Chapter 1

Overview

Typical functions that computer systems perform:

- sending and receiving messages, photos and videos,
- performing complicated calculations,
- making railway or airline reservations,
- creating and storing documents,
- accessing documents from remote systems,
- displaying graphics of amazing colour and clarity

… and the list could go on.

Computer *system*

- Functionality is achieved through proper interconnection and coordinated operation of a number of separate *sub-systems:*
  - processor,
  - memory
  - keyboard,
  - disk drive,
  - display, etc.

Brief history of computers

- First generation of computers using vacuum tube technology were developed in the middle of the twentieth century
- Conceptual breakthroughs achieved by John von Neumann and Alan Turing
- Second generation of computers used silicon transistors
- Third generation of computers were built using Integrated Circuit (IC) technology
- Fourth generation of computers (current generation) uses VLSI circuits (Very large scale integration)

Information

- Communication and storage of information requires the use of *symbols*, such as alphabets or numbers
- For quantifying the concept of information, we make use of the very simple *binary alphabet*, which has only the two symbols ‘0’ and ‘1’
- A sequence of symbols – e.g. letters of alphabet, decimal digits, binary digits – is known as a *string* of symbols.
- A string of bits of length *n* has information content of *n* bits.
- For a symbol other than binary, we have to find its *equivalent number of bits*.

- For the string of four letters ‘ABCD’, the equivalent number of bits / information content of the string can be found as follows:
  - Assume that the symbols are selected from upper case English letters, which are twenty six in number.
  - The total number of different such strings of length four is:
    26x26x26x26 = 456976,
    since each of the four letters in the string can be independently selected in 26 different ways.
- The string ‘ABCD’ is therefore one out of 456976 possible such strings.

- How many bits do we need to be able to construct 456976 distinct binary strings?

- This number is 19, since 2^{18} is less than 456976, but 2^{19} is greater than 456976.

- Modern computer systems handle millions of bits of information
  - 1 *kilo*bit is equal to 1024 bits
  - 1 byte is equal to 8 bits
  - 1 *kilo*byte is equal to 1024 bytes
  - 1 *mega*bit is equal to 1024 kilobits
  - 1 *mega*byte is equal to 1024 kilobytes
  - 1 *giga*bit is equal to 1024 megabits
  - 1 *giga*byte is equal to 1024 megabytes

- A computer records a bit as ‘0’ or ‘1’, using a *two-state device* – the basic building block of a computer

Hardware and Software

- *Hardware* refers to the various physical sub-systems and components of a computer system

- A computer system must include at least two hardware elements in it

  (i) a memory element

  (ii) an element to execute instructions (processor)

- *Computer program* is a set of instructions which together perform a well-defined function

- Software refers to all the programs which may run on the computer system

- Software includes:
  - *Applications software*
  - *System software*

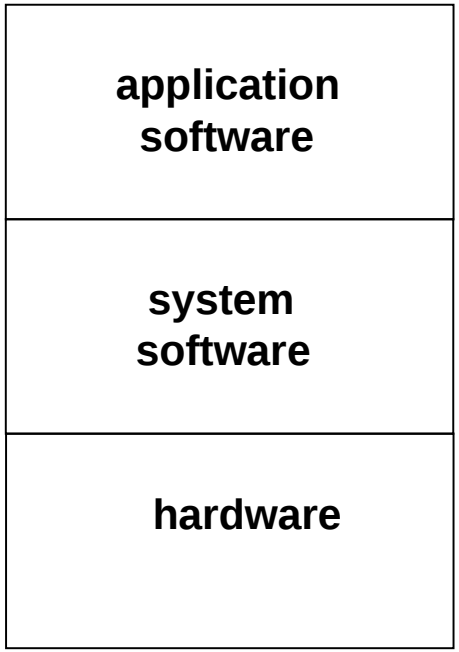
Algorithm

- A clearly specified set of steps to perform a function.

- An algorithm can be described by *algorithmic notation*, such as the following:

```
repeat the following ten times
{
   read a number from the input device
   multiply it by three
   display the number on output device
}
```

Relationship between applications software, system software, and hardware



Processing

- Instructions go through two basic phases:
  - Fetch
  - Execute

- Algorithmic notation for a typical processor

```
repeat forever
{
  fetch next processing instruction from
      main memory
  execute – i.e. carry out – the instruction
      fetched
}
```

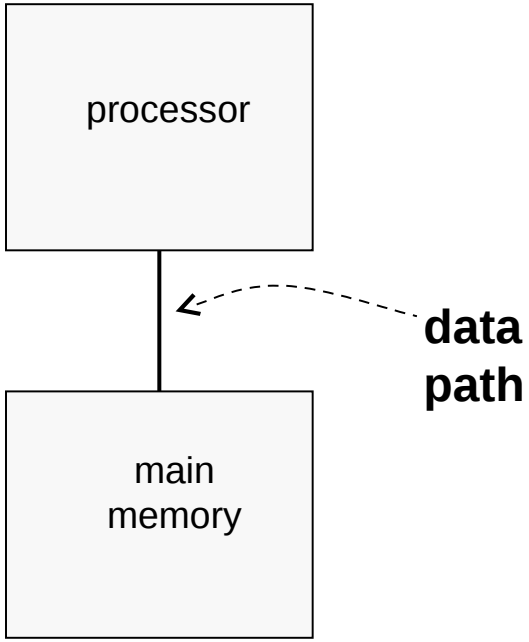
Memory

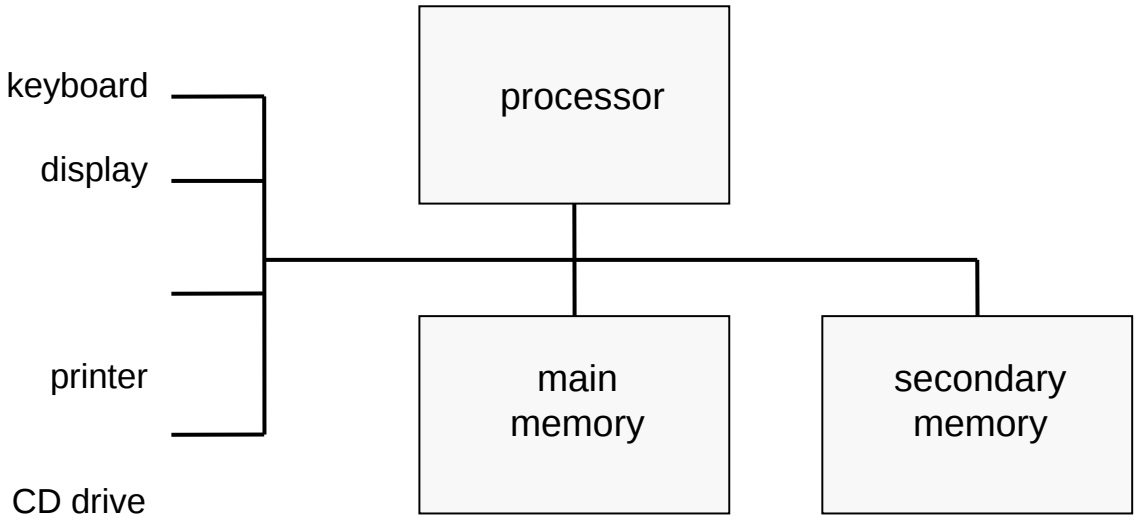
- Main memory stores information and instructions
  i.e. main memory stores *program(s)* as well as *data*

- Storage media may be
  - Volatile: faster and more expensive per bit (main memory)
  - Non-volatile: slower and relatively inexpensive (secondary memory)

- A typical computer system therefore needs main memory as well as secondary storage

- The figure below shows a simple computer with memory and processor
- *Data path* between main memory and processor provides for transfer of instructions and data between them





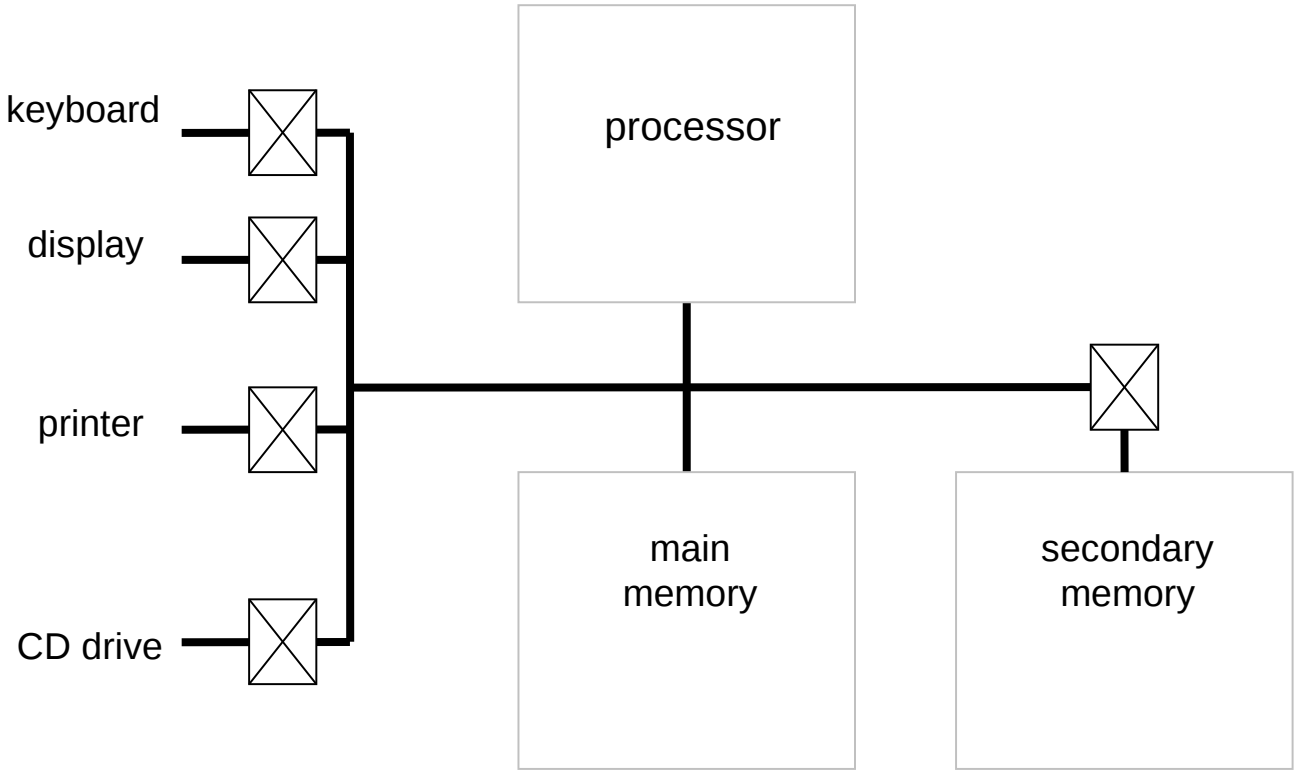
Computer system with input and output functionality added

- A few important related concepts
  - Microcontroller
  - Logical / Physical organisation of computer system
  - Interface standards and compatibility

Models of computer systems

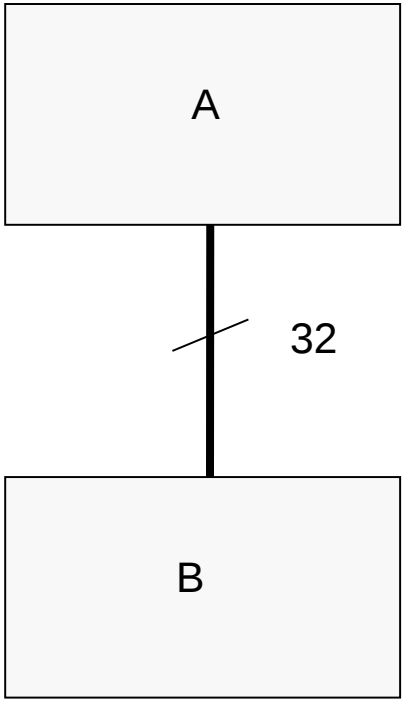
- *von Neumann model* (seen above)
  - Basic design of a computer system
  - Single processor and common main memory for programs and data

- Alternative models
  - Two different memory elements for instructions and data
  - Multiple processors and common main memory
  - Data flow architecture

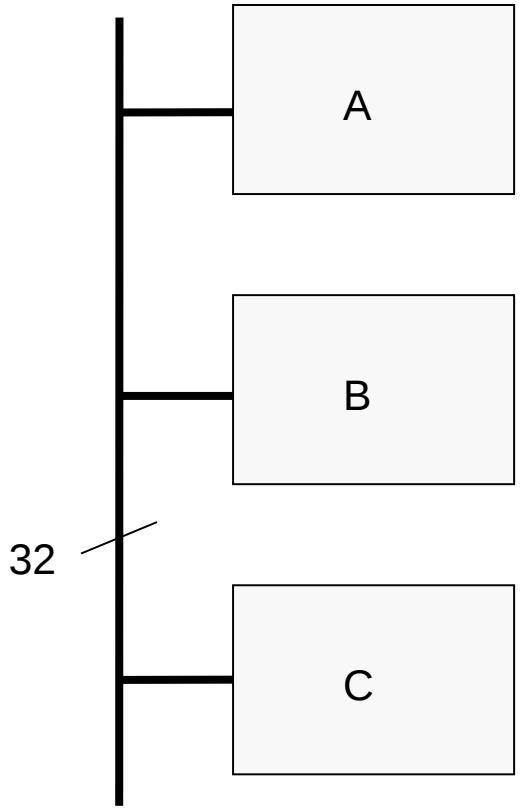


A Computer system with emphasis on the interconnections between sub-systems

(a) A **data path** of width 32 bits between sub-systems A and B



(b) A **bus** of width 32 bits with subsystems A, B and C connected to it



A few important concepts

- Parallel versus serial connection
- Timing and control signals
- Device controller
- Disk controller
- *Width* of a data-path
- Modularity

Differences between computer systems may be in terms of:

- rate at which information can be processed,
- amount of main memory and secondary memory available,
- variety of input and output devices available,
- quality of graphical display unit,
- power consumption,
- size and portability,
- expandability, i.e. range of ‘add-on’ options available,

Summary

- A broad overview of computer systems, and brief history of development of computer systems
- Basic concepts of *information*, *hardware*, *software* and *algorithm*
- *Fetch* and *Execute* phases of instruction
- Sub-systems of a computer system: *processor*, *main memory*, *secondary memory*, *input & output* facilities
- D*ata paths* connect these sub-systems together
- *von Neumann* model of computer systems