Universidad
Carlos III de Madrid

Introduction to digital integrated circuits

INTEGRATED CIRCUITS AND MICROELECTRONICS

Outline

- 1.1 Integrated Circuits. Advantages and disadvantages over non integrated circuits.
  - Digital Circuits Implementation
  - Advantages of ICs
  - Moore's Law
- 1.2 Design process of an integrated circuit. Abstraction Levels.
  - Bottom-up and bottom-down methodologies.
  - Design process: steps and tools.
  - Abstraction levels

Digital Circuits Implementation (I)

- Discrete components (standard)
  - 74xx, 54xx
- Integrated Circuits
  - ASIC: Application Specific Integrated circuit
- Programmable circuits
  - PLD, SPLD: (Simple) Programmable Logic Devices
  - CPLD: Complex Programmable Logic Devices
  - FPGA: Field Programmable Gate Array

Digital Circuits Implementation (II)

- Microprocessor systems
  - Microprocessor and additional components (standard components)
  - SoC: System on Chip (ASIC)
  - SoPC: System on Programmable Chip (FPGA)
    - Soft core
    - Hard core
- Design:
  - Board Level Hardware Design
  - Software Design
    - Tools: compilers

Digital Circuits Implementation (III)

	Discrete components	CPLD	FPGA	ASIC
Integration (Number of logic gates)	Very low	Medium	High	Very High
Speed	Low < 100 MHz	Medium <200 MHz	Medium <500 MHz	Very High <4 GHz
Power consumption	High	Medium	Medium	Low
Initial cost (design+prototype)	High	Medium	Medium	Very High
Cost per unit	High	Medium	Medium	Very Low

Moore's Law (Intel co-founder)

- The maximum number of transistors that can be integrated in a I.C. will double about every 2 years (20 months)

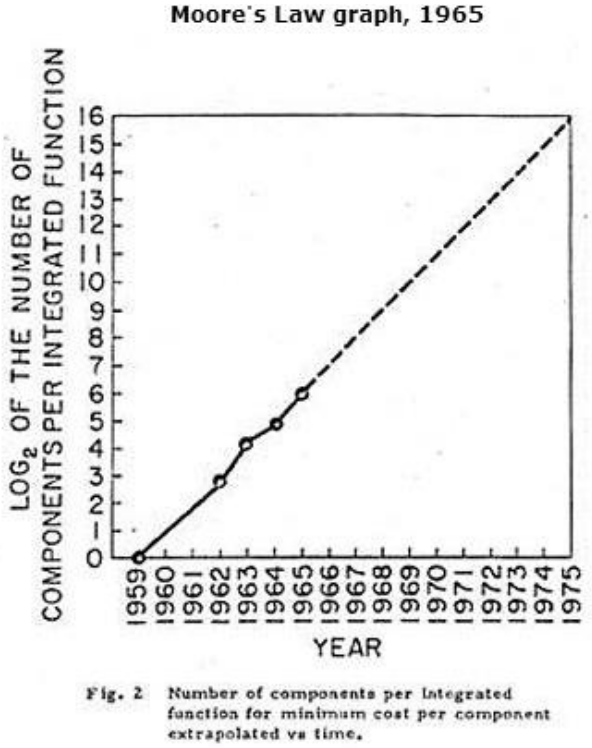
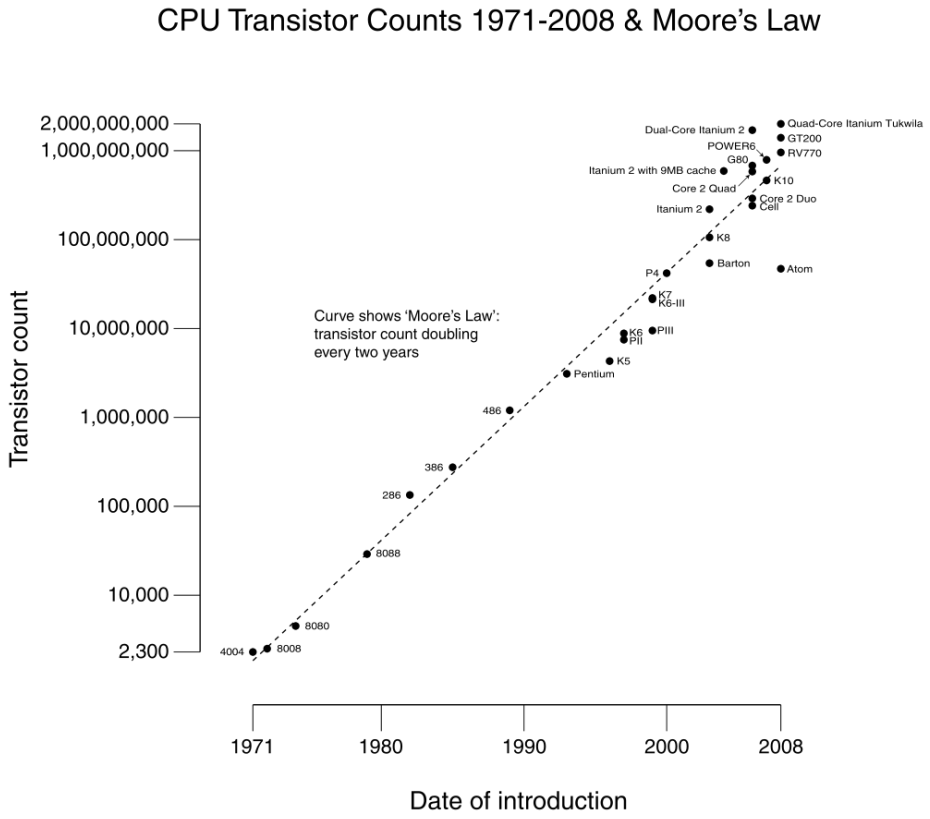
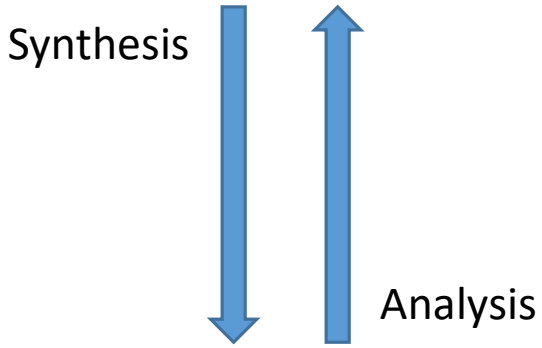


Fig. 2 Number of components per integrated function for minimum cost per component extrapolated vs time.



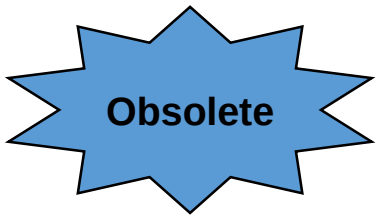
Abstraction Levels

- Levels:
  - System
  - Algorithm
  - Register Transfer Level (RTL)
  - Logic: logic gates
  - Physical: transistors

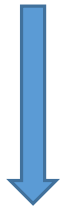


Design Process

- Bottom-up Methodology
  - Specification
  - Block partition
  - Logic gates design (schematic)
  - Block assembly
  - Physical design



- Top-down Methodology
  - Specification
  - Architectural design
  - Detailed design
  - Physical design



  - Use of Hardware Description Languages (HDL)
  - Intensive use of simulation and automatic synthesis

Design process of integrated circuits

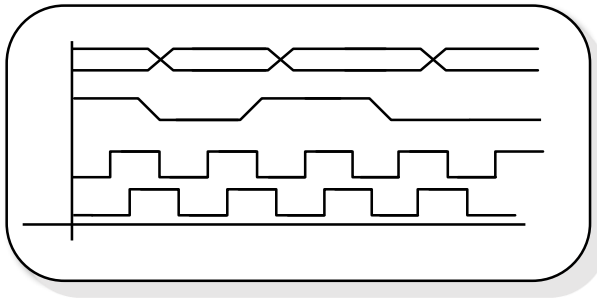
- Functional specification
  - System description, chronograms
- Architectural design
  - RTL design (registers, buses, state machines...)
- Detailed design
  - Automatic Synthesis
  - Logic Design (gates, flip-flops, ...)
- Physical design
  - Transistors, place & route

Automatic Tools

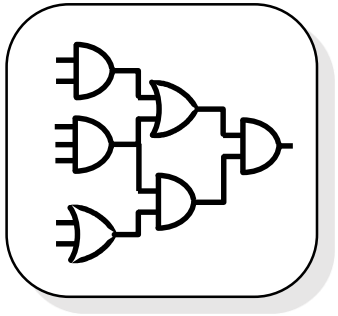
- Manufacturing and test, or programming
- Operation

Design tools

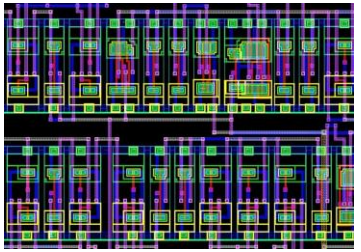
- Simulation
  - Functional description
  - RTL description (synthesizable)
  - List of gates (netlist)
  - With or without delays



- Synthesis
  - Transform RTL descriptions into logic gates
  - Optimize logic for area and/or speed
  - Determine gate delays



- Place & Route/Implementation
  - Place and interconnect logic gates
  - Extract delays of interconnections



Hardware Description Languages (HDLs)

- HDLs allow us to:
  - Design at a higher abstraction level -> Increase productivity
  - Simulate designs for operation validation
    - Simulation is more efficient when performed at a higher abstraction level
  - Synthesize designs to obtain an optimal implementation, depending on the target:
    - Area
    - Speed
  - Increase portability and reuse
- Essential for modern design for:
  - Complex designs(>10K gates)
  - ASIC or CPLD/FPGA technologies

Goals

- Basic digital circuits
  - Combinational circuits:
    - Logic gates, multiplexers, decoders, arithmetic circuits…
  - Sequential circuits:
    - Flip-flops, registers, counters, finite state machines, memories
- VHDL
  - Functional description
  - Hierarchy: blocks and components
  - Simulation: testbenches
  - Synthesis
- Circuit Design
  - Datapath, control