Computer Aided Design of Electronics
[Datorstödd Elektronikkonstruktion]

Zebo Peng, Petru Eles, and Nima Aghaee
Embedded Systems Laboratory
IDA, Linköping University

www.ida.liu.se/~TDTS01

Electronic Systems
Objectives

- Basic principles of computer-aided design for electronic systems (Electronic Design Automation).
- Electronic system design at high levels of abstraction.
- Synthesis and optimization algorithms.
- Test and design for testability techniques.
- The hardware description language VHDL and its use in the design/synthesis process.

Course Organization

- 10 Lectures (Petru and Zebo):
  - Introduction and basic terminology.
  - VHDL: overview and simulation semantics.
  - Behavioral and structural modeling with VHDL.
  - High-level synthesis of digital systems.
  - Heuristics and optimization algorithms.
  - Testing and design for testability.
- Invited industrial lecture (Björn Fjellborg, Ericsson)
- Laboratory part (Nima):
  - Three seminars on assignments and CAD systems.
  - Lab assignments.
Recommended Literature

  - Other VHDL books can also be used.
  - A VHDL Cookbook is available at the course website.
- Articles to be distributed, including:
  - High-level synthesis of digital circuits.
  - Optimization heuristics.
  - Design for test and built-in self-test.
- Lecture notes (www.ida.liu.se/~TDTS01).

Lecture I

- Trend in microelectronics
- The design challenges
  - Different design paradigms
- The test problems
The Technological Trend

Moore’s Law:
Number of transistors per chip would double every 1.5 years

Clock Frequency (every 2 years)
Performance
Memory capacity

Intel Microprocessor Evolution

Images courtesy of Intel Corporation
The SIA Roadmap

<table>
<thead>
<tr>
<th>Year</th>
<th>Feature size (nm)</th>
<th>Logic: trans/cm²</th>
<th>Trans/chip</th>
<th>#pads/chip</th>
<th>Clock (MHz)</th>
<th>Chip size (mm²)</th>
<th>Wiring levels</th>
<th>Power supply (V)</th>
<th>High-perf pow (W)</th>
<th>Battery pow (W)</th>
</tr>
</thead>
<tbody>
<tr>
<td>2002</td>
<td>130</td>
<td>18M</td>
<td>67.6M</td>
<td>2553</td>
<td>2100</td>
<td>430</td>
<td>7</td>
<td>1.5</td>
<td>130</td>
<td>2</td>
</tr>
<tr>
<td>2005</td>
<td>100</td>
<td>44M</td>
<td>190M</td>
<td>3492</td>
<td>3500</td>
<td>520</td>
<td>7-8</td>
<td>1.2</td>
<td>160</td>
<td>2.4</td>
</tr>
<tr>
<td>2008</td>
<td>70</td>
<td>109M</td>
<td>539M</td>
<td>4776</td>
<td>6000</td>
<td>620</td>
<td>8-9</td>
<td>0.9</td>
<td>170</td>
<td>2.8</td>
</tr>
<tr>
<td>2011</td>
<td>50</td>
<td>269M</td>
<td>1523M</td>
<td>6532</td>
<td>10000</td>
<td>750</td>
<td>9</td>
<td>0.6</td>
<td>175</td>
<td>3.2</td>
</tr>
<tr>
<td>2014</td>
<td>35</td>
<td>664M</td>
<td>4308M</td>
<td>8935</td>
<td>16900</td>
<td>900</td>
<td>10</td>
<td>0.5</td>
<td>183</td>
<td>3.7</td>
</tr>
</tbody>
</table>

SIA = Semiconductor Industry Association

System on Chip (SoC)

- Hardware
  - Microprocessor
  - ASIC
  - Analog circuit
  - Sensor
- Software
  - Embedded memory
  - DSP
  - Network
  - High-speed electronics

Source: Stratus Computers
Lecture I

- Trend in microelectronics
- The design challenges
- Different design paradigms
- The test problems

Main Design Tasks

- System specification (functional and requirement)
- Hardware/software trade-offs
- Architecture selection and exploration
- Synthesis and optimization
- Implementation
- Testing and design for testability
- Analysis and simulation
- Verification and validation
- Design management: storage of design data, cooperation between tools, design flow, etc.
Design Objectives

- **Unit cost**: the cost of manufacturing each copy of the system, excluding NRE cost.
- **NRE cost (Non-Recurring Engineering cost)**: The one-time cost of designing the system.
- **Size**: the physical space required by the system.
- **Performance**: the execution time or throughput.
- **Power**: the amount of power consumed by the system.
- **Testability**: the easiness of testing the system to make sure that it works correctly.
- **Flexibility**: the ability to change the functionality of the system without incurring heavy NRE cost.
- **Correctness, safety, etc.**

---

The Productivity Gap is Becoming a Canyon

- **Silicon technology is outstripping design capability**

![Graph showing the productivity gap between available silicon gates and design and verification gates over time from 1980 to 2010.](image)
Mixed Technologies for Electronics

- Embed in a single chip: Logic, Analog, DRAM blocks
- Embed advanced technology blocks:
  - FPGA, Flash, RF/Microwave
- Beyond Electronic
  - MEMS (Micro Electro Mechanical Systems)
  - Optical elements

The Challenges

- System complexity
  - Increasing functionality and diversity
  - Increasing performance
- Stringent design requirements
  - Low cost
  - Low power
  - High reliability, testability, and flexibility
- Technology challenges for cost-efficient implementation
  - Deep submicron effects
  - Issues related to process variation
Possible Solutions

- Powerful design methodology and tools.
- Advanced architecture (modularity).
- Extensive design reuse.

Design Paradigm Shift

Lecture I

- Trend in microelectronics
- The design challenges
  - Different design paradigms
- The test problems
Capture and Simulate

- The detailed design is captured in a model.
- The model is simulated.
- The results are used to guide the improvement of the design.
- All design decisions are made by the designers.

Abstraction Hierarchy

- Layout/silicon level — The physical layout of the integrated circuits is described.
- Circuit level — The detailed circuits of transistors, resistors, and capacitors are described.
- Logic (gate) level — The design is given as gates and their interconnections.
Abstraction Hierarchy (Cont’d)

- Register-transfer level (RTL) — Operations are described as transfers of values between registers.

- Algorithmic level — A system is described as a set of usually concurrent algorithms.

- System level — A system is described as a set of processors and communication channels.

Gajski’s Y-Chart

**Behavioral domain**
- Algorithms, processes
- Register-Transfer Spec.
- Boolean Eqn.
- Translator functions

**Structural domain**
- Logic level
  - CPU, Memory, Bus
  - ALU, Reg., MUX
- Circuit level
  - Gate, Flip-Flop
- Transistor
  - Transistor layouts
  - Standard-Cell/Subcell
  - Macro-CMOS chips
- Physical/geometrical domain
  - Board, MCMs

Code snippet:
```
For I=0 To 2 Loop
  Wait until clk’event and clk = ’1’;
  If (rgb[I] < 248) Then
    P = rgb[I] mod 8;
    Q = filter(x, y) * 8;
    End If;
```
The Three Domains

- Structural domain — A component is described in terms on an interconnection of more primitive components.
- Behavioral domain — A component is described by defining its input/output response.
- Physical/geometrical domain — A component is described in terms of its physical placement and characteristics (e.g., shape).

A Typical Top-Down Design Process

Informal Specification

Behavioral Domain

System level

RT-level

Logic level

Circuit level

Transistor functions

Boolean logic

Register-Transfer Spec.

Algorithms, processes

Informal Specification

Structural Domain

Circuit level

Gate/Flip-Flop

Transistor functions

Standard Cell/Subcell

Board, MCMs

Macro-Cell, chips

Physical Domain
Describe and Synthesize Paradigm

- Description of a design in terms of behavioral specification.
- Refinement of the design towards an implementation by adding structural details.
- Evaluation of the design in terms of a cost function and the design is optimized w.r.t. the cost function.

\[ o_1 = (a + b) c + d c; \]
\[ o_2 = (d + f) c; \]
\[ o_3 = (a + b) d + df; \]

High-Level Describe and Synthesize

- Description of a design in terms of behavioral specification.
- Refinement of the design towards an implementation by adding structural details.
- Evaluation of the design in terms of a cost function and the design is optimized for the cost function.

For I=0 To 2 Loop
Wait until clk'event and clk='1';
If (rgb[I] < 248) Then
\[ P = rgb[I] \mod 8; \]
...
Core-based Design

- Utilization of pre-designed and pre-verified cores:
  - Reuse of large IP blocks, such as CPU, DSP, memory modules, communication infrastructure, and analog blocks.
- Divide-and-conquer design methodology.
- Flexibility based on different core description levels:
  - Soft: RT level (synthesizable VHDL/Verilog).
  - Firm: Gate-level netlist.
  - Hard: Layout.
- Legal issues:
  - IP right protection.
  - Liability in case of failures.

Platform-based Design

- A platform is a partial design:
  - for a particular application area;
  - includes embedded processor(s);
  - may include embedded software;
  - customizable to specific requirements.
- A platform captures the good solutions to the important design challenges in a given application area.
- A method for design re-use at all abstraction levels based on assembling and configuring platform components in a rapid and reliable fashion.
  - It reuses architectures.
### Platform-based Design Steps

- Design the platform.
  - A highly configurable system architecture.
  - Optimize for performance, power, etc.
  - Useful for a set of applications.
  - Tools to explore the different configurations.

- Use the platform.
  - Modify hardware components for the customer’s needs.
  - Optimize the software.
  - HW/SW integration and test.

---

### Lecture I

- Trend in microelectronics
- The design challenges
- Different design paradigms
- The test problems
Testing and its Current Practice

- Testing aims at the detection of physical faults (production errors/defects and physical failures).
- Different from the design task, testing is performed on each individual chip, after they are manufactured.
- The common approach to perform testing is to utilize an Automatic Test Equipment (ATE).

Testing of Mixed Technologies

How to test the mixed chip?

- Use multiple ATEs for a single chip: Logic ATE, Memory ATE, Analog ATE, etc.
  - Usually time consuming.
- Employ a super ATE with combined capabilities.
  - Usually very expensive.
High Complexity

- # of transistors increases exponentially.
- # of access port remains stable.
- Implication:
  - # of transistors per pin (Testing Complexity Index) increases rapidly.

![Implications of SIA Roadmap: Testing](chart)

Built-In Self Test (BIST)

- Solution: Dedicated built-in hardware for embedded test functions.
  - No need for expensive ATE.
  - At-speed testing.
  - Concurrent test possible.
  - Support O&M.
  - Support field test and diagnosis.

![Built-In Self Test](chart)
Challenges to CAD Communities

- System specification with very high-level languages.
- Modeling techniques for heterogeneous system.
- Hardware/software co-design.
- Testing issue to be considered during the design process.
- Design verifications -> get the whole system right the first time!
- Very efficient power saving techniques.
- Global optimization.

The Electronics System Designer
Conclusion Remarks

- Much of design of digital systems is managing complexity.
- What is needed: new techniques and tools to help the designers in the design process, taking into account different aspects.
- We need especially design tools working at the higher levels of abstraction.
- If the complexity of the microelectronics technology will continue to grow, the migration towards higher abstraction level will continue.