Digital System Design
Lecture 1: Introduction

Amir Masoud Gharehbaghi
amgh@mehr.sharif.edu
Table of Contents

- Introduction to Digital Systems
- Digital Versus Analog Systems
- Contemporary Digital Design
- Modeling Abstractions
- Abstraction Levels
- Y Chart
System

- System: a set of related parts that actuate as a whole to achieve a given goal.
- System has:
  - Inputs
  - Outputs
  - Behavior
- Behavior: a function that translates inputs to outputs.
System (cont.)

- An entity consisting of Hardware and Software
  - Hardware:
    - High speed
    - Low power consumption
    - Less price (probably)
  - Software:
    - Flexible
    - Easy to modify and upgrade
Hardware Systems

- Components are Electronic blocks
  - Analog
  - Digital
  - Mixed-Signal
Analog Systems

- Process time-varying signals
  - Take value across a continuous range
    - Voltage
    - Current
    - ...
  - Continuous time

![Graph showing continuous voltage over time]
Digital Systems

- The same as analog systems.
- But they pretend they don’t!
  - Function over finite values.
  - In discrete time domain.

![Diagram showing a column of values: 1, 0, 1, with time on the x-axis and voltage on the y-axis. The values are represented by a rectangular figure at different voltage levels.]
Mixed-Signal Systems

- Have both analog and digital parts.
  - Digital part:
    - Control and data process. (DSP)
  - Analog part:
    - Sensing and actuating environment.
Advantages of Digital Systems

- High noise immunity
- Adjustable precision
- Ease of design (automation) and fabrication, therefore, low cost
- Better Reliability
- Less need to calibration and maintenance
- Ease of diagnosis and repair
- Easy to duplicate similar circuits
- Easily controllable by computer
Disadvantages of Digital Systems

- Lower speed
- Needs converters to communicate with real world, therefore more expensive and less precision
  - Digital to Analog (D/A)
  - Analog to Digital (A/D)
Digital Abstraction

- Digital circuits actually deal with analog voltages and currents.
- Digital abstraction allows analog behavior to be ignored (most cases).
- Simplest form of abstraction is Binary system, two values exists:
  - 0, Low, False
  - 1, High, True
Digital Abstraction (cont.)

- Association of a range of analog values with each logic value (0, 1).
- The difference between the range boundaries is called **noise margin**.
Synchronous vs. Asynchronous

- Synchronous system: Elements change their values at certain specified times (clock event).
- Asynchronous systems: Outputs can change at any time.

Example: digital alarm set to 13:59
- Synchronous: 12:59 -> 13:00 -> 13:01
Gates

- Gates are the most basic digital devices.
- A gate has one or more inputs and produces an output that is a function of the current input values.
- A gate is a combinational circuit, because its output depends only on the current input combination.
A flip-flop is a device that stores either a 0 or a 1. The state of a flip-flop is the value currently stored. The stored value can only be changed at certain times, regulated by a "clock" input. A digital circuit that contains flip-flops is called a sequential circuit. The output of a sequential circuit depends, at any time, not only on its current input but also on the past sequence of inputs that have been applied to it. A sequential circuit has memory of past events.
Contemporary Digital Design

- **Major changes in recent years:**
  - More complex designs (SoC, SoPC)
  - Shorter Time-to-Market (TTM)
  - Cheaper products

- **Scale**
  - Pervasive use of computer-aided design tools over hand methods
  - Multiple levels of design representation
Contemporary Digital Design (cont)

- **Time**
  - Emphasis on abstract design representations
  - Programmable rather than fixed function components
  - Automatic synthesis techniques
  - Importance of sound design methodologies

- **Cost**
  - Higher levels of integration
  - Use of simulation to debug designs
EDA/CAD Tools

- CAD (Computer Aided Design) tools are nowadays essential part of digital design.
- HDLs (Hardware Description Language)
- Design Automation (DA) helps us improve productivity and also predictability of behavior
EDA/CAD Tools (cont.)

- Schematic Entry
- HDL Compilers and Simulators
- Synthesis Tools
- Timing Analyzers
- Verification Tools
  - Simulators
  - Testbenches
  - Design Checkers
  - Formal and Semi-Formal Verifiers
Integrated Circuits (ICs)

- An **IC** is a collection of gates fabricated on a single silicon chip.

- Different sizes of ICs:
  - **SSI** (Small Scale Integrated)
    - Small number of gates (1 to 30 gates)
  - **MSI** (Medium Scale Integrated)
    - Decoder, counter, register (30 to 300 gates)
  - **LSI** (Large Scale Integrated)
    - Small memories, PLDs (300 to 300k gates)
  - **VLSI** (Very Large Scale Integrated)
    - Microprocessors, memories (> 1M transistors)

- P4 has more than 42M transistors!
SSI ICs

- Dual in-line pin (DIP) packages.
- A pin diagram shows the assignment of device signals to package pins.
- Nowadays, SSI ICs are used as "glue" to tie together larger components in complex systems.
- SSI ICs have been largely supplanted by PLDs (Programmable Logic Devices).
Programmable Logic Devices (PLDs)

- Some ICs can have their logic function "programmed" into them after they are manufactured.
- Most of them can even be reprogrammed, which allows bugs to be corrected without replacing or rewiring the device.
- PLD (PLA or PAL): two-level structure of AND and OR gates with user-programmable connections.
- CPLDs (Complex PLDs) and FPGAs (Field Programmable Gate Arrays) were devised to accommodate larger systems.
- HDLs and the respective tools allow a design to be compiled, synthesized, and downloaded into a device in a short time.
- This permits rapid prototyping to be a reality.
Design Hierarchy

- System
  - Data Path
    - Data Register
    - Combinational Logic
  - Controller
    - State Register
    - Combinational Logic
Design Process

Abstraction Levels

High
- System Specification
- System
- Functional Modules
- Gate
- Circuit
- Device

Low
# Abstraction Levels

<table>
<thead>
<tr>
<th>Design Levels</th>
<th>Design Descriptions</th>
<th>Primitive Components</th>
</tr>
</thead>
</table>
| System (Architecture)| System Languages
                        | High-Level Languages            | Functional Blocks             |
| Behavioral (Algorithmic)| HDLs
                          | Math Equations                  | Functional Blocks             |
| Functional (RTL)     | HDLs
                        | FSMs                            | Registers
                        | ALUs, Multiplexers,...         |
| Logic                | HDLs
                        | Boolean Equations
                        | Timing Diagrams
                        | Gate Netlist                   | Logic Gates
                        | Flip-Flops                    |
| Circuit (Switch)     | Transistor Netlist                 | Transistors                   |
| Device (Geometrical) | Graphical                          | Polygons                      |
Abstraction Levels (simple)

- Architectural Level
  - Operations implemented by resources
- Logic Level
  - Logic functions implemented by gates
- Geometrical Level
  - Devices are geometrical objects
Modeling Views

- Behavioral view:
  - Abstract function

- Structural view:
  - An interconnection of parts

- Physical view:
  - Physical objects with size and positions
Modeling Views (cont.)

- Correspondence with:
  - Synthesis
  - Analysis
  - Optimization
  - Refinement
  - Physical design
  - Extraction
Y Chart