

# EEE 296O -- Advanced Timing Analysis

**Indicate Required or Elective Course here**

**Date:** Enter the date this outline was created

Elective Course

02/26/2007

**Indicate Course Area here (e.g., controls, communications, power, electronics or digital design)**

Digital Design

**Course Coordinator:** Enter the name of the person(s) responsible for the course here.

Dr. Jing Pang

**Catalog Description:** Enter the proposed catalog description and number of units for the course here.

3 units

The objective of this course is to cover the advanced static timing analysis techniques. Topics in this course include ASIC design methodology, static timing analysis, timing design constraints, design reports, clock timing issues, timing exception, operating conditions, hierarchical analysis, performance and power issues.

**Prerequisites:** Enter course numbers for all prerequisite classes here.

EEE/CSC 273

**Text:** Enter the proposed textbook here.

Advanced ASIC chip synthesis: using synopsys design compiler, physical compiler, and primetime,  
By Himanshu Bhatnagar,  
Publisher: Kluwer Academic Publishers, 2001  
ISBN 0-79-237644-7

**Additional Resources:** Enter any other recommended books or additional resources here, and indicate if they are optional or required.

**Optional:**

Timing verification of application-specific integrated circuits,  
By Farzad Nekoogar,  
Publisher: Prentice Hall PTR, New Jersey, 1999  
ISBN 0-13-794348-2

Verilog Styles for Synthesis of Digital Systems  
By David R Smith, Paul D Franzon.  
Publisher: Prentice Hall, New Jersey, 2000  
ISBN: 0-201-61860-5

**Course Objectives:** Enter a brief narrative describing the important topics this course covers and the expected learning outcomes here.

**Important Topics and Expected learning outcomes:**

- Learn the difference between traditional verification methodology and static timing analysis

- Learn the role of the timing analysis in the ASIC design flow
- Learn static timing analysis skills and tools, emphasizing timing constraints, operating conditions, timing reports, and analysis
- Learn about the important technology factors which affect the power and the performance

**Prerequisites by Topic:**

1. Enter a list of the important concepts that students need to know before taking this course here.  
**Hardware Description Language**
2. Important prerequisite concept 2  
**Computer-aided logic synthesis**
3. Important prerequisite concept 3  
**Computer-aided logical verification and testing**

**Topics Covered:**

1. Enter a list of the important concepts that students will learn in this course here.  
**ASIC design methodology, traditional verification methodology versus static timing analysis**
2. Important concept 2  
**Performance and power**
3. Important concept 3  
**Timing constraints**
4. Important concept 4  
**Timing exceptions**
5. Important concept 5  
**Design reports**
6. Important concept 6  
**Standard delay format file**
7. Important concept 7  
**Operating conditions**
8. Important concept 8  
**Hierarchical analysis**

**Evaluation:** Indicate how you plan to evaluate student performance in this course, including the weight given to each item. (e.g., homework (15%), projects (25%), midterm exam (30%) and final exam (30%). Describe any significant details, such as providing examples of typical projects.

<b>Grading:</b>	<b>Projects</b>	<b>25%</b>	
	<b>Two Midterms</b>	<b>30%</b>	
	<b>Quizzes</b>	<b>15%</b>	
	<b>Final (Comprehensive)</b>	<b>30%</b>	
	<b>93% A</b>	<b>90-92% A-</b>	
	<b>88-89% B+</b>	<b>82-87% B</b>	<b>80-81% B-</b>

**78-79% C+**  
**66-69% D**

**73-77% C**  
**60-65% D-**

**70-72% C-**  
**<60% F**

### Example Project: Hierarchical Analysis

- 1 use compile\_stamp\_model script with the two files “STAMP\_MODEL\_Y.mod” and “STAMP\_MODEL\_Y.data” to generate stamp model database and use “-output ./STAMP.Y” option to save it
- 2 source qtm\_model\_stack.pt
- 3 report\_qtm\_model
- 4 save\_qtm\_model -output ./QTM/STACK -format db
- 5 set search\_path { . Designs STAMP QTM DV\_BLOCK\_OPT }
  
6. set link\_path { \* libs/tut\_model\_lib.db QTM/STACK\_lib.db STAMP/Y\_lib.db }
7. read\_db AM2910.db
8. link\_design AM2910
9. report\_reference
10. set\_operating\_conditions -library pt\_lib -min BCCOM -max WCCOM
11. set\_wire\_load\_mode top

```
set_wire_load_model -library pt_lib -name 05x05 -min  
set_wire_load_model -library pt_lib -name 20x20 -max
```

12. repot\_port
13. Set the Timing Constraints

```
create_clock -period 30 [get_ports CLOCK]
```

```
set clk [get_clocks CLOCK]
```

- clock uncertainty should be 0.5
- minimum clock latency is 3.5
- maximum clock latency is 5.5
- minimum clock transition is 0.25
- maximum clock transition is 0.3
- Specifies the value of setup equal to 0.5 and hold time equal to
- for clock gating checks.
- minimum clock pulse width is 2.0

14. Displays attributes on the current\_design by using report\_design
15. Shows possible timing problems for design by using check\_timing

16. Setting up the port delays and checking the timing by using the following scripts:

```
#####  
set nonclock [remove_from_collection [all_inputs] [get_ports CLOCK]]  
set_input_delay 0.0 $nonclock -clock $clk  
set_output_delay 2.0 [get_port INTERRUPT_DRIVER_ENABLE] -clock $clk  
set_output_delay 1.25 [get_port MAPPING_ROM_ENABLE] -clock $clk
```

```
set_output_delay 0.5 [get_port OVERFLOW] -clock $clk
set_output_delay 1.0 [get_port PIPELINE_ENABLE] -clock $clk
set_output_delay 1.0 [get_port Y_OUTPUT] -clock $clk
```

```
set_driving_cell -lib_cell IV -library pt_lib [all_inputs]
set_capacitance 0.5 [all_outputs]
check_timing
```

17. Generate Design Vision Script called “AM2910.dctcl” and the Prime Time Script called “AM2910.pt” by using

```
write_script
```

Find online help for this script.

```
##### Initial Timing Analysis #####
```

18. report\_constraint
19. report\_constraint -all\_violators
20. report\_timing

**Question 1:** Use man help to find out the difference between “report\_constraint” and “report\_timing” script, give a summary of the difference in your report.

**Question 2:** Show the timing reports.

```
##### Setting Timing Exceptions #####
```

21. set\_false\_path -from U3/OUTPUT\_reg[\*]/CP -to U2/OUTPUT\_reg[\*]/D
22. set\_multicycle\_path -setup 2 -from INSTRUCTION[\*] -to U2/OUTPUT\_reg[\*]
23. set\_multicycle\_path -hold 1 -from INSTRUCTION[\*] -to U2/OUTPUT\_reg[\*]

24. write a script to generate exception report
25. write a script to generate ignored exception report
26. write a script to generate all\_violated constraints
27. write a script to report setup and hold timing violations

```
##### Context Characterization #####
```

28. set\_operating\_conditions -library pt\_lib WCCOM
29. report\_design
30. characterize\_context {U2 U3}
31. write\_context U2 -output ./SCRIPT\_OP/UPC.char.dctcl -format dctcl
32. write\_context U3 -output ./SCRIPT\_OP/REGCNT.char.dctcl -format dctcl
33. write\_script -format ptsh -output ./SCRIPT\_OP/AM2910.new.pt

# Don't quit prime time. Start design vision next.

```
##### Block Optimization by Design Vision #####  
.synopsys_dc.setup file:
```

```
set search_path { . designs SCRIPT_OP STAMP QTM DC_BLOCK_OPT }  
set target_library { libs/tut_model_lib.db QTM/STACK_lib.db STAMP/Y_lib.db }  
set link_library { libs/tut_model_lib.db QTM/STACK_lib.db STAMP/Y_lib.db }  
define_design_lib WORK -path ./work
```

```
34. read_db UPC.db  
35. link  
36. set_local_link_library {}  
37. source UPC.char.dctcl  
38. compile  
39. write -format verilog -output designs/UPC.opt.v
```

```
40. read_db REGCNT.db  
41. link  
42. source REGCNT.char.dctcl >> /dev/null  
43. compile  
44. write -format verilog -output designs/REGCNT.opt.v
```

```
#Definition: /dev/null: On UNIX, this is a virtual-file that can be written to. Data written to this file gets  
#discarded. It is similar to the file call NUL on Windows machines.
```

```
# quit Design Vision after you finish the above scripts and then go back to prime time.
```

```
##### Swapping in the optimized Blocks #####
```

```
45. read_verilog {REGCNT.opt.v UPC.opt.v}  
46. current_design AM2910  
47. swap_cell U3 {REGCNT.opt.v:REGCNT}  
48. swap_cell U2 {UPC.opt.v:UPC}  
49. Write a script to apply the constraints written in "AM2910.new.pt" file  
50. Write a script to check if the timing constraints are complete  
51. report_constraint -all_violators  
52. report_timing
```

```
##### Case analysis #####
```

```
53. set_case_analysis 0 [get_ports CONDITION_CODE]  
54. Write a script to generate case analysis report  
55. Write a script to generate the disabled timing report  
56. report_timing  
57. set_case_analysis 1 [get_ports CONDITION_CODE]  
58. report_timing  
59. remove_case_analysis [get_ports CONDITION_CODE]  
60. Write a script to generate case analysis report
```

##### Mode analysis #####

61. report\_mode
62. set\_case\_analysis 0 [get\_pins U4/OPERATION[\*]]
63. set\_mode data U4/core
64. report\_mode
65. report\_timing -to Y\_OUTPUT\*
66. report\_case\_analysis > rn11.txt
67. reset\_mode
68. report\_mode

### REPORT

1. Show all of your scripts.
2. Answer Questions 1 and 2. Compare the timing reports before block optimization and after context characterization and block optimization.
3. Give a summary of Hierarchical Design Analysis method, explain how to identify timing problems, and what are the possible methods to improve design timing.

**Contribution of Course to the Professional Education Component:** Indicate how this course fulfills this ABET requirement (e.g., developing student's analytical and critical-thinking skills, science and design content, etc.) **Note: This section is not required for graduate courses.**

**Relationship of Course to Program Outcomes:** Indicate how this course fulfills this ABET requirement (e.g., knowledge of mathematics, engineering science, core topics in major, development of problem solving skills, etc.) **Note: This section is not required for graduate courses.**

## Course Outline/Schedule

<i>Week</i>	<i>Topic</i>	<i>Text Reference</i>
1	ASIC design methodology, traditional verification methodology versus static timing analysis	Himanshu text: 1-17
2	Verilog Review	Refer David text
3	Improving timing, area and power performance	David text: 177-196
4	Technology library, delay model and delay calculation	Himanshu text: 101-122
5	Timing in ASICS, clock definitions, basic clocking, synchronization and metastability	Farzad text: 45-92
6	Midterm 1 review and midterm 1	
7	Boundary condition, constraining designs	Himanshu text: 101-122
8	Optimizing Design, design reports	Himanshu text: 125-148, handout
9	Timing analysis environment and design commands of timing analysis tools	Himanshu text: 239-259
10	Standard Format Delay File	Himanshu text: 225-237
11	Midterm 2 review and midterm 2	
12	Prelayout and postlayout timing analysis and timing reports	Himanshu text: 274-291
13	Timing exceptions	Himanshu text: 261-273
14	Hierarchical analysis	Himanshu text: 292-303, handout
15	Analyzing designs with asynchronous logic, final exam review	Handout
16	Final Exam	