1. Introduction

- Chapter 1 of textbook by John F Wakerly
- The materials in this chapter are not covered in the pre-recorded lectures.
- Students are required to do self-study for this chapter.
- Essential concepts will be discussed in Tutorial 1.

Quick links to concepts

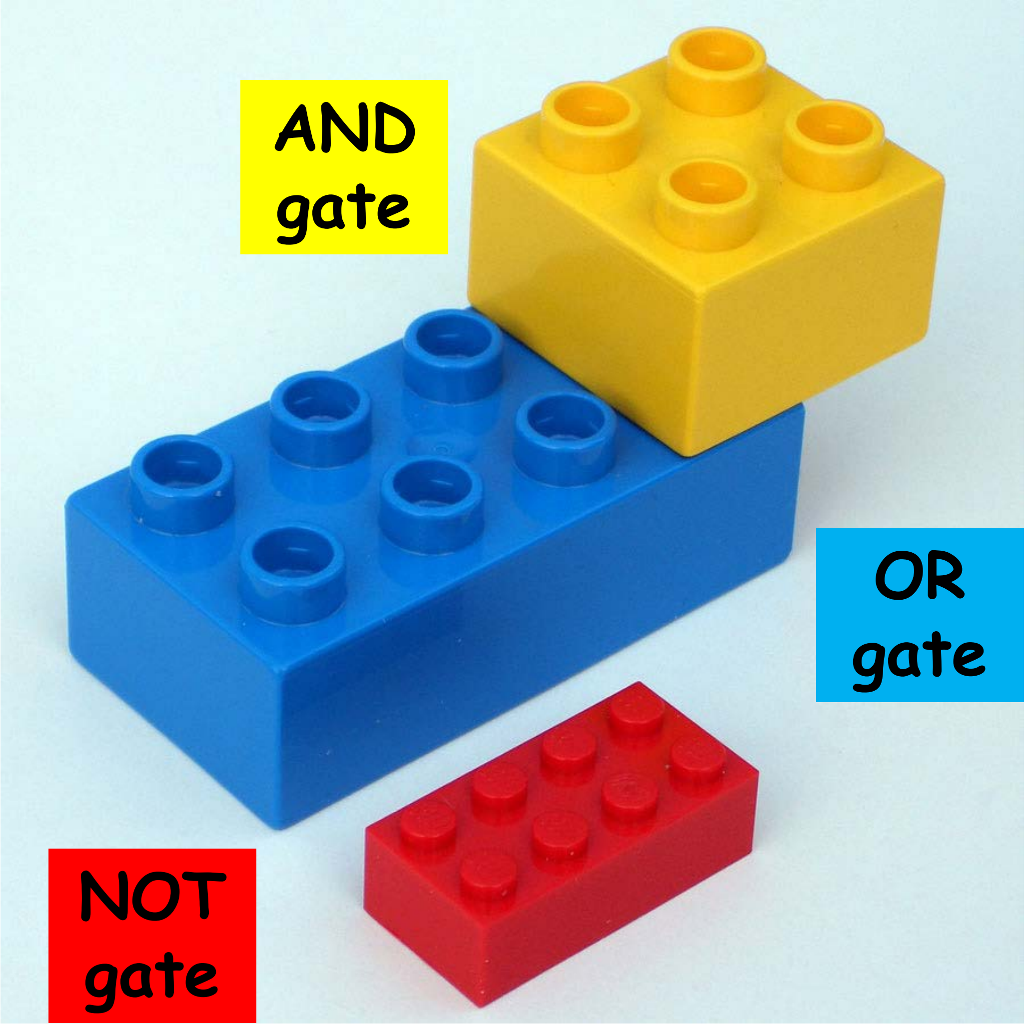
1. Analog versus Digital
2. Digital number systems
3. Electronic aspects and software aspects of digital design
4. Integrated logic circuits
5. Programmable logic devices
6. Serial and parallel data transfer

About Digital Design

- **Digital Design is also known as Logic Design, or Digital Electronics.**
- **Why do you need to learn it?**

  - it provides the fundamentals of designing modern digital gadgets and computer systems, including digital watches, smart phones, tablet PCs, laptops, digital cameras, remote controllers etc.

Digital design is like playing with building blocks…



You can create almost anything… using the basic logic gates

Limited only by your imagination!

A Successful Digital Designer

A successful digital designer should be competent in:

- **- Debugging (systematic approach – not by trial-and-error)**
- **- Business requirements & practices (documentation, specifications)**
- **- Risk taking (in making design decisions)**
- **- Communication (both directions: speak and listen)**

Analog versus Digital

- **Analog quantities happen all around us, examples are**

  - time, temperature, pressure, brightness of lights, loudness of sounds, etc.
- **An analog quantity changes over a <u>continuous</u> range of values.**

- **Digital quantities are countable, examples are**

  - the number of people in a lecture theatre

  - the number of letters in a word

  - the number of AUs you need to obtain in order to graduate

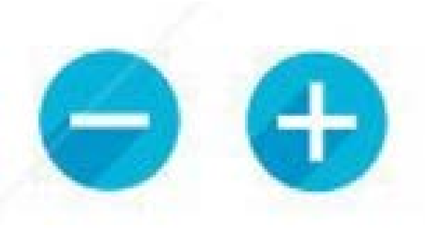
Analog versus Digital

Analog:

this was how people adjusted the volume of their speakers

Digital:

this is how people tend to adjust the volume of their speakers



- **A digital quantity changes in <u>discrete</u> steps.**
- **Analog quantities can be represented in digital format by using <u>sampling and quantisation</u>.**

- **Examples of analog quantities that have been digitised:**

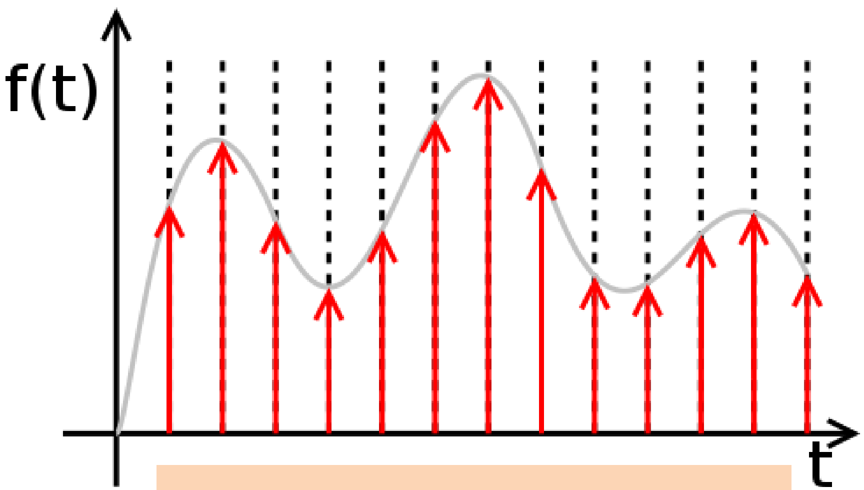
  - digital clock/watch (time is analog)

  - digital thermometer (temperature is analog)

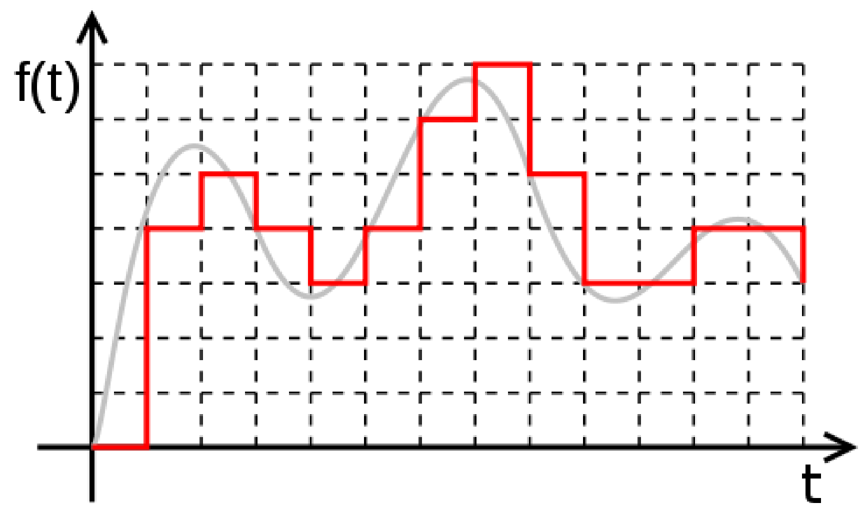
  - digital camera (light intensity is analog)

  - digital audio/video recordings (light and sound intensities are analog)

- f(t) is an analog signal continuously varying with time (faint curve)
- Sampling f(t) at periodic intervals will generate the discrete time signal (red arrows)
- Quantisation of the discrete time signal will produce the digital signal (red lines)



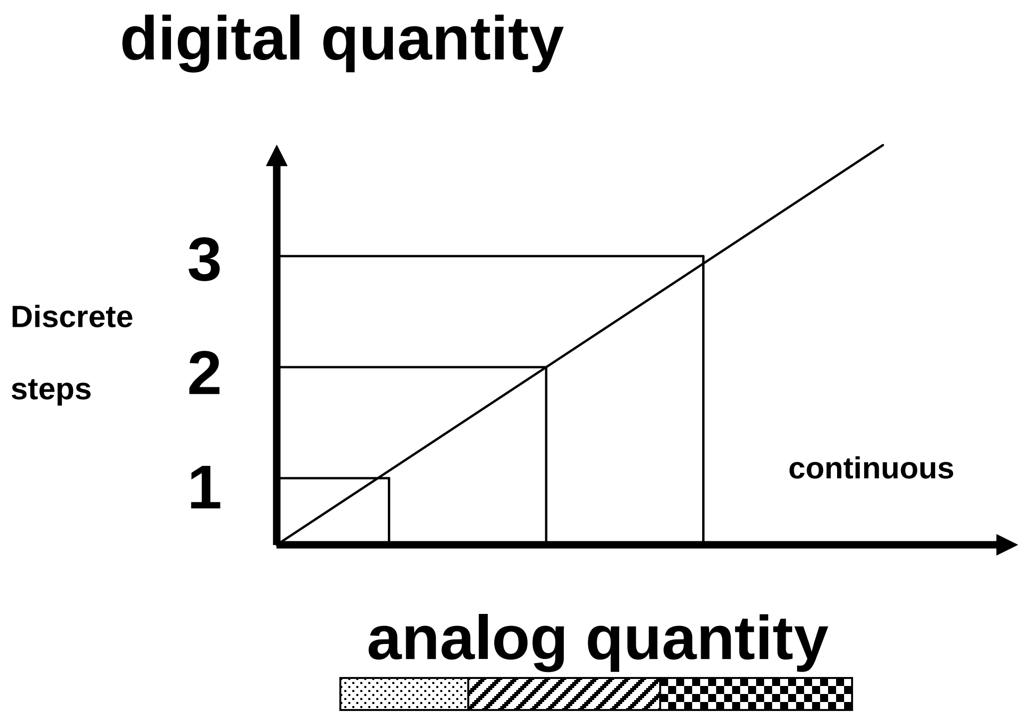
A Discrete-time Signal



A Digital Signal

What we should be aware about Quantisation

- A range of analog values is lumped together and assigned a representative digital value. For example, 0V to 0.8V is assigned 0, 2V to 5V is assigned 1
- A many-to-one mapping (using the above example, 2V is mapped to 1, similarly 3V, 3.8V, 5V are also mapped to 1)
- A finite amount of precision is lost in the process (1 can mean 2V, 1 can also mean 5V)
- Example on next page:



All the values in the range are quantised to 1

e.g. 0 – 3.499999

All the values in the range are quantised to 2

e.g. 3.5 – 9.999999

All the values in the range are quantised to 3

e.g. 10 – 15.999999

Advantages of Digital Techniques Over Analog Techniques

- **Easier to design**
- **Storage is easy**
- **Greater accuracy & precision**
- **Programmability**
- **Less susceptible to noise**
- **VLSI (very large scale integration) technology**

  - high speed

  - low cost

  - small size

Limitations of Digital Technique

- **The real world is mainly analog in nature, hence there is a need to**

  - convert analog inputs to digital form

  - process the digital information

  - convert the digital result back to analog form

- **The advantages of digital techniques usually outweigh the additional time, complexity and expenses involved in A(analog) to D(digital) and D to A conversions.**

You may like to view these:

[Digital and Analog Signals (Austin Lutz) - YouTube]

http://www.diffen.com/difference/Analog_vs_Digital

http://www.cnet.com/news/digital-vs-analog-audio-which-sounds-better/#!

Digital Number Systems

- Number systems commonly used in digital quantities are
  - decimal (base 10) 10 symbols: 0, 1, 2, 3, …, 9
  - binary (base 2) 2 symbols: 0, 1
  - octal (base 8) 8 symbols: 0, 1, 2, 3, …, 7
  - hexadecimal (base 16)
  - 16 symbols: 0, 1, 2, 3, …, 9, A, B, C, D, E, F
- The decimal system is most commonly used in daily life because we have 10 fingers.

Position-value system

- **A single digit cannot represent a large value, hence multiple digits are needed.**
- **In a position-value number system, the value of a digit depends on the position of the digit. Example**
- **in the decimal number 123**

1: most significant digit (left most digit) – MSD

- it carries a weight of “hundred”

2: this digit carries a weight of “ten”

3: least significant digit (right most digit) – LSD

- it carries a weight of “one”

Position-value system

Example:

In the binary number 10

1: most significant bit (left most digit) – MSB

- it carries a weight of “two”

0: least significant bit (right most digit) – LSB

- it carries a weight of “one”

You will learn more about numbers in the self-study topic 2a.

You may like to view these:

[COUNTING SYSTEMS AND NUMERALS]

[Numbers' & Numeral systems' history and curiosities]

Electronic aspects

- **In digital circuits, the binary number system is preferred because only 2 symbols are needed**
- **Simpler electronic circuit design**
- **A binary digit (commonly known as bit) is either 0, or 1.**
- **4 bits make a nibble, e.g. 1001**
- **8 bits make a byte, e.g. 1100 0101**
- **In digital electronic circuits, the binary information is represented by voltage or current.**

- **Examples:**

  0 volt – 0.8 volt represent 0

  2 volts – 5 volts represent 1

  For TTL devices

  0 volt – 1.5 volts represent 0

  3.5 volts – 5 volts represent 1

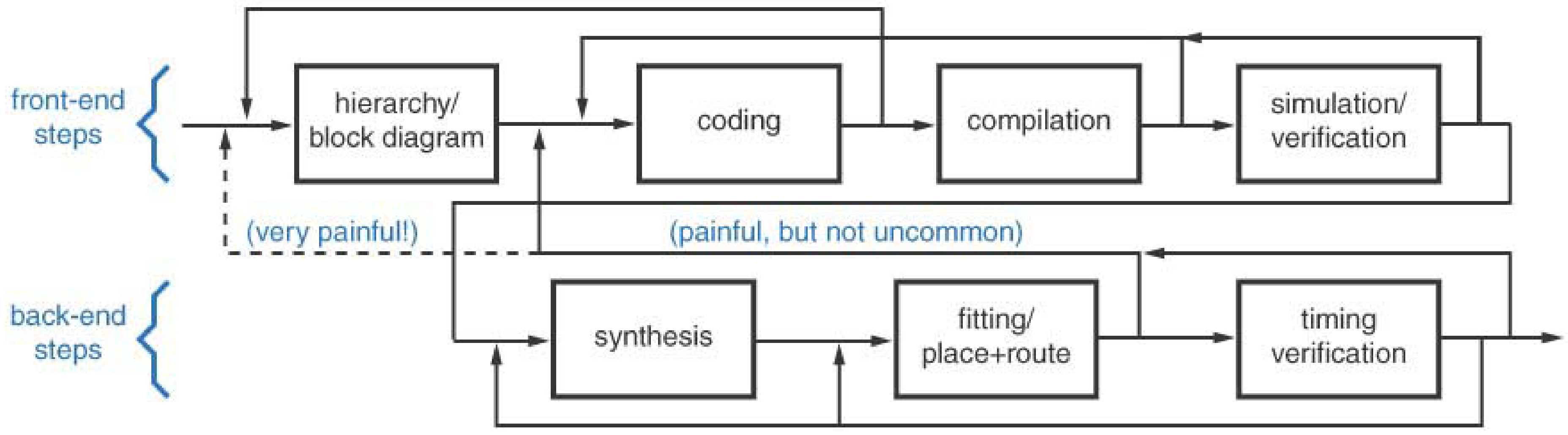
  For CMOS devices

- **The exact voltage level (e.g. 0.1 or 0.5 volt) in Digital Logic is not as important compared to Analog Systems.**

Software aspects

- Modern digital design involves computer-aided design (CAD) software tools
- Schematic entry: use a software tool to draw circuit connections diagrams
- HDL: use hardware description language to describe the logic circuit (e.g. Verilog)
- Synthesizer: creates a circuit realisation based on the above inputs

- **Simulator: predicts the electrical and functional behaviour of a circuit without actually building it**
- **Test bench: a software environment to test the simulated circuit's functional and timing behaviour**
- **You will use some of these tools in the lab experiments**
- **See Fig. 1.19 on the next page.**



From *Digital Design*: *Principles and Practices*, Fourth Edition, John F. Wakerly, ISBN 0-13-186389-4.

Fig. 1.19 HDL-based design flow

Integrated Logic Circuits

- Logic circuits are usually fabricated as integrated circuits (ICs) using various semiconductor technologies – see Fig. 4.29 on next page
- You will use some of these ICs in the lab experiments
- The circuit's logic can range from very simple to very complex

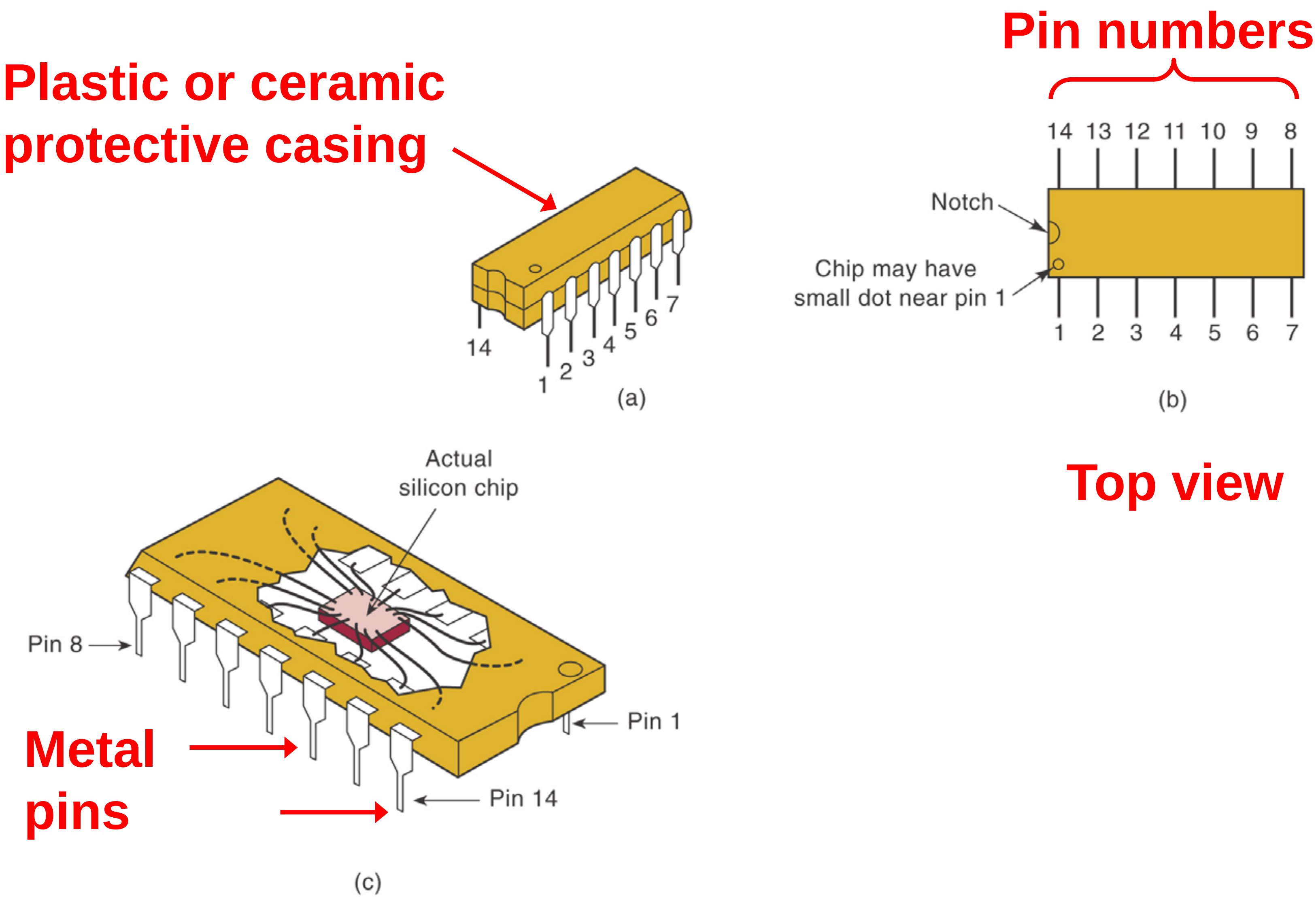


Figure 4.29: (Tocci 10th Ed) Dual-in-line Package

Programmable Logic Devices

- **In some integrated circuits (ICs), the circuit's logic function can be changed, i.e. programmable**
- **This allows bugs to be fixed or circuit behaviour to be modified without physically replacing or rewiring the device**
- **An example is FPGA, field-programmable gate array**
- **You will be using it in the lab experiments**

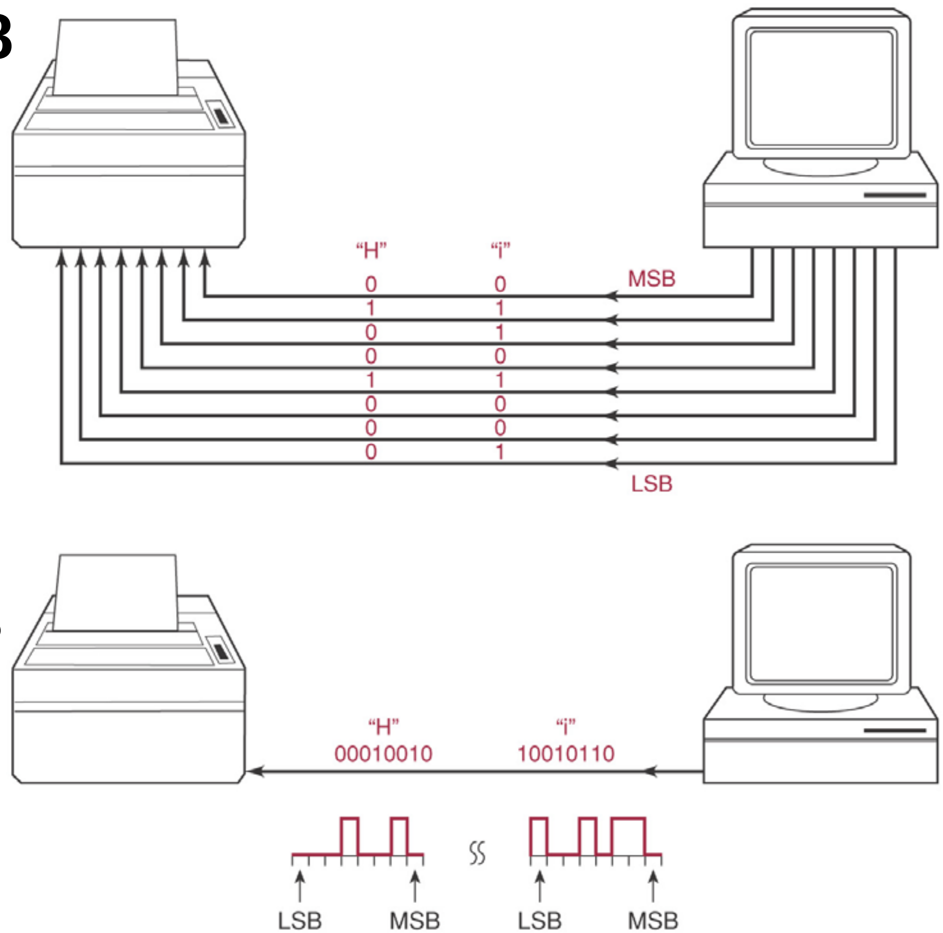
Digital Data transmission

- Serial versus Parallel
- **Parallel**: think 4 checkout counters at the supermarket. 4 customers can be served at the same time.
- **Serial**: think 1 checkout counter at the supermarket. Only 1 customer can be served at any time.
- Trade off is **Simplicity/Cost** versus **Speed**

  E.g. Fig. 1.10, Transmission of 8 bits of data.

Parallel: all 8 bits transmitted at the same time

Advantage: High speed



Serial: 8 bits transmitted one bit at a time

Advantage: Low cost

Figure 1.10 (Tocci 10th Ed) Parallel and Serial Transfer

Example:

- **1 serial line to transmit 8 bits, say at 1 bit per millisecond. Total time taken to transmit is 8 milliseconds. But 1 serial line costs only, say $1 (low cost option).**

- **8 parallel lines can transmit all 8 bits simultaneously in one millisecond. But 8 lines may cost $8 (high speed option).**
- **Data Transfer Methods - YouTube**

Further explorations (optional)

- How stuffs work - analog and digital
- http://en.wikipedia.org/wiki/Positional_notation
- How AMD makes microprocessors (youtube)
- Moore's law - Wikipedia, the free encyclopedia
- Lec-39 introduction to fpga - YouTube