

Overview of Design Methodologies

--- A Review

Lecture 1

ECE 156B

1

What is a design flow

- A design flow is a flow chart
 - Each component in this flow chart is performing a specific design task, such as logic synthesis
- A design flow is determined by the **Electronic Design Automation** (EDA) software tools you purchased
 - Or based on some in-house (software) tools
- In a design flow, you may have >50 such tasks to perform

ECE 156A

2

Design flow depends on style

- Not everyone follows the same flow
 - No exactly same flow for two companies
- Different design styles result in different flows
- An FPGA-style is very different from ASIC-style, and very different from processor-style
- Every design process is unique in some sense
 - Flow is proprietary to a company

ECE 156A

3

Examples

- For fastest time-to-market, rapid prototyping
 - Use FPGA
- For fast time-to-market, low cost ASIC
 - Use commercial standard EDA design flow
 - Make everything as automatic as possible
- For high-performance, high volume design
 - No high-performance and low volume (why?)
 - Use custom design flow, optimize in every step

ECE 156A

4

Design Flow

- Derive design specification
 - Can be vague and inaccurate
- Construct simulation model
 - Behavior and/or RTL
 - Construct *assertions*
- Functional verification of simulation model
 - Subject to discuss next week
- Derive schematics
 - Synthesis (involve logic minimization), or
 - 2-level, multi-level, sequential, behavior
 - Custom design high performance components
 - Learn this from your circuit design classes

ECE 156B

5

Design Flow

- Equivalence checking (RTL Vs. Schematics)
 - Pure combinational check (block by block)
 - Use SAT and BDD
- Derive gate-level test model
 - Test based on some fault models
 - Design for test and test generation
- Placement and routing
- Clock tree design and routing
- Performance validation
 - Timing analysis, power analysis, physical design rule check

ECE 156B

6

Post-silicon stage

- Setup manufacture line
 - Test chip, yield learning
 - Either done by foundry or before/during design stage
- Obtain test silicon chips
 - Diagnosis of bugs/ Yield improvement
 - Silicon debugging/ On-chip repair
 - Engineering fixes and re-spin
 - Determine mass-production test strategies
 - Speed sorting method
 - System bring-up test
 - Performance validation
- Mass production
 - Manufacturing test
 - Performance sorting
 - System test
 - Reliability test
- Ship to customers

ECE 156A

7

Post-Shipping stage

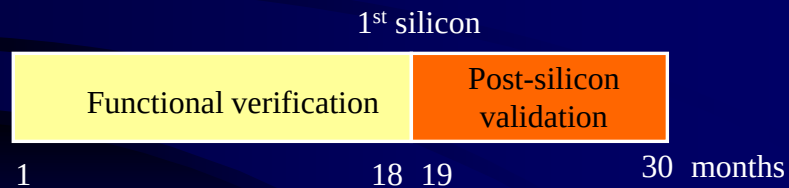
- Yield improvement
 - Identify common failure modes
 - Root causing
 - Generate fix(es)
- Test cost reduction
 - Identify redundant tests
 - Adapt test processes to save cost by removing redundant tests
- Failure analysis
 - Obtain customer returns
 - Diagnosis and silicon debug (root causing)
 - Feedback to design team

ECE 156A

8

Where Spends Most Resources?

- Functional Verification
- Post-Silicon Validation



ECE 156B

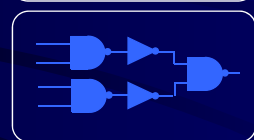
9

Design Data Hierarchy

- ↓ Specification
- ↓ HDL -- Behavior
- ↓ HDL -- RTL
- ↓ Netlist/schematic
- ↓ GDS II -- Layout
- ↓ Geometries on the die



```
behavior of nand4 is
begin process(a,b,c,d)
begin
z<=NOT(a AND b AND c AND d)
end process
end architecture behavior
```



- = ? Functional Verification
Usually piecemeal, e.g. FV, protocol checking
- = ? Logic Verification
- = ? Testing

Check against each level of transformation

ECE 156B

10

Synthesis (Front End)

- **Logic synthesis**
 - Extracting combinational logic from RTL
 - 2-level logic minimization
 - Algebraic decomposition
 - Multi-level logic minimization
 - Synthesis for test structures (test synthesis)
 - Timing optimization (gate-level)
- **Technology mapping**
 - Mapping gates to library cells

ECE 156B

11

Synthesis

- **Behavioral synthesis**
 - Resource allocation (Design automation conference, 1986, pp 474-480)
 - Pipelining (DAC 86, pp. 454-560, 461-466)
 - Control flow parallelization (IEEE Tran. On Computer-aided design of Ics, 1986, No 2, pp 259-269)
 - Communicating sequential processes
 - Partitioning (ICCAD 1989, pp. 280-283)
- **Sequential synthesis**
 - Register movement/retiming (FOCS, 1981, pp 23-26)
 - State minimization
 - State Assignment (Encoding) (IEEE TCAD, Dec, 1988, 1290-1300)
 - Synthesis for testable design
 - State machine verification

ECE 156B

12

Physical design synthesis (back end)

- Placement
 - Global/local
- Routing
 - Global/local
 - Clock tree design
- Physical design rule checking
 - Timing analysis (usually cell-based)
 - Layout vs. schematic (LVS) checking
 - Noise analysis and signal integrity check
 - Design rule sign-off

ECE 156B

13

Synthesis is an optimization process

- Area optimization
 - Gate count
 - Physical dimension
- Critical path (timing) optimization
 - Number of gates from PIs to POs
 - Delay from PIs to POs
- Optimized for testability
 - Redundancy removal
- Power optimization
 - Minimizing switching activities

ECE 156B

14

Post-Silicon Validation

- **DV (Design)**
 - Verify design sensitivity to various environmental conditions, (e.g. voltage, temperature, frequency, and process variation [to some degree]) with a given test suite -- fullchip level test (diagnostic)
- **SV (system)**
 - Verify that product is functional in a given system (designed to facilitate debugging) with real peripherals, BIOS, OS and Applications
- **CV (Commercial/Compatibility)**
 - Verify that product is functional across OEM systems, OSes, applications
- **CMV (Circuit Marginality)**
 - Verify that product is free from sensitivities to voltage/temp/frequency in a system level operation
- **MV (Manufacturing)**
 - Verify that a given product can be manufactured in HVM (high volume manufacturing); Yield is not impacted if circuit is manufactured in high volume, over time: process variation (+ all of the above)
- **RV (Reliability)**
 - Verify that the product has a low infant mortality rate and achieve low FIT (failure in time)

ECE 156B

15

Challenges in Validation

- Lack of an unified, systematic approach for all the problems
 - Rely on experiences
 - Rely on a collection of approaches
- A subtle problem can cause serious delay in time-to-market
 - It may take months to figure out what is going on for a “bug/defect”
 - The amount of data needed to be processed are extremely large
- New phenomenon appear as we move to very deep sub-micron domain

ECE 156B

16

Quality Issues

- How do we know we have done a good job?
 - Never can be sure?
 - Can our customers find out?
 - How is your manager going to say to his/her manager?
- How do we know what it is going wrong?
 - Locate a design error or manufacturing defect
 - Analysis the cause
 - Go back and fix the design or the process
 - can be costly

ECE 156A

17

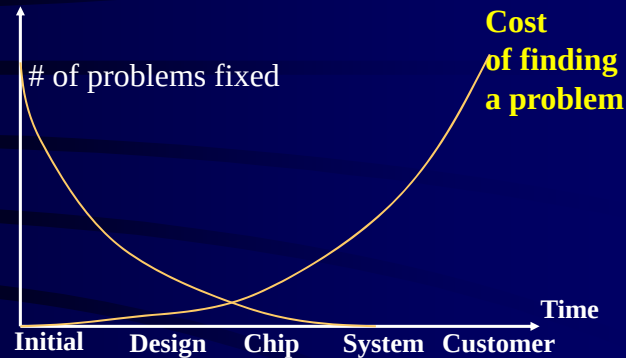
No Perfect Solution!

- Design automation is achieved by a collection of approaches - no single magic solution!
 - Involve 100 different algorithms
 - Involve tens of different tools
 - Involve different engineers to own different aspects of a design block
 - Apply more formal and systematic approaches

ECE 156B

18

Do Everything (Possible) Early



- Because the later you have a problem, the higher price it will cost you

ECE 156B

19

Design Art

- Design is a “constrained optimization” process
 - Optimize for your goals
 - Timing, power, area, functionality
 - Under some constraints
 - Constraints may not be well defined
 - Limited time, resources, expense, chip area, etc.
- Constrained optimization is to explore some tradeoff in the design space
 - Understanding the tradeoff is the key
 - The space may not be convex and hence, you may have many local optimal but not global optimal points

ECE 156A

20

The Existing Three Walls again

- We are hitting the **verification** wall
 - The simulation time drags time-to-market
 - Don't know if we have simulate enough
- We are hitting the **power** wall
 - Too many transistors on a single die
 - Leakage is serious for 90nm or below
 - Wireless applications require low power
- We are hitting the **process** wall
 - 22nm is in mass product
 - Most people stays at 90nm or 65nm
 - Analog world stays at 90nm or higher
 - Process variations are too high for today's design methods