Lecture 23
Encounter in Depth and Conclusion

Xuan ‘Silvia’ Zhang
Washington University in St. Louis

http://classes.engineering.wustl.edu/ese461/
Some Final Administrative Stuff
Class Project Presentation

• Presentation order
  - Team 1 through 5
  - 16min per team (14min presentation + 2min Q&A)
  - all team members must participate

• Suggested contents
  - brief intro
  - design approach/debug methods
  - behavioral simulation results
  - design compiler results, post-synthesis simulation
  - physical layout
  - achieved performance (speed, power, area)
  - lesson learned
Final Project Report

- Due on 12/12 at noon
- Single submission as a team
- Required contents
  - design strategy (techniques applied)
  - achieved performance
  - report where the numbers come from
  - detailed explanation on the simulation results
  - behavioral simulation
  - synthesized simulation
    - digest of the timing, power, and area reports
    - division of work, individual contribution
    - appendix: all source codes, netlist, screen capture, etc
      (see final project description)
Course Evaluation

- Appreciate your feedback
- Start on today
- Please complete by December 12th
- Will account for 3 points in the final grade
Encounter in Depth
Chapter 4: Data Preparation

- Technology file
  - design rules and physical library: .LEF

- I/O assignment
  - manually create I/O assignment file

- Timing libraries
  - .lib

- Timing constraints
  - .sdc (write_sdc)

- Check designs
  - checkDesign
Chapter 5: Importing and Exporting Designs

- Prepare the netlist
  - synthesized netlist with unique cell types (.syn.v)
- Begin with LEF and Verilog
  - page 123
- Load config files
  - loadConfig
- Save and restore designs
- Import and export design data
  - floorplan, I/O, etc.
- Convert to GDSII
  - setStreamOutMode
• Utilization

• Edit Pins
  - use the Pin Editor: spreading pins, spacing

• Resize and Rotate

• Add core ring
  - core ring, block ring

• Add stripes
  - core area, over block area

• Global net connections
  - globalNetConnect -type pgpin -pin pin_name -all -override
Chapter 15: Placing the Design

- Prepare for placement
  - checkDesign, checkPlace
  - timeDesign -prePlace
  - createObstruct (no need)
  - planDesign or manual place and fix hard blocks

- Add well-tap and end-cap cells

- Place standard cells
  - placeDesign
  - setPlaceMode

- Check Placement
  - checkPlace
Chapter 16: Synthesizing Clock Trees

- Clock tree specification file
  - automatic mode
- Pre-CST and post-CST optimization
  - ckECO -preRoute
  - ckECO -clkRouteOnly
  - ckECO -postRoute
  - reportClockTree -postRoute
Chapter 16: Synthesizing Clock Trees

MacroModel pin mem_core/clk 20ps 18ps 18ps 18ps Off
AutoCTSRootPin SH1/I23/Z
NoGating rising
Buffer INV14 CLKBUF12 CLKBUF40 CLKBUF20 DEL4
MaxDelay 5ns
MinDelay 0ns
MaxSkew 500ps
End

Phase Delay 1

CTS delay1

Phase Delay 2

Buffer Input Transition Time

Skew

Phase Delay 3

Sink Input Transition Time

Phase Delay 4

MacroModel Pin mem_core/clk

Added by CTS

CTS does not trace through gates, because NoGating rising is specified, but the skew is balanced.
Chapter 20: Using the NanoRoute Router

- **Routing Phases**
  - global routing
  - detailed routing: switch boxed (SBoxes)

- **Preparation**
  - checkPlace, verifyGeometry (optional)

- **Specify routing layer**
  - routeBottomRoutingLayer
  - routeTopRoutingLayer

- **Routing commands**
  - routeDesign, setNanoRouteMode, setAttribute
  - globalRoute, detailRoute

- **Check congestion**
Monitoring and Verification

- Utilization (floorplanning)
  - target utilization (TU=%), effective utilization (EU=%)

- Congestion analysis table

<table>
<thead>
<tr>
<th>Layer</th>
<th>OverCon #Gcell</th>
<th>OverCon #Gcell</th>
<th>OverCon #Gcell</th>
<th>OverCon #Gcell</th>
<th>OverCon #Gcell</th>
<th>%Gcell OverCon</th>
</tr>
</thead>
<tbody>
<tr>
<td>Metal 1</td>
<td>22(0.01%)</td>
<td>10(0.00%)</td>
<td>0(0.00%)</td>
<td>0(0.00%)</td>
<td>0(0.00%)</td>
<td>(0.01%)</td>
</tr>
<tr>
<td>Metal 2</td>
<td>5531(2.39%)</td>
<td>1680(0.73%)</td>
<td>370(0.16%)</td>
<td>123(0.05%)</td>
<td>0(0.00%)</td>
<td>(3.33%)</td>
</tr>
<tr>
<td>Metal 3</td>
<td>4114(1.78%)</td>
<td>19(0.01%)</td>
<td>0(0.00%)</td>
<td>0(0.00%)</td>
<td>0(0.00%)</td>
<td>(1.79%)</td>
</tr>
<tr>
<td>Metal 4</td>
<td>1333(0.58%)</td>
<td>137(0.06%)</td>
<td>0(0.00%)</td>
<td>0(0.00%)</td>
<td>0(0.00%)</td>
<td>(0.64%)</td>
</tr>
<tr>
<td>Metal 5</td>
<td>5852(2.53%)</td>
<td>4(0.00%)</td>
<td>0(0.00%)</td>
<td>0(0.00%)</td>
<td>0(0.00%)</td>
<td>(2.53%)</td>
</tr>
<tr>
<td>Metal 6</td>
<td>27(0.01%)</td>
<td>0(0.00%)</td>
<td>0(0.00%)</td>
<td>0(0.00%)</td>
<td>0(0.00%)</td>
<td>(0.01%)</td>
</tr>
<tr>
<td>Total</td>
<td>16879(1.22%)</td>
<td>1850(0.13%)</td>
<td>370(0.03%)</td>
<td>123(0.01%)</td>
<td>0(0.00%)</td>
<td>(1.39%)</td>
</tr>
</tbody>
</table>

- Verify violations (Chapter 34)
  - connectivity
  - metal density
  - geometry
  - antennas
Conclusion
Topics Covered

- **Technology and Methods**
  - digital binary logic, Moore’s Law
  - level of abstraction -> design automation principles

- **Design Flow**
  - Algorithmic and architecture optimization
  - Synthesis: power, area, timing constraints
  - Static Timing Analysis
  - Physical Design: floorplan, place and route

- **Languages and Tools**
  - Verilog, Tcl
  - Synopsys VCS (Verilog Simulation)
  - Synopsys Design Compiler (Netlist Synthesis)
  - Cadence SOC Encounter (Physical Design)
Example Position

- Communications/DSP algorithms and efficient implementations.
- Demodulation, modulation, digital filters, physical layer in communications
- SOC architectures (interfaces, busses etc)
- Knowledge and hand-on experience with industry ASIC design flow including RTL coding, debugging, verification, synthesis and supporting timing closure.
- Experience with design tools such as NCSIM (and/or VCS), Cadence RC or Synopsys DC compiler,
- Experience with multiple IC tape-out in industry.
- Experience in chip bring up and performance measurement for IC and systems in laboratory to characterize and debug building blocks

This is a full time job in California, base salary > $100,000
The Trend: Follow, Catch, or Create?

- **Intelligent Recognition**
  - computer vision, artificial intelligence

- **Internet of Things**
  - Sensing (Analog)
  - Computing (Digital)
  - Wireless (RF)
  - Energy harvesting (Power)

- **Software-Hardware Co-design**
  - Analog/Digital/Mixed Signal/Radio...
  - Interface/Communication/Internet/Cloud...
  - Application/Regulation/Resource/Material...
ESE 566A: Modern System-on-Chip Design

- Advanced topics
  - system-on-chip
  - software/hardware partition
  - high-level synthesis
  - reliability, resilience, security

- More Project-centric
- More open-ended and research-oriented
Research Theme (XZ Group)

- **Problem**
  - designing micro-scale autonomous systems with enhanced security and resilience.

- **Approach**
  - co-design of algorithm, computer architecture, circuits, and sensing and actuation mechanisms.

- **Projects**
  - reconfigurable deep learning hardware
  - energy-efficient software-assisted power delivery
  - verifiable hardware against side-channel attack
  - sensor-fusion chip for vision-based robotic control
  - analog-coprocessor to speed up scientific computing
  - novel devices for non-reciprocal energy transfer
Questions?

Comments?

Discussion?