CS250
VLSI Systems Design

Lecture 3: Technology Introduction

Spring 2016

John Wawrzynek

with

Chris Yarp (TA)

Thanks to John Lazaro for lots of slides
Silicon “ingots” are grown from a “perfect” crystal seed in a melt, and then purified to “nine nines”.
Ingots sliced into 450\(\mu\)m thick wafers, using a diamond saw.
An n-channel MOS transistor (planar)

**Vd = 1V**

- **Vg = 0V**
  - $I \approx nA$
  - polysilicon gate, dielectric, and substrate form a capacitor.
  - nFet is off (I is "leakage")

**Vd = 1V**

- **Vg = 1V**
  - $I \approx \mu A$
  - Vg = 1V, small region near the surface turns from p-type to n-type.
  - nFet is on.
Mask set for an n-Fet (circa 1986)

$V_d = 1\text{V}$  
$V_g = 0\text{V}$  
$V_s = 0\text{V}$

$\downarrow I \approx nA$

Top-down view:

- #1: n+ diffusion
- #2: poly (gate)
- #3: diff contact
- #4: metal

Layers to do
p-Fet not shown.
Modern processes have 6 to 10 metal layers (or more) (in 1986: 2).
“Design rules” for masks, 1986 ...

- Poly overhang. So that if masks are misaligned, we still get channel.
- Minimum gate length. So that the source and drain depletion regions do not meet!
- Metal rules: Contact separation from channel, one fixed contact size, overlap rules with metal, etc ...

#1: n+ diffusion
#2: poly (gate)
#3: diff contact
#4: metal
How a fab uses a mask set to make an IC

Vd = 1V

Vg = 1V

Vs = 0V

Top-down view:

#1: n+ diffusion
#2: poly (gate)
#3: diff contact
#4: metal

Masks

I ≈ μA
Start with an un-doped wafer ...

**Steps**

#1: dope wafer p-

#2: grow gate oxide

#3: deposit polysilicon

#4: spin on photoresist

#5: place positive poly mask and expose with UV.

UV hardens exposed resist. A wafer wash leaves only hard resist.
Wet etch to remove unmasked ...

HF acid etches through poly and oxide, but not hardened resist.

After etch and resist removal
Use diffusion mask to implant n-type

accelerated donor atoms

Notice how donor atoms are blocked by gate and do not enter channel.

Thus, the channel is “self-aligned”, precise mask alignment is not needed!
Metallization completes device

Grow a thick oxide on top of the wafer.

Mask and etch to make contact holes.

Put a layer of metal on chip. Be sure to fill in the holes!
Final product...

"The planar process"
Jean Hoerni, Fairchild Semiconductor 1958

Top-down view:
Gordon Moore
UCB B.S.
Chemistry, 1950.
MOS in the 70s

1971 state of the art.

Intel 2102, a 1kb, 1 MHz static RAM chip with 6000 nFETs transistors in a 10 μm process, like the one we just saw.
By 1971, “Moore’s Law” paper was already 6 years old ...

But the result was empirical.

Understanding the physics of scaling MOS transistor dimensions was necessary ...

Original “Moore’s Law” paper data points.
If we scale the gate length by a factor $\kappa$, how should we scale other aspects of transistor to get the “best” results?

Fig. 1. Illustration of device scaling principles with $\kappa = 5$. (a) Conventional commercially available device structure. (b) Scaled-down device structure.
Dennard Scaling

Things we do: scale dimensions, doping, Vdd.

What we get: $\kappa^2$ as many transistors at the same power density!

Whose gates switch $\kappa$ times faster!

Power density scaling ended in 2003 (Pentium 4: 3.2GHz, 82W, 55M FETs).

TABLE I

<table>
<thead>
<tr>
<th>Device or Circuit Parameter</th>
<th>Scaling Factor</th>
</tr>
</thead>
<tbody>
<tr>
<td>Device dimension $t_{ox}$, $L$, $W$</td>
<td>$1/\kappa$</td>
</tr>
<tr>
<td>Doping concentration $N_a$</td>
<td>$\kappa$</td>
</tr>
<tr>
<td>Voltage $V$</td>
<td>$1/\kappa$</td>
</tr>
<tr>
<td>Current $I$</td>
<td>$1/\kappa$</td>
</tr>
<tr>
<td>Capacitance $\epsilon A/t$</td>
<td>$1/\kappa$</td>
</tr>
<tr>
<td>Delay time/circuit $VC/I$</td>
<td>$1/\kappa$</td>
</tr>
<tr>
<td>Power dissipation/circuit $VI$</td>
<td>$1/\kappa^2$</td>
</tr>
<tr>
<td>Power density $VI/A$</td>
<td>1</td>
</tr>
</tbody>
</table>
Figure 1.11 Growth in clock rate of microprocessors in Figure 1.1.

Between 1978 and 1986, the clock rate improved less than 15% per year while performance improved by 25% per year. During the "renaissance period" of 52% performance improvement per year between 1986 and 2003, clock rates shot up almost 40% per year. Since then, the clock rate has been nearly flat, growing at less than 1% per year, while single processor performance improved at less than 22% per year.

Dennard Scaling ended ... when we hit the "power wall"
Moore’s Law

We still scale to get more transistors per unit area ... but we use design techniques to reduce power.
Bulk versus SIO Processing

‣ “Silicon on Insulator”

‣ Lower parasitic capacitance -> lower energy, higher-performance

‣ Also used for “radiation hard” application (space craft) - sapphire instead of Oxide.

‣ 10 - 15% increase in total manufacturing cost due to substrate cost.
Current state-of-the-art photolithography tools use deep ultraviolet (DUV) light with wavelengths of 248 and 193 nm, which allow minimum feature sizes below 50 nm.

Optical proximity correction (OPC) is an enhancement technique commonly used to compensate for image errors due to diffraction or process effects.
Latest Modern Process

Transistor channel is a raised fin.
Gate controls channel from sides and top.

Intel 22nm Process

United States Patent

Hu et al. Filed: Oct. 23, 2000

FINFET TRANSISTOR STRUCTURES HAVING A DOUBLE GATE CHANNEL EXTENDING VERTICALLY FROM A SUBSTRATE AND METHODS OF MANUFACTURE

Inventors: Chenming Hu, Alamo; Tsu-Jae King, Fremont; Vivek Subramanian, Redwood City; Leland Chang, Berkeley; Xuejue Huang; Yang-Kyu Choi, both of Albany; Jakub Tadeusz Kedzierski, Hayward; Nick Lindert, Berkeley; Jeffrey Bokor, Oakland, all of CA (US); Wen-Chin Lee, Beaverton, OR (US)
When will it end?*

难度

14nm

On 5 September 2014, Intel launched the first three Broadwell-based processors that belonged to the low-TDP Core M family, Core M 5Y10, Core M 5Y10a and Core M 5Y70.[19]

In February 2015, Samsung announced its flagship smartphones Galaxy S6 and Galaxy S6 Edge would feature 14 nm Exynos systems-on-a-chip.[20]

On March 9, 2015, Apple Inc. released the "Early 2015" MacBook and MacBook Pro, which utilized 14 nm Intel processors. Of note is the i7-5557U, which has Intel Iris 6100 graphics and two cores running at 3.1Ghz, using only 28 watts.[21][22]

On September 25, 2015, Apple Inc. released iPhone 6s and iPhone 6s Plus, which are equipped with "desktop-class" A9 chips[23] that are fabricated in both 14 nm by Samsung and 16 nm by TSMC.

10nm

On 23 May 2015, Samsung Electronics showed off a 300 mm wafer of 10 nm FinFET chips.[15]

7nm

Although Intel has not yet divulged any certain plans to manufacturers or retailers, it has already stated that it would no longer use silicon at this node.[7] A possible replacement material for silicon would be indium gallium arsenide (InGaAs).[8]

In April 2015, TSMC announced that 10 nm production would begin in 2016, followed by 7 nm production in 2017.[9]

* From Wikipedia
Processing Enhancements

- **Trench isolation**: Shallow trench isolation (STI), a.k.a. Box Isolation Technique, prevents current leakage between n-well and p-well devices.

- **High-K dielectrics / Metal gate**: Replacing the silicon dioxide gate dielectric with a high-$\kappa$ material allows increased gate capacitance without the concomitant leakage effects.

- **Strained Silicon**: A layer of silicon in which the silicon atoms are stretched beyond their normal interatomic distance leading to better mobility, resulting in better chip performance and lower energy consumption.

- “Gate Engineering”: for within-die choice of multiple transistor threshold voltages ($V_t$) to optimize delay or power.
Photo: Global Foundries fab floor, before equipment arrives.
Design Styles

Chip Designers: Head in the Clouds

Design Kit: Design Rules, Device Models, Standard layouts

IC Process Designers: Feet on the Ground
Structured Custom Design

※ Wiring by abutment. Rectangular leaf cell layout is hand-crafted so that edge wires “match up” when cells are tiled in 1-D or 2-D.

※ Cell compilation. Designers write programs (“cell compilers”) to tile leaf cells into larger logic blocks. Wire routing comes “for free”.

※ Parameters. N-bit datapath compilers.
RAM Compilers

On average, 30% of a modern logic chip is SRAM, which is generated by RAM compilers.

Compile-time parameters set number of bits, aspect ratio, ports, etc.
“Structured Custom” CPU (David Johannsen, Caltech, ’77)
Limitations

Labor intensive.

Key metric is the number of leaf cells required to efficiently use a given area of silicon.

Memory arrays and FPGAs are a good fit.

Still, even today it is common to see custom layout in critical parts of CPU logic.
Standard Cell Design

Process Design Kit: Design Rules and Device Models

Logic schematics using library gates.

Gate Library. Fixed-height, to be placed in rows. Vdd and Gnd rails connect by abutment.

I/O Ports. Auto-router places wires over cell to connect them.
Place & Route

Router connects gate ports to match schematic.

Router "optimizes" relative lengths of wire to meet constraints.

Software places cells into rows, to "optimize" area, performance, and power constraints.

We put "optimize" in quotes to reflect the NP-hard nature of the algorithms behind place & route.
Benchmark by a custom design house (Obsidian).

In general, they claim: “30% of the power, twice the speed, and 4 times the density of standard cells”.

Custom layout (left) is a factor of 2.2 smaller than standard cell layout (right).
At the start of the 1980s, the standard-cell flow was driven by hand-drawn schematics.

By the early 1990s, schematics were replaced with Verilog/VHDL, to drive logic synthesis, whose output was integrated into standard cell back ends.
Logic design
... as I learned it in 1981 ...
In the early 1980s, progress in academia and industrial labs made the problem domain tractable ...

**Given:** Finite-State Machine $F(X,Y,Z, \lambda, \delta)$ where:
- $X$: Input alphabet
- $Y$: Output alphabet
- $Z$: Set of internal states
- $\lambda: X \times Z \rightarrow Z$ (next state function)
- $\delta: X \times Z \rightarrow Y$ (output function)

**Target:** Circuit $C(G, W)$ where:
- $G$: set of circuit components $g \in \{\text{Boolean gates, flip-flops, etc}\}$
- $W$: set of wires connecting $G$
In the second half of the 1980s, the startup that became Synopsys developed Design Compiler (dc) ... Eventually, Verilog/VHDL.

1986 “Pitch Slide” for EDA startup Optimal Solutions

Productivity the key “value add”

Technology Mapping
Modern ASIC Methodology and Flow

- RTL Synthesis Based

HDL specifies design as combinational logic + state elements

Cell instantiations needed for blocks not inferred by synthesis (typically RAM)

Event simulation verifies RTL

“Formal” verification compares logical structure of gate netlist to RTL

Place & route generates layout

Timing and power checked statically

Layout verified with LVS and GDRC

Diagram:

1. Specification
2. RTL (Verilog/VHDL) + cell instantiations
3. "formal" verification
4. Logic synthesis
5. Event simulator
6. Gate netlist (with area/perf/pwr estimates)
7. Cell place & route
8. GDS
9. Timing/power analysis
10. GDRGC, LVS, other checks
Systems on a Chip (SoCs)

Today’s chips are mosaics. The chip design process often consists of licensing “intellectual property (IP)” from other companies (large like CPUs and GPUs, small like DRAM controllers & analog blocks).

On chip buses. IP blocks are often designed to hook up to standardized on-chip buses, defined by CPU IP vendors like ARM.

Process Design Kit: Design Rules and Device Models
Chip designed by Apple, but many blocks are licensed from third parties. Some are standard cells, others full custom.
On-chip bus hierarchy for an ARM-based system...
The Programmable Imperative

Instead of doing your own chip, buy a standard-product chip that is programmable in ways more sophisticated than a PC. Examples: FPGAs (Field Programmable Gate Arrays), specialized CPU-based chips.

Build or Buy? “Buy” wins at lower volumes. Cross-over shifting rightward over time.

Traditional FPGA versus ASIC argument (circa 2000)

• **ASIC:** High NRE costs ($2M for 0.35um chip). Relatively Low cost per die.

• **FPGAs:** Very low NRE costs. Relatively low silicon efficiency ⇒ high cost per part.

• Cross-over volume from cost effective FPGA design to ASIC in the 10K range.
Cross-over Point has Moved Right

- **ASIC**: Increasing NRE costs (verification, mask costs, etc.)
  - Fewer silicon designs becomes inevitable.
- **FPGAs**: Move in to fill the need, furthermore, FPGAs better able to follow Moore’s Law, relatively cheaper to test.
- Cross-over volume now >100K.
Xilinx ZYNQ

A dual-core ARM SoC with a full set of peripherals.

Plus, a significant portion of the chip area devoted to Xilinx FPGA elements, that interact with ARM cores efficiently.
Gate Array

- Prefabricated wafers of "active" & gate layers & local interconnect, comprising, primarily, rows of transistors. Customize as needed with "back-end" metal processing (contact cuts, metal wires). Could use a different factory.

- CAD software understands how to make gates and registers.

![Diagram of two-step manufacture process for Gate Array.](image-url)
Gate Array

- Shifts large portion of design and mask NRE to vendor.
- Shorter design and processing times, reduced time to market for user.
- Highly structured layout with fixed size transistors leads to large sub-circuits (ex: Flip-flops) and higher per die costs.
- Memory arrays are particularly inefficient, so often prefabricated, also:

Sea-of-gates, structured ASIC, master-slice.
• Gate Array like devices (structured ASICs) could return to fill the gap?
• Post-fab customization with limited mask layers.
  ❯ Lower NREs than ASICs, more silicon efficiency than FPGAs.
Summary: So what *has* changed in 30 years?
Processing advances

4µm

45nm
IC Technology Stuff (1)

- **Feature size:**
  - *then:* \( \sim 4 \mu m \)  
  - *now:* .014\( \mu m \)  
  - *moving to:* .010\( \mu m \)

- **Interconnect:**
  - *then:* 2 layers  
  - *now:* \( \sim 10 \) layers  
  - *then:* aluminum  
  - *now:* copper

- **Transistors:**
  - *then:* planar MOSFET  
  - *now:* same + fin-fets

- **Layout and GDRs:**

- **Circuits:**
  - *then:* clocked static CMOS  
  - *now:* same (lots of crazy stuff in between)

Interesting, though, most CMOS circuits and layouts designed in 1980 would work if fabricated on today’s IC process.
IC Technology Stuff (2)

- **Transistors:**
  - *then:* near perfect switch  
  - *now:* leaky

- **Power consumption:**
  - *then:* dynamic (switching) energy  
  - *now:* approaching 50% static leakage (back to the future - nMOS has similar problem)

- **New improved devices coming soon:** FinFETs

- **Chip Input/Output**
  - *then:* parameter pads  
  - *now:* often area pads

- **Lithographic Mask Costs:**
  - *then:* few $k  
  - *now:* $M (full die, 45, 28, 14nm)
IC Technology Stuff (3)

- **Device reliability:**
  
  _then:_ devices nearly never fail  _future_ (<65nm): high soft and hard error rates

- **Process variations across die, die-to-die:**
  - Statistical variations in processing (wire widths/resistivity, transistor dimensions/strengths, doping inconsistencies) become apparent at smaller geometries.
  - Some circuits fast, others slow. Some high-power, some low.

- **Yield on leading edge processes dropping dramatically**
  - IBM quoted yields of 10 – 20% on Cell processor
Design Stuff

- **Chip functionality:**
  
  *then:* limited by area  *now:* usually limited by energy dissipation

- **Design cost:**
  
  *now:* design costs in +$50M range for full-die custom designs (high percentage in verification)

- **Implementation Alternatives:** more alternatives that trade up-front design costs for per unit costs.

- **FPGA compete aggressively with custom silicon**
  
  *then:* most custom designs implemented at silicon level
  
  *now:* many more custom designs implemented with FPGAs

- **Standard design abstraction:**
  
  *then:* transistors circuits  *now:* RTL in HDLs, standard “cores” and standard cells (higher productivity, somewhat less area/energy efficient) - High-level Synthesis (HLS) on its way.
## Implementation Alternatives

<table>
<thead>
<tr>
<th>Implementation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Full-custom:</td>
<td>All circuits/transistors layouts optimized for application.</td>
</tr>
<tr>
<td>Standard-cell:</td>
<td>Arrays of small function blocks (gates, FFs) automatically placed and routed.</td>
</tr>
<tr>
<td>Gate-array (structured ASIC):</td>
<td>Partially prefabricated wafers customized with metal layers or vias.</td>
</tr>
<tr>
<td>FPGA:</td>
<td>Prefabricated chips customized with loadable latches or fuses.</td>
</tr>
<tr>
<td>Microprocessor:</td>
<td>Instruction set interpreter customized through software.</td>
</tr>
<tr>
<td>Domain Specific Processor:</td>
<td>Special instruction set interpreters (ex: DSP, NP, GPU).</td>
</tr>
</tbody>
</table>

By “ASIC”, most people mean “Standard-cell” based implementation.

**What are the important metrics of comparison?**
The Important Distinction

• Instruction Binding Time
  ▶ When do we decide what operation needs to be performed?

• General Principles

  Earlier the decision is bound, the less area, delay/energy required for the implementation.
  Later the decision is bound, the more flexible the device.
Engineering Challenge

Application

Physics
CS250 Design Refinement

- Accelerator Algorithm (spec/simulator)
- Micro-Architecture Design (Manual)
- Detailed micro-arch design (Manual)
- RTL (Chisel)
- Synthesis (automated) + Instantiation
- Gate netlist (Stdcell Library)
- Place and Route (automated)
- Layout (Stdcell Library)

- Register File
- AL
End of Introduction
part 2