CS 150 Spring 2004

FPGA CAD TOOL FLOW

Lab Lecture 2

January 30, 2004

Eric Chung
Greg Gresho
Aaron P. Hunt
Sandra Pinto

What is CAD?
Computer Aided Design

Why don’t humans design digital systems from start to finish?

What is the Flow? (generic)

- Design Entry
  - verification
- Synthesis
  - verification
- Place and Route
  - verification
- Hardware Programming
  - verification

What is the Flow? (specific)

Design Entry

- Schematics  (not in CS 150)
- Hardware Description Language (HDL)
  - Human readable
  - Hierarchical
  - Meaningful naming
  - "Think Hardware"

Why use an HDL?

Digital Design Productivity, in Gates/Week

Source: DataQuest

| Behavioral HDL | 2K-10K |
|RTL HDL| 1K-2K|
|Gates| 100-200|
|Transistors| 10-20|
Design Verification
- develop a test bench
  - Drive Inputs (a.k.a. test vectors)
    - random vectors – need many
    - targeted vectors – check anticipated problems
  - Check Outputs

Key concept: coverage
- how many of the potential problems have we tested?

Design Verification

ModelSim allows us to simulate HDL code, giving us the ability to identify any design errors before synthesis.

Synthesis

Not all Verilog is compatible with Synplify. Check Synplify often for warnings and errors if your synthesis fails.

Synthesis

converts HDL into a netlist of simple gates
Place and Route

FPGA is like a giant grid...
find an optimal arrangement of gates (LUTs) and paths to connect them

Accurate Timing Simulation

- ModelSim initially uses ideal values
  - not necessarily very accurate
- At the end of place and route, the Xilinx tools know the exact wire lengths and delays
- We can feed this exact timing back into ModelSim
- Simulate with real, accurate timing
- Tricky process: TAs will help

Hardware Verification

- Careful verification and testing after every step of the flow makes this last step easy
- However, debugging errors here can be incredibly tedious
  - incredibly-expensive in industry (millions of $)
- If it works, you’re done!