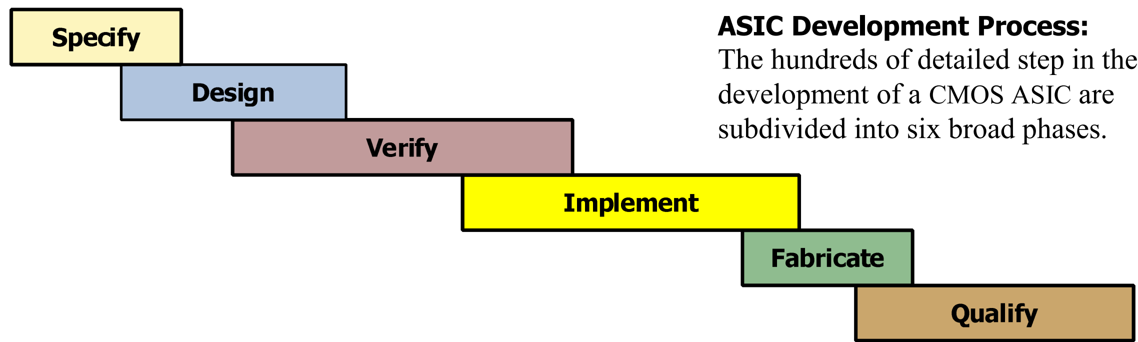
Outline: Nanometer ASIC Seminar



Length: Two Full Days (say, 10am—6pm)
Speaker: Charles Dancak [Dan-chek], UC Extension in Silicon Valley, USA
Labs: None

Scope of Seminar:

The purpose of this seminar is to explain the steps involved in developing an ASIC, at nanometer feature sizes (32 nm and below).The flow described is *tool-neutral*. The speaker relies on his wide background in front-end design, design-for-test, MOS fabrication, and solid-state physics to explain key concepts at an intuitive level, to connect the dots, demystify common acronyms, and impart a solid understanding of how today's SOCs are designed, verified, laid out, and—following tape-out—fabricated and packaged. The seminar includes insights into how EDA tools (such as Synopsys ***Design Compiler***) really work.

Phase 1: Specification

The audience is first introduced to a few basic ASIC elements, ensuringeveryone is on the same page. As a lead-in, the story of Jack Kilby, and his idea for the first integrated circuit, is told. Soon the audience understands how MOS *transistors* act as tiny on/off switches, come in two complementary types, their speed ultimately limited by *channel length,* and power usage dramatically reduced. Additional topics are:

- Switching action of a NAND gate, a NOR gate, and layout of a synthesis-friendly standard cell.
- Specifying a CMOS chip as a *product on paper:* feature grid; feasibility study; die costing.
- The fabless ASIC ecosystem; processing options; shuttle wafers; other key planning details.
- Running case study: Broadcom 1103 CMOS Ethernet IP phone chip.

Phase 2: Hands-On Design

Based on the spec, design activity now begins for the digital and on-chip analog blocks on IC:

- Case study: writing RTL code for a simple digital function (a serializer) in SystemVerilog.
- How ASIC synthesis tools transform an RTL description into optmized standard-cell circuits.
- Computing gate-level delays from standard-cell library data and estimated net lengths.
- Synthesizing the code into technology-specific logic. Applying a clock-period constraint.
- Single-cycle timing: Does new data traverse the critical path within one clock period?
- Writing RTL assertions, like executable comments that are checked by simulators or FEC.

Subtopic: IP-Based Design

Buying or building IP cores, getting the right type and version, and interconnecting the blocks with standardized buses is a primary activity in SOC design. This subtopic includes:

- Core formats: soft, firm, and hard IP. How does format affect pricing?
- Core types: CPU, DSP, GPU, RPU, encryption, digital interfaces, analog.
- Classes of CPU core: low-end for MCU and IoT, midrange for embedded performance efficiency, high end. Multi-core clusters; homogeneous and heterogeneous computing.
- On-chip busing and networks. Using SystemVerilog's **`interface`** construct to encapsulate busing fabric with concise and reusable coding.

Subtopic: RF/Analog Design

- Generic design flow for on-chip RF/wireless, sufficiently insensitive to process variations.
- Case study: Voltage-controlled oscillator (VCO) design, through GDSII layout.
- Write Verilog-AMS model for VCO to catch bugs like wrong polarity or crossed signals.

Phase 3: Functional Verification

This phase verifies that the RTL source code for the digital blocks on the ASIC accurately reflects the original functionality in the spec. Simulation is supplemented by FEC and by emulation.

- Testbench evolution from simple block-level, to layered system-level, testbenches using transactions, constrained random stimuli (CRV), embedded SystemVerilog assertions, functional-coverage metrics.
- Formal equivalence checking (FEC) to confirm—without need of vectors applied over lengthy times —that functionality does not change after scan insertion, clock gating, and change orders (ECOs).
- Hardware emulation: building an FPGA-based hardware model—slower-running than the ASIC, but ideal for early debug (especially hardware-firmware issues) of the yet-to-be-fabricated chip design.

Phase 4: Implementation

During this phase, activity crosses over from the logical to the physical domain. The synthesized gate-level netlist is mapped onto a rectangular die, and I/O pads are placed.Closure must be met on timing goals, power consumption, cell area, and chip testability. Steps covered include:

- Final synthesis of the fully-verified RTL source code; meeting budgets for timing, area, etc.
- How clock signals are handled during logic synthesis and physical design. Clock-domain crossing.
- What is a timing violation? By what methods is it fixed? How are gate and net delays computed?
- Physical design steps: floorplanning;; placement of standard cells; clock-tree synthesis; routing.
- Readying a die for fabrication: power-grid layout; fixing congestion or excess IR drop; DRC rules.

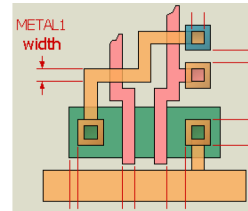
Subtopic: Design for Test

DFT closure requires an ASIC that's highly testable, often at 98% coverage. Subtopics include:

- Inserting scan chains, and generating test patterns for stuck-at, bridging, and delay faults.
- Built-in self-test for memory blocks (MBIST), and for logic blocks (LBIST and STUMPS).
- Inserting boundary-scan (JTAG) logic around periphery of a chip, for board-level testing.

Subtopic: Tape-Out

Use physical verification tools to confirm that the post-routed layout of the entire chip meets timing constraints, has no signal-integrity violations, and satisfies design rules. Includes:



- What causes signal-integrity issues, like crosstalk-induced delays?
- What are geometric (DRC) and electrical (ERC) rules? Why custom rules?
- Layout vs. schematic (LVS) checking, to verify layout matches schematics.
- Streaming a GDSII file, a full-chip geometric description of routed layout.
- Uploading GDSII file to a mask shop. Preparing for mask generation (OPC).

Phase 5: Fabrication

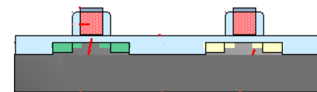
After tape-out, the huge chip-design database is sent to the mask shop and wafer fab. Using masks, the design is imprinted onto the surface of blank wafers. How are line widths of 32 nm and below achieved? Challenges due to short-channel effects of MOS transistor. The nonplanar FinFET transistor is explained.

- How a GDSII database is used to make a set of accurate masks or reticles. What limits the resolution?
- Key fabrication techniques: photolithography; oxidation; ion implant; metallization; IMD, CMP.
- How innovative developments like *double patterning* enable the continuation of Moore's Law.

Subtopic: FEOL Fabrication

- Shallow-trench isolation—the secret to economical CMOS wafer processing.
- Forming a polysilicon gate (including HKMG) and implanting source/drain.
- Fabricating complementary *n*- and *p*-type transistors—the basis of an inverter.

Subtopic: BEOL Fabrication



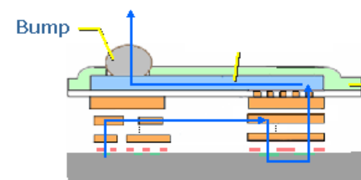
- How METAL1 layer is patterned via copper damascene processing.
- Insulating each metal layer from the next (IMD), cutting contacts/vias.
- Bumping of wafers intended for flip-chip packaging. Older wire-bond.
- Wafer sort: testing to identify known-good die (KGD) for packaging.

Comment: "The silicon fabrication topic alone was worth the price...now I know what those guys in white lab coats actually deal with!"

Phase 6: Qualification

This phase explains how good die are packaged using flip-chip or wire-bond, tested, validated at rated speed, and qualified for use in the customer's system, over a product's lifetime. Includes:

Subtopic: Package Design and Assembly



- Package codesign: electrical; thermal; mechanical.
- Flip-chip technology vs. wire-bonding.
- Signal flow off-chip, through metal layers, RDL layer; bumps, and out to the PC board. Simulating the effects of packaging.
- Silicon vias, and 3-D packaging technology.

Subtopic: Testing and Characterization

- Inside the ATE: How a test vector detects one stuck-at fault out of millions on an IC.
- Shmoo plots; analyzing chip operation at process corners (e.g. slow-slow, fast-fast).
- Functional testing over a range of frequencies; measuring f_{max} and static I_{ddq}.

Subtopic: Silicon Validation

The last chance to prevent bugs from reaching the customer base, validation confirms that the manufactured chip operates correctly—at-speed and in-system—over a range of conditions:

- Design validation board, with custom (FPGA) hardware to exercise the device at speed.
- Use logic analyzers, scan-chain dumps, JTAG ports, and often on-chip instrumentation.
- Repair issues, often using metal-mask fixes based on spare gates or flops.

Speaker's Bio:

Charles Dancak holds an MSEE from the University of Wisconsin (Madison), and also an MS in Solid-State Physics from NY Polytechnic Institute (Brooklyn). A Staff Engineer at Synopsys for ten years, he has taught workshops in eight countries outside the US. He has worked at Teradyne, Cadence, and Silicon Compilers. His very first job out of school was in MOS process technology at Intel (where, as a junior engineer, he once sat down to lobster dinner next to Gordon Moore at a team function to celebrate an E^2PROM product milestone). He developed the earliest version of this seminar for Broadcom, and taught it at two dozen sites. He now teaches ASIC-related courses—including a hands-on version of this seminar—at UC Santa Cruz Extension in Silicon Valley. He recently published a SystemVerilog paper entitled: ***Preponed Timing in RTL Assertions***.

Comment: "The systematic presentation enabled me to understand chip-related acronyms and concepts that puzzled me for years."