Introduction to ASIC Design

Dr. Paul D. Franzon

Outline
1. The wonderful world of Silicon
2. Application Specific Integrated Circuits (ASICs)
   - Typical applications, types, decision making
3. ASIC Design Flow

References
1. Smith, Application Specific Integrated Circuits, Chapter 1.

The Wonderful World of Silicon

About every two years, the number of transistors on a CMOS silicon chip doubles and the clock speed doubles….This rate of improvement will continue for the next 20 years.

Technology Drivers:
- Decreasing lithographic feature size, typically measured by the transistor gate length:
  - 0.35 μm …. 0.25 μm …. 0.18 μm 0.15 μm …etc…. 0.050 μm (?)
- Increasing wafer size:
  - 6 inch diameter …. 8 inch diameter ….etc….. 12 inches (?)
- Increasing number of metal interconnect layers:
  - 4 …. 6 …. 8 …… 9 (?)
- Approximately constant cost per wafer to manufacture:
  - About $2,000 - $4,000 per wafer
- Increasing IC yields for ‘large’ (> 1 sq. cm.chips): 60% …. 90%
From the Semiconductor Roadmap

Projections for ‘leading edge’ ICs: (www.sematech.org)

<table>
<thead>
<tr>
<th>Year</th>
<th>1997</th>
<th>1999</th>
<th>2001</th>
<th>2012</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gate length (nm)</td>
<td>200</td>
<td>140</td>
<td>120</td>
<td>35</td>
</tr>
<tr>
<td>Microprocessor total transistors/chip</td>
<td>11M</td>
<td>21M</td>
<td>40M</td>
<td>1.4B</td>
</tr>
<tr>
<td>Largest Chip</td>
<td>22 x 22</td>
<td>25 x 32</td>
<td>25 x 34</td>
<td>25 x 52</td>
</tr>
<tr>
<td>Microprocessor Clock Speed (MHz)</td>
<td>750</td>
<td>1200</td>
<td>1400</td>
<td>3000</td>
</tr>
<tr>
<td>ASIC clock speed (high performance) (MHz)</td>
<td>300</td>
<td>500</td>
<td>600</td>
<td>1500</td>
</tr>
</tbody>
</table>

ASICs vs. what?

Application Specific Integrated Circuit
- A chip designed to perform a particular operation as opposed to General Purpose integrated circuits:
- An ASIC is NOT software programmable to perform different tasks

General Purpose Integrated Circuits:
- Programmable microprocessors (e.g. Intel Pentium Series, Motorola HC-11)
  - Used in PCs to washing machines
- Programmable Digital Signal Processors (e.g. TI TMS 320 Series)
  - Used in many multimedia, sensor processing and communications applications
- Memory (dRAM, SRAM, etc.)

Examples of ASICs:
- Video processor to decode MPEG-2 digital TV signals
- Audio processor to perform Dolby AC3 encoding
- Low power DSP/controller for cell phone
1. Full Custom ASICs

- Every transistor is designed and drawn by hand
- Typically only way to design analog portions of ASICs
- e.g. part of a custom DES (encryption) processor designed as part of a research project at NCSU:

Gives the highest performance but the longest design time
- Typically only used for analog portions and for very high volume parts (e.g. microprocessors) or for small parts to be used in many different designs

2. Standard-Cell-Based ASICs

- or ‘Cell Based IC’ (CBIC) or ‘semi-custom’
- **Standard Cells** are custom designed and then inserted into a library

*These cells are then used in the design by being placed in rows and wired together using ‘place and route’ CAD tools
- Some standard cells, such as RAM and ROM cells, and some datapath cells (e.g. a multiplier) are tiled together to create **macrocells**
- Custom designed blocks (e.g. microprocessors) might be mixed in as well (sometimes called **megacells** or **hard macros**.)
 ASIC Styles

Sample ASIC floorplan:

- Standard Cell designs are usually synthesized from an RTL (Register Transfer Language) description of the design
- ASIC design is much quicker than full custom design
- A full set of masks is still required for fabrication

3. Gate-Array Based ASICs
- In a gate array, the transistors level masks are fully defined and the designer cannot change them
- The design instead programs the wiring and vias to implement the desired function
- Gate array designs are slower than cell-based designs but the implementation time is faster as less time must be spent in the factory
- RTL-based methods and synthesis, together with other CAD tools, are often used for gate arrays.
...ASIC Styles

4. Programmable Logic Devices (PLDs)
   • are off-the-shelf ICs that can be programmed by the user to perform various functions (usually just combinational logic functions)
   • The simplest PLD is an EPROM
   • There are no custom mask layers so final design implementation is a few hours instead of a few weeks
   • Simple PLDs are used for simple functions

5. Field Programmable Gate Arrays (FPGAs)
   • Off-the-shelf chips that the user programs to perform simple functions
   • Can be quite complex, capable of implementing 100,000s of gates
   • Some companies call them ‘complex PLDs’
   • e.g. XILINX:

   *Programmable Interconnect Array:*

   ![Programmable Interconnect Array](image)

   *Configurable Logic Block (CLB):*

   ![Configurable Logic Block](image)

Source: XILINX Application notes
**Research Project….Data Programmable Cache**

**New Ideas**
- Cache lines and FPGA lines alternated
- Cache lines can be used as
  - Cache for embedded CPU
  - Store FPGA configurations
  - Store data for FPGA
- FPGA can switch to any of 4 ‘contexts’ in one clock cycle

**Impact**
- 2x performance over
  - FPGA
  - FPGA + SRAM

---

**Decision Making**

Alternative implementation approaches:
- Program a PC or a microcontroller
  - (some microcontrollers have high-performance 32- or 64-bit CPUs)
- Build a full custom, cell-based or gate array ASIC
- Use a PLD or FPGA

<table>
<thead>
<tr>
<th>Approach</th>
<th>Design Productivity (gates/day)</th>
<th>Manufacturing NRE + lead time</th>
<th>Manufacturing cost/unit</th>
<th>Clock frequency</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cell-based</td>
<td>200</td>
<td>$50k (full mask set) 8 weeks</td>
<td>1x</td>
<td>1x</td>
</tr>
<tr>
<td>Gate Array</td>
<td>200</td>
<td>$10k &lt; 1 week</td>
<td>~1.2x</td>
<td>~0.8x</td>
</tr>
<tr>
<td>FPGA</td>
<td>500</td>
<td>$0 1 day</td>
<td>~5x</td>
<td>~0.3x</td>
</tr>
</tbody>
</table>
Other Issues:

- Design tools and design training
  - Cell-based ICs require more investment but give more capable designs
- Time to Market
  - Profits increase dramatically with every month shaved off time-to-market (about 30%/3-months, according to one study)
  - Approaches:
    - Consider using an FPGA for product introduction, or at least prototyping
    - Enhance design productivity as much as possible
      - people, tools, training, methods and techniques
    - Good design verification is very important for Cell-based ICs

ASIC Design Methodology

Most ASICs are designed using a RTL/Synthesis based methodology

- Design details captured in a simulatable description of the hardware
  - Captured as Register Transfer Language (RTL)
  - The two most common RTLs are Verilog and VHDL
- Automatic synthesis is used to turn the RTL into a gate-level description
  - ie. AND and OR gates, etc.
  - Chip-test features are usually inserted at this point
- Physical Design tools are then used to turn the gate-level design into a set of chip masks (for photolithography) or a configuration file for downloading to an FPGA
- At each stage, extensive verification is performed.
• It's often useful to write a behavioral simulatable specification
  • Captures intent well
  • Gives an independent source of verification vectors

...ASIC Design Flow

HDL Design Capture

- Design Specification
- Behavioral Description
- RTL Description

Simulation Vectors

RTL Functionality Verified?

Yes

No

HDL Design Synthesis

- Constraints
- Logic Optimization
- Logic to Technology
- Timing/Area Optimization
- Insert Scan Chains
- Test vector gen.

Netlist/SDF Functionality/Timing Verified?

Yes

No

Back annotation of timing data
ECE 520 Class Notes

...ASIC Design Flow

Design Implementation

Yes

Floor Planning

Place & Route

Physical Layout

Yes

Layout

No

Functionality/Timing

Verified?

Yes

Chip Production

Back annotation of timing data

ECE 492B / ECE 520 ASIC Design

Objective: Learn how to design a digital ASIC using Register Transfer Languages (Verilog), modern synthesis tools (Synopsys) and modern verification and back end tools (Cadence). Learn how the tools constrain and permit modern design. Survey ASIC architectures and conduct a project in multimedia or communications.

©2000, Dr. Paul D. Franzon, www.ece.ncsu.edu/erl/faculty/paulf.html
**Future Tool Issues**

**Growing Chip Complexity**
- 10M+ transistors/chip very close
- combined with a shortage of designers, and need for shorter design turnaround times
  - **Design Reuse**
    - Components designed to be reused in other designs
    - A methodology for fast integration of library-based designs

**Increasing importance of wiring**
- Wiring parasitic capacitance increasing faster than gate capacitance
- leads to wiring having a strong effect on delay and noise
- Often referred to as the “deep sub-micron” problem because its importance becomes pronounced at gate sizes below 0.18 µm.

**Solution:**
- Short term: More tightly integrated tool flows, better estimators
- Medium term: Combined logical and physical synthesis

**Summary**
- Over the next ten years, product growth will be driven by:
  - Continued explosive growth of the internet
  - Insatiable demand for graphics and multimedia
  - Insidious insertion of electronics and computers into our everyday lives
- Many of the resulting products will require specialized silicon chips to meet performance (speed/size/weight/power/cost) demands - ASICs
- To match this product need, the capability of a silicon CMOS chip will continue doubling every 2-3 years until after 2015.
  - To sell a product at $300-$1,000, it can only include one high value chip
  - Thus product performance is determined by the performance of that one chip
  - AND talk about planned obsolescence!!
- ASIC styles include full custom (for analog) and RTL-based design: Cell based (semi-custom), Gate Array or FPGA implementation
- ASIC design methodology includes logic, timing, and physical design
  - Unfortunately, design productivity is not keeping up with chip performance growth