VLSI Design at Intel

Dr. Steve Haynal

Formerly with Strategic CAD Labs, Intel
Outline

- Marching to Make Moore's Law True
- Manufacturing
- Design

Please Ask Questions!
Pop Quiz

• Order the following in order of size (smallest first)

- Influenza A virus
- Transistor in high volume microprocessor in 2007
- Water molecule
- Grains of sand
First microprocessor (1971)
For Busicom calculator

Characteristics
- 10 \( \mu \text{m} \) process
- 2300 transistors
- 400 – 800 kHz
- 4-bit word size
- 16-pin DIP package

Masks hand cut from Rubylith
- Drawn with color pencils
- 1 metal, 1 poly (jumpers)
- Diagonal lines (!)
The First Nehalem Processor

A Modular Design for Flexibility
Moore’s Law

- In 1965, Dr. Gordon Moore, co-founder of Intel, extended a line on a lin-log paper plotting the number of transistors in each integrated circuit generation (4 at the time)
  - The slope indicated a doubling of the number of transistors every year.
  - In the article, Dr Moore expected this to go on for a few more generations and then taper off.
- In 1975, Dr. Moore revised the curve
  - New slope predicted doubling every 2 years.
  - Again Dr. Moore expected this to continue for a few more generations and then taper off.
- For the next 30 years, Dr. Moore revisited the topic regularly
  - The slope stayed constant at doubling every 18-24 months
  - He always thought it would continue a few more generations and then taper off.
Popularization of Moore’s Law:

- “The Definition of “Moore’s Law” has come to refer to almost anything related to the semiconductor industry that when plotted on semi-log paper approximates a straight line”

Dr. Gordon Moore
Plenary Talk
February 19, 1995
E-Beam, X-Ray, EUV and I-beam Lithography for Manufacturing V
Santa Clara, CA, USA
What He Actually Said...

• “I wanted to get across the idea that integrated circuits were a way to make electronics cheap. You could see the technology was going to let you make more complex things and the costs were going to go down. That was really the message I wanted to get across”

Dr. Gordon Moore
San Jose Mercury News
April 2, 2005
Moore’s Law in Practice:

Intel Microprocessors Complexity 1971 - 2006
The progress has been staggering. In 1955, the annual production of transistors could be measured in the millions. In 2003, production came to around a quintillion, or a trillion million.

In 1954, the average price of a transistor cost $5.52. In 2004, the average cost was 191 nanodollars, or 191 billionths of a dollar.
The Result
The Blessing/Curse of Moore’s Law

😊 We know where to go, what is needed and when it is needed.

"More than anything, once something like [Moore’s Law] gets established, it becomes more or less a self-fulfilling prophecy. The Semiconductor Industry Association puts out a technology road map, which continues this generation [turnover] every three years. Everyone in the industry recognizes that if you don’t stay on essentially that curve they will fall behind. So it sort of drives itself."

(Moore, 1996)

😢 We have to get to the next generation when the law says we should – no matter how hard it is...
Through the Next Decade and Beyond

Future options subject to change
Moore’s Law has been declared “about to die” regularly for the last 40 years.

[Moore] was right, more or less. But soon, Moore’s law will collide with a much less flexible set of laws — the laws of physics. Within the next decade … [optical lithography] probably won’t be dexterous enough to shrink transistors smaller than 50 nanometers.

Warning!

So far, all wrong!

…”The price per transistor will bottom out sometime between 2003 and 2005. From that point on, there will be no economic point to making transistors smaller. So Moore's Law ends in seven years.”

Forbes, March 25, 1995

…[Intel] use innovative design rules and advanced mask techniques to extend the use of 193nm dry lithography to manufacture its 45nm processors …[in which transistors are] 20nm wide

Moore’s law will continue to drive the semiconductor industry. However ... Design challenges must be overcome to fully utilize the transistor availability!
Manufacturing
In the Beginning...

- Polycrystalline silicon (sand) is made ultra-clean
- Starting from a seed crystal, a large silicon ingot (single crystal) is slowly pulled from melted silicon using the Czochralski process
  - ~300mm in diameter
  - ~1-2 meter in length
- Thin wafers (~0.75 mm) are cut (sliced)
- Wafers are polished to mirror likeness
Fabrication ICs on Wafer

- Patterning that creates shapes for processing
  - Lithography using masks and photoresist
- Modification of electrical properties
  - Diffusion and ion implantation
- Deposition of various materials
  - Physical/chemical vapor deposition, molecular beam epitaxi and/or atomic layer deposition
- Removal of unwanted material
  - Wet/dry etching, chemical-mechanical planarization
- Total number of steps to make a modern microprocessor IC: ~350
Photolithography

- Oxidation
- Photoresist (PR) coating
- Exposure
- Photoresist development and bake
- Acid etching
  - Unexposed (negative PR)
  - Exposed (positive PR)
- Spin, rinse, and dry
Mask Making

• Start with extremely clean and smooth glass plate
• Deposit a layer of chrome
• Using a laser or e-beam to draw (by removing) the desired pattern
• Use a stepper to expose the mask at every location.
  ➢ Extreme position accuracy is needed!
• This sounds fairly simple. HOWEVER…
What Happens in Reality?

- Illuminate mask with 193nm wavelength light (UV)
- Consider the result for various sizes of patterns
- The feature sizes are much smaller than the wavelength of the light $\rightarrow$ diffraction destroys the pattern

Source: Synopsys Inc.
What is going on?

- Assume we use 193nm coherent (laser) light
- Assume a simplistic photo-resist model
  - Simple threshold model
- If we plot the intensity and resist response for a single small opening in the mask for a typical manufacturing lens configuration, we get:
What is going on? Part 2

• With two holes we get:
In More Detail

- Plotting the electrical field as well:
Potential Solutions

- Reduce the wavelength of the light
  - Extreme-UV (13.5nm) is being explored using reflective optics and reflective masks
    - Incredibly complex and expensive process
- Change what you try to pattern
  - Make a mask that *after exposure* yields the desired pattern on silicon wafer
  - Some approaches:
    - Optical proximity correction
    - Phase shifting masks
    - Immersion lithography
Using “Optical Tricks”

- With two holes, but using opposite phase, we get:
Complex example:

If we want the pattern:

we can use the mask:

which yields:

30nm
Manufacturing Challenges

- Enable Moore's law with new engineering feats
- Design rule complexity
- Variation
- Low power
- Fab costs
Design
The Design Process at 10,000 ft

Architecture Analysis
Development of micro-architecture
Mapping of RTL to transistors
Development of mask that yield transistors
Making Silicon + Stepping(s)

Original Product Target

Chip

Validation

MAS: Micro-Architecture Specification
RTL: Register-Transfer Language

This is the theory...
In Practice...

- Original Product:
  - ~2-3 years

- Target Repainted to fit Reality:
  - ~1 year

- Roles:
  - Architect
  - Micro-Architect
  - Design Engineer
  - Mask Designer
  - Test Engineer
  - ~1 year
Effort over Time

First P4 Architecture Development Effort
Architecture Analysis

Example: Single-core or dual-core microprocessor?
Processor Architecture 101

Delivered Performance ~
Instructions per cycle (IPC) * Frequency

Goal is higher performance and lower power

Power consumption ~
\[ C_{\text{dynamic}} \times V \times V \times \text{Frequency} \]

\( C_{\text{dynamic}} \) is roughly a product of area and activity:
area ~ “how many transistors” * “how big transistors”
activity ~ “how often do they change from on to off and back”

V = voltage
Processor Architecture 101

Delivered Performance ~
Instructions per cycle (IPC) * Frequency

Power consumption ~
$C_{\text{dynamic}} \cdot V \cdot V \cdot \text{Frequency}$

+ 

For Silicon: Frequency ~ V
Conclusion: Dual-core gives better performance and performance/watt!
Micro-Architecture Development

Examples: Floor planning & Accumulator design

Architecture Analysis
Development of micro-architecture
MAS
RTL
Schematics
Mapping of RTL to transistors
Layout/Mask
Development of mask that yield transistors and wires
Making Silicon + Stepping(s)
Original Product Target
Chip
Validation

Examples: Floor planning & Accumulator design
Floorplanning & $\mu$-Architecture

Function Decomposition Architectural Breakdown

Flopped Repeater

Layout Floorplan #1

Pick a frequency $F$

Piped interconnect distance 1 Tick

Layout Floorplan #2

A-align
B-add
C-normalize
D-round
Accumulator Design

• How to design a fast, but power-efficient, accumulator supporting back-to-back additions?
  ➢ The circuit reads a 64-bit quantity every clock cycle
  ➢ The circuit adds the input to a running sum
  ➢ The output of the sum is delayed by one clock cycle
Alternative 1

• Pros:
  - Simple to model
  - Output arrives very early in the clock cycle

• Cons:
  - 64 bit adder is very difficult to make fast and power-efficient
  - Inputs are needed very early in the clock cycle
Alternative 2

• Pros:
  • Simple to model
  • Input can arrive very late in the clock cycle
  • Output arrives very early in the clock cycle

• Cons:
  • 64 bit adder is very difficult to make fast and power-efficient
Alternative 3: Split & Stagger Add

• Pros:
  ➢ Only need to create a 32-bit addition in one cycle (plenty of time->low power)
  ➢ Most significant inputs can arrive late in the clock cycle
  ➢ Outputs arrive early in the clock cycle

• Cons:
  ➢ Least significant inputs are needed fairly early in the clock cycle.
Mapping to Transistors

Example: How to design a 32-bit adder?
Alternative 1: Serial Adder

- **Pros:**
  - Very small
  - Very power efficient

- **Cons:**
  - Extremely slow

700ps
720μm²
Alternative 2: Kogge-Stone Adder

- **Pros:**
  - Minimum logic depth
  - Very fast

- **Cons:**
  - Very large
  - Very power hungry
  - Difficult to route (many wires)

- Timing:
  - 120ps
  - 2900µm²

- Timing:
  - 200ps
  - 2700µm²
Alternative 3: Sklansky Adder

- **Pros:**
  - Minimum logic depth
  - Fairly power efficient if not pushed for speed

- **Cons:**
  - High fanout of some gates
  - Large, if pushed for speed

<table>
<thead>
<tr>
<th>Delay</th>
<th>Area</th>
</tr>
</thead>
<tbody>
<tr>
<td>225ps</td>
<td>1270μm²</td>
</tr>
<tr>
<td>150ps</td>
<td>2100μm²</td>
</tr>
</tbody>
</table>
Layout Creation

Example: How to make the mask for a CMOS inverter?
Simplified CMOS Inverter
P-Well Mask
Active Mask
Poly Mask
P+ Select Mask
N+ Select Mask
Contact Mask
Metal Mask
The Challenge in Mask Design

- “Design Rules” are used to ensure only well behaved transistors/wires/… are produced.
- Design rules have been added to ensure reliability and/or “manufacturability”
- As feature sizes have become smaller and smaller, the number and complexity of the design rules have increased enormously
- Checking for violation (DRC) is now extremely time consuming
Making of Silicon + Testing

How to get, test and check first/second/... silicon?

Architecture Analysis
Development of micro-
architecture
Mapping of RTL to
transistors
Development of mask
that yield
transistors
and wires
Making Silicon +
Stepping(s)
Chip
Original
Product
Target

Validation
Making of Silicon + Testing

- The layout masks (the desired masks) are sent through the mask-generation program that computes the needed masks to get the desired results
  - Extremely compute intensive
  - Extremely large amounts of data
- Once chips are manufactured, testing is started
  - First ½ hour on tester yields more cycle run than total number of cycles simulated pre-Silicon!
  - Testing is often done without heat sink so we use a lot of liquid Nitrogen!
Validation

How to: 1) check we captured what we wanted
2) check that we did not make a mistake along the way
What Needs to be Validated?

- Functionality
- Performance
- Power & Thermal
- Physical form
- Documentation
- Reliability
- Testing procedure
- …
## Coverage

<table>
<thead>
<tr>
<th></th>
<th>Pro</th>
<th>Con</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Formal Verification</strong></td>
<td>• 100% coverage</td>
<td>• Requires special skills</td>
</tr>
<tr>
<td></td>
<td>• Proves absence of bugs</td>
<td>• Constrained by complexity</td>
</tr>
<tr>
<td><strong>Directed Random Tests</strong></td>
<td>• Targets areas most likely to be of concern</td>
<td>• Requires strong uArch knowledge</td>
</tr>
<tr>
<td></td>
<td>• Greatly reduces cycle requirements</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Develops strong uArch knowledge</td>
<td></td>
</tr>
<tr>
<td><strong>Generic Random Tests</strong></td>
<td>• After generator created, easy to write</td>
<td>• Requires almost $\infty$ cycles / time</td>
</tr>
<tr>
<td></td>
<td>• Requires little uArch knowledge</td>
<td>• Difficult / impossible to avoid broken features</td>
</tr>
<tr>
<td></td>
<td>• Can create things no one would ever think of</td>
<td></td>
</tr>
<tr>
<td><strong>Directed Tests</strong></td>
<td>• Easy to write</td>
<td>• Requires almost $\infty$ number of tests</td>
</tr>
<tr>
<td></td>
<td>• Easy to understand</td>
<td>• Difficult to hit uArch conditions</td>
</tr>
<tr>
<td></td>
<td>• Easy to reuse</td>
<td></td>
</tr>
</tbody>
</table>

100% Covered

Low % Covered
Design Productivity Trends

Complexity outpaces design productivity

Courtesy, ITRS Roadmap
Design Abstraction Levels
Design Challenge: Validation

- The designs are getting increasingly complex
- The real logic definition is a very low-level description of the design and is constantly changing
- Too many design models that need to be verified against each other
- Today 1/3-1/2 of design team is devoted to validation
  - for ASICs >50% validation!
Design Crisis

- **Complexity of design**
  - More transistors →
  - More functionality →
  - More design effort

- **Number & size of models**
  - Performance, ERTL, GRTL, Schematics, ...
  - Multi-million line RTL

- **Validation of design**
  - Bug rate rising 4x per lead
  - Trillions of simulation cycles on a rapidly changing model

- **Multi-objective convergence**
  - Timing, power, area, etc. feedback way too late in design schedules

Bottom line: Existing design approaches inadequate for the design of future processors/chips
Mind the Gap!

**Increased attention to details of physical & manufacturing realities**

**Opposing Forces driving System Design**

**Ever higher abstraction levels needed to capture system functionality**
Answer to Pop Quiz

• Order the following in order of size (smallest first)

1. Water molecule ~1nm
2. Transistor in high volume microprocessor in 2007 ~30nm
3. Influenza A virus ~100nm
4. Grains of sand ~100,000nm
Questions