model

- A refinement of the von Neumann model, the system bus model has a CPU (ALU and control), memory, and an input/output unit.

- Communication among components is handled by a shared pathway called the *system bus*, which is made up of the data bus, the address bus, and the control bus. There is also a power bus, and some architectures may also have a separate I/O bus.
Levels of Machines

• There are a number of levels in a computer (the exact number is open to debate), from the user level down to the transistor level.

• Progressing from the top level downward, the levels become less abstract as more of the internal structure of the computer becomes visible.

<table>
<thead>
<tr>
<th>High Level</th>
<th>User Level: Application Programs</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>High Level Languages</td>
</tr>
<tr>
<td></td>
<td>Assembly Language / Machine Code</td>
</tr>
<tr>
<td></td>
<td>Microprogrammed / Hardwired Control</td>
</tr>
<tr>
<td></td>
<td>Functional Units (Memory, ALU, etc.)</td>
</tr>
<tr>
<td>Low Level</td>
<td>Logic Gates</td>
</tr>
<tr>
<td></td>
<td>Transistors and Wires</td>
</tr>
</tbody>
</table>
A Typical Computer System

- Monitor
- Diskette drive
- CD-ROM drive
- Hard disk drive
- Sockets for plug-in expansion cards
- Sockets for internal memory
- Keyboard
- CPU (Microprocessor beneath heat sink)
The Motherboard

- The five von Neumann components are visible in this example motherboard, in the context of the system bus model.

Manchester University Mark I

- Supercomputers, which are produced in low volume and have a high price, have been largely displaced by, high-volume low-priced machines that offer a better price-to-performance ratio.

(Source: http://www.paralogos.com/DeadSuper)
Moore’s Law

- Computing power doubles every 18 months for the same price.
- Project planning needs to take this observation seriously: an architectural innovation that is being developed for a projected benefit that quadruples performance in three years may no longer be relevant: the architectures that exist by then may already offer quadrupled performance and may look entirely different from what the innovation needs to be effective.