Imperial College London

Lecture 1: Overview – Digital Concepts

Digital Electronics I
Professor Peter Cheung
Department of EEE
Imperial College London
(Slides based on Floyd & Tocci)

Imperial College London

Aim

- to give a first course in **digital electronics** providing you with both the knowledge and skills required to design simple digital circuits and preparing you for a second, more advanced, course next year.

Imperial College London

Objectives

- to impart to you a **formalism of logic** enabling you to **analyse logical processes**
- to enable you to implement **simple logical operations** using combinational logic circuits
- to enable you to understand common forms of **number representation** in digital electronic circuits and to be able to convert between different representations
- to enable you to understand the logical operation of simple **arithmetic** and other **MSI circuits (Medium Scale Integrated Circuits)**
- to impart to you the concepts of **sequential circuits** enabling you to analyse sequential systems in terms of **state machines**
- to enable you to **implement synchronous state machines** using flip-flops

Imperial College London

Course Content

- 15 Lectures
  1. Overview
  2. Introduction to Data Representation
  3. Boolean Algebra and Combination Logic 1
  4. Boolean Algebra and Combination Logic 2
  5. Combinational Logic Gates and Implementation
  6. More Gates and Multiplexers
  7. Data representation 2
  8. MSI Devices
  9. Programmable Devices
  10. Sequential Circuits
  11. State machines 1
  12. State machines 2
  13. Design of Synchronous Sequential Circuits
  14. Application Examples
  15. Revision

Tutorial Questions

- accompany each lecture
- a chance to practice the techniques studied
- graded according to difficulty:
  - * easy, only a little interesting
  - ** harder, more interesting
  - *** challenge, very interesting
- completion of all * and ** questions is essential to your success
- completion of *** questions indicates a very good understanding
- answers given out shortly after questions
- not assessed

Examination

- In the summer term
- past papers available in advance to show the style

Lectures

- Fifteen lectures of about 50 minutes each
  - copies of the overhead slides given out
  - some blanks in the slides for you to fill in, for example:
  - the truth table for an AND gate is:

X	Y	Z=X.Y
0	0	0
0	1	0
1	0	0
1	1	1

- references given to the course book as we go along
- you are expected to read the relevant sections of the book as “homework” just after the lecture

Study Groups

- A chance to ask questions about
  - the work presented in lectures
  - the tutorial questions

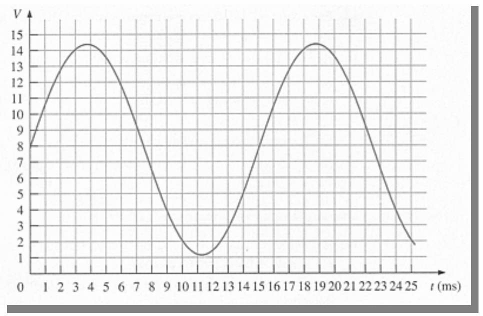
Imperial College London

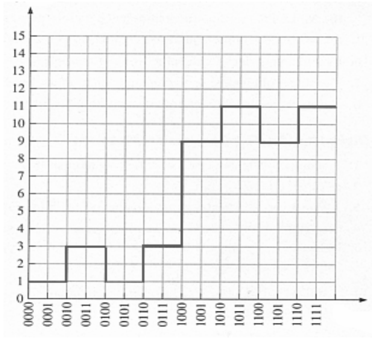
Text Books

- "Digital Systems – Principles and Applications", 9th Ed, R. J. Tocci, N. S. Widmer, G. Moss, Pearson, 2004 (~£45)
- "Digital Fundamentals with PLD Programming", T.L. Floyd, Prentice Hall, June 2005 (~£45)

Imperial College London

Digital and Analog Quantities





Analogue quantities have continuous values

Digital quantities have discrete sets of values

Imperial College London

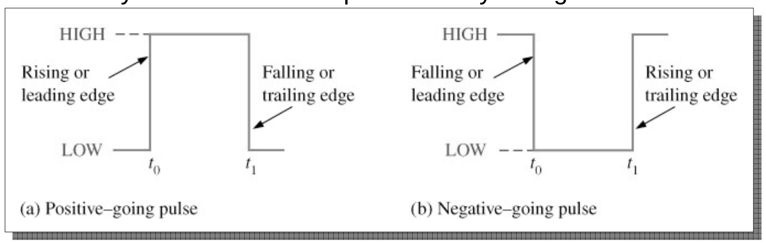
Binary Digits, Logic Levels, and Digital Waveforms

- The conventional numbering system uses ten digits: 0,1,2,3,4,5,6,7,8, and 9.
- The binary numbering system uses just two digits: **0** and **1.**
- The two binary digits are designated **0** and **1**
- They can also be called LOW and HIGH, where **LOW = 0** and **HIGH = 1**

Imperial College London

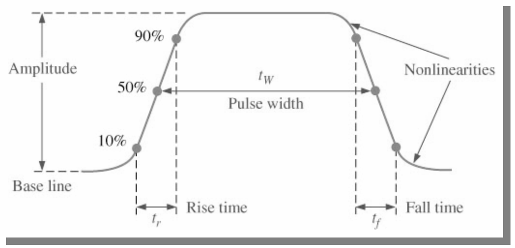
Binary Digits, Logic Levels, and Digital Waveforms

Binary values are also represented by voltage levels



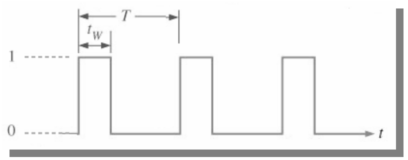
Major parts of a digital pulse

- Base line
- Amplitude
- Rise time (t_r)
- Pulse width (t_w)
- Fall time (t_f)



Binary Digits, Logic Levels, and Digital Waveforms

- t_w = pulse width
- T = period of the waveform
- f = frequency of the waveform



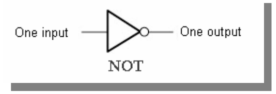
$$f = \frac{1}{T}$$

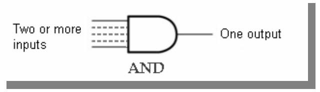
The duty cycle of a binary waveform is defined as:

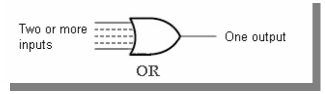
$$\text{Duty cycle} = \left(\frac{t_W}{T}\right)100\%$$

Basic Logic Operations

There are only three basic logic operations:

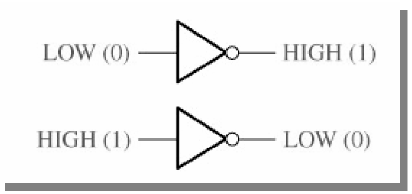






Basic Logic Operations

The NOT operation

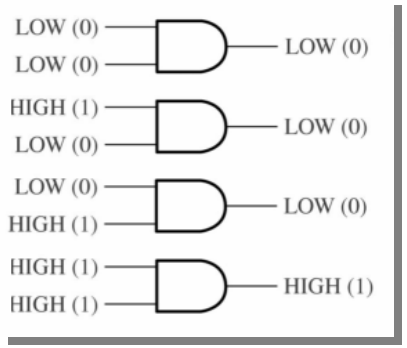


- When the input is LOW, the output is HIGH
- When the input is HIGH, the output is LOW

The output logic level is always opposite the input logic level.

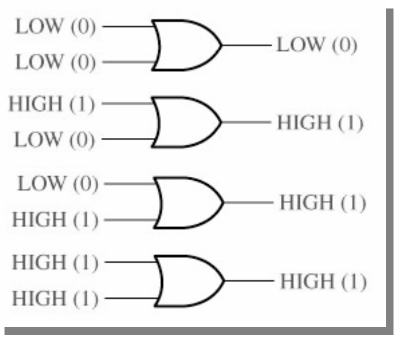
Basic Logic Operations

- The AND operation
  - When any input is LOW, the output is LOW
  - When both inputs are HIGH, the output is HIGH



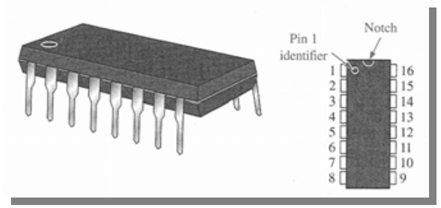
Basic Logic Operations

- The OR operation
  - When any input is HIGH, the output is HIGH
  - When both inputs are LOW, the output is LOW

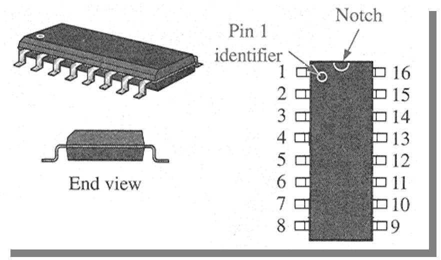


Fixed-Function Integrated Circuits

- Dual in-line package (DIP)

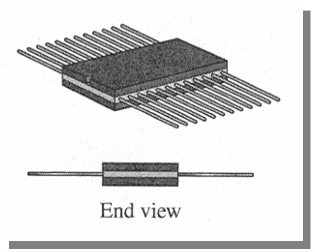


- Small-outline IC (SOIC)

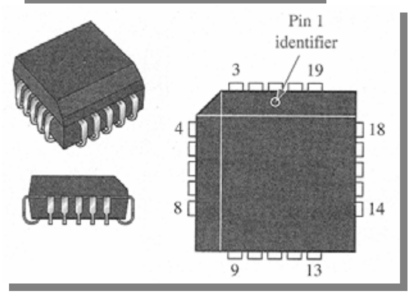


Fixed-Function Integrated Circuits

- Flat pack (FP)

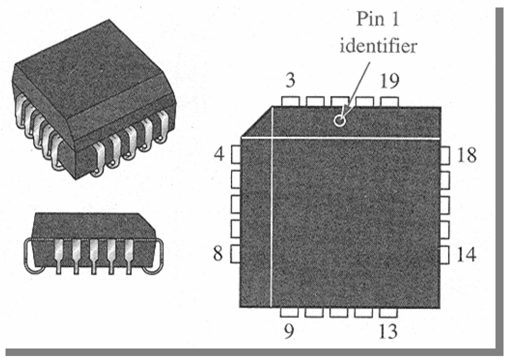


- Plastic-leaded chip carrier (PLCC)



Fixed-Function Integrated Circuits

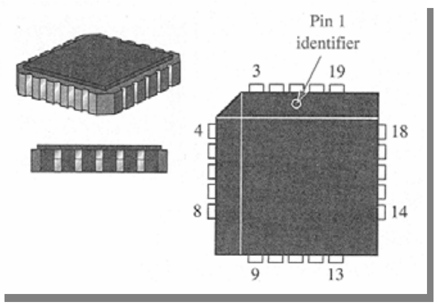
- Plastic-leaded chip carrier (PLCC)



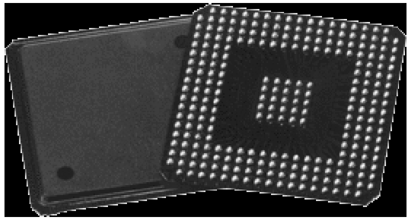
Imperial College London

Fixed-Function Integrated Circuits

- Leadless-ceramic chip carrier (LCCC)



- Ball Grid Array (BGA)



Imperial College London

History

- The first electronic logic was implemented using valves or relay as switches
  - slow by today's standards
  - large
  - got hot
  - relatively unreliable
- Transistor switches used now
  - many transistors can be "integrated" onto a single chip of silicon
  - fast (switch on and then off in around < 100 picosecond)
  - very small (order of 0.1 micron)
  - can get warm
  - very reliable

Imperial College London

Example Applications

- Numeric
  - Calculators for addition, multiplication etc.
  - Aircraft navigation systems for calculating position, ETA etc.
  - Computers for averaging your exam marks
- Non-numeric
  - Parking meters for timing
  - Satellite TV encoding and decoding for revenue protection
  - Disk drives for controlling the rotation and head position

Imperial College London

Design Example: Traffic Light Controller

- Specification
  - The traffic light points in 4 directions (N, S, E, W)
  - The lights on N and S are always the same, as are E and W
  - It cycles through the sequence green-yellow-red
  - N/S and E/W are never both green or yellow
  - Green lasts 45 seconds, yellow 15 seconds, red 60 seconds

- What are the outputs?
  - 12 (one for each light) but only 6 are unique
- What are the inputs?
  - start the controller (reset)
  - timing inputs (clocks)
- What about
  - performance
  - reliability
  - cost
  - power consumption
  - size, etc ?

- What about the logic?
  - IF N/S is green
  - AND E-W is red
  - AND 45 seconds has expired since the last light change
  - THEN the N/S lights should be changed from green to yellow
- What about the digital techniques to implement this?
  - It looks like a computer programme (that's logical!)
  - We need to form logical combinations of inputs
  - We need to conditionally set outputs according to the logical results