Syllabus

COEN207 SoC Verification
Department of Computer Engineering
Santa Clara University

Dr. Ming-Hwa Wang
Phone: (408) 525-2564
Course website:
Office Hours:
Summer Quarter 2012
Email address: mwang2@engr.scu.edu
http://www.cse.scu.edu/~mwang2/verification/
Monday 9:15-9:45pm

Course Description

A typical SoC costs tens of millions of dollars and involves tens of engineers in various geographical locations. It also incorporates a large number of heterogeneous IP (intellectual property) cores. A single error may dictate a Fab spin of over a million dollar, and typically costs much more by delaying TTM (time-to-market). Therefore, SoC verification is a major challenge that needs to be mastered by design engineers. This course presents various state-of-the-art verification techniques used to ensure thorough testing of the SoC design. Both logical and physical verification techniques will be covered. Also, the use of simulation, emulation, assertion-based verification, and hardware/ software co-verification techniques will be discussed.

Prerequisites

Logic Analysis and Synthesis (COEN 200), Logic Design Using HDL (COEN 303), Data Structures and Computer Algorithms.

Required Textbooks

1. "Verification Methodology Manual for SystemVerilog", Janick Bergeron, Eduard Cerny, Alan Hunter, Andy Nightingale, ISBN: 978-0387255385, Springer, 2005
2. "SystemVerilog for Design: A Guide to Using SystemVerilog for Hardware Design and Modeling, 2nd Edition", by Stuart Sutherland, Simon Davidmann, Peter Flake, P. Moorby, ISBN: 978-1441941251, Springer, 2010

Recommended Textbooks

1. "SystemVerilog for Verification: A Guide to Learning the Testbench Language Features, 3rd Edition", by Christian B. Spear, ISBN: 978-1461407140, Springer 2012
2. "Open Verification Methodology Handbook: Creating Testbenches in SystemVerilog and SystemC", by Mark Glasser, Harry Foster, Tom Fitzpatrick, Adam Rose, Dave Rich, ISBN: 978-0123743985, Morgan Kaufmann, 2009
3. "The Art of Verification with SystemVerilog Assertions", by Faisal Haque, Jonathan Michelson, Khizar Khan, ISBN: 978-0971199415, Verification Central, 2006

References

1. "Writing Testbenches using SystemVerilog", by Janick Bergeron, ISBN: 978-1441939784, Springer, 2010
2. "SystemVerilog Functional Verification", by Sasan Iman, ISBN: 978-0071489041, McGraw-Hill, 2009
3. "ESL Design and Verification: A Prescription for Electronic System Level Methodology" by Grant Martin, Brian Bailey, Andrew Piziali, ISBN: 0123735513, Morgan Kaufmann 2007
4. "Co-verification of Hardware and Software for ARM SoC Design" by Jason Andrews, ISBN-10: 9780750677301, Newnes, 2005
5. "Comprehensive Functional Verification: The Complete Industry Cycle" by Bruce Wile, John Goss, Wolfgang Roesner, ISBN-10: 9780127518039, Morgan Kaufmann 2005
6. "Verilog and SystemVerilog Gotchas: 101 Common Coding Errors and How to Avoid Them", Stuart Sutherland, Don Mills, ISBN: 978-1441944023, Springer 2010
7. "FPGA Simulation: A Complete Step-by-Step Guide", Ray Salemi, ISBN: 978-0974164908, Boston Light Press, 2009
8. "Step-by-step Functional Verification with SystemVerilog and OVM", Sasan Iman, ISBN: 978-0981656212, Hansen Brown Publishing, 2008
9. "Assertion-Based Design (Information Technology: Transmission, Processing and Storage, 2nd Edition", Harry D. Foster, Adam C. Krolnik, David J. Lacey, ISBN: 978-1402080272, Kluwer Academic, 2004
10. "ESL Design and Verificaiton: A Prescrption for Electronic System Level Methodology (System on Silicon)", Grant Martin, Brian Bailey, Andrew Piziali, ISBN: 978-0123735515, Morgan Kaufmann, 2007
11. "High-Level Synthesis: from Algorithm to Digital Circuit", Philippe Coussy, Adam Morawiec, ISBN: 978-9048179237, Springer, 2010
12. "System Verilog for Verification", Tom Fitzpatrick, A. Salz, David Rich, Stuart Sutherland, ISBN: 978-0387255712, Springer, 2006
13. "SystemVerilog Assertions Handbook, 2nd Edition", by Ben Cohen, Srinivasan Venkataramanan, Ajeetha Kumari, Lisa Piper, ISBN: 978-0970539489, VhdlCohen Publishing 2010

Course Objectives

1. To learn advanced and cutting edge state-of-the-art knowledge and implementation in SoC verification.
2. To read and understand research publications in the technical area of SoC verification, beyond that of the traditional textbook level.
3. To conduct independent/group project and to equip for scholarly research in SoC verification.

Expected Learning Outcomes

1. Understand SoC verification and their implementation.
2. Understand the idea of SoC verification and implement example reusable verification environment using SystemVerilog and OVM.

3. Understand details of constraint randomization, coverage-driven verification, assertion-based verification, formal verification, co-verification, co-simulation, ESL modeling, etc.
4. Read current research papers and implement example research group project in SoC verification.
5. Explore other advanced topics if time permits.

Grading Policy

Course grade is determined based on the total score (maximum 1000 points + up to 200 optional bonus points for extra work) from:

1. Mid-term and final exams of 250 points each (close book with one A4 note and a calculator.) Makeup exams (must have a very good reason for makeup) are much difficult than normal exams.
2. Two programming assignments of 100 points each (late penalty: 20 points/day.) Makeups are more difficult too.
3. A group programming term project of 300 points (late penalty: 60 points/day.) No makeup is allowed.
4. Bonus assignments will be assigned at each lecture with 20 points each, and due before next lecture begin. You can either email the assignment electronically or hand in the written one in class. The solution for bonus point will be posted on my protected web page. No late work accepted for bonus assignments. 75-80% of exam questions are similar to bonus.
5. Class average targeted at **A-**.

Table 1: Grade-score table

1000 - 1200	950 - 999	900 - 949	850 - 899	800 - 849	750 - 799	700 - 749	650 - 699	0 - 649
A	A-	B+	B	B-	C+	C	C-	F

Course Schedule (Monday 7:00pm-9:45pm)

Table 2: Course Schedule

#	week	Readings	Remarks
1	5/14	introduction	
2	5/21	object-oriented	submit due 5/21
3	5/28	SystemVerilog	
4	6/3	SystemVerilog	program #1 due 6/2
5	6/10	OVM	
6	6/17	OVM	
7	6/24	testbench	program #2 due 6/23
8	7/8	testbench	problem due 7/8 mid-term exam 7/8
9	7/15	assertion	group & topic due 7/12 paper presentation 7/15
10	7/22	assertion	proposal due 7/22
11	7/29	ESL	
12	8/5	co-simulation	
13	8/12	review/evaluation	final exam 8/12
14	8/19	review/evaluation	project defense 8/19

Reminder

- No cheating, and no register complaint without talking to me first.
- No incomplete. No sit-in or audit the class except formally registered.
- Read files under /home/mwang2/tips for help.
- Handouts, assignments, and solutions will be posted on the web. You should check the class web site at least once a week. You are responsible for printing and bring the handout to the class.
- Put a cover page posted my website on top of each of your submitted written assignment.
- Office hours: Monday 9:15pm-9:45pm.

Honor Code

All students taking course in the school of engineering agree, individually and collectively, they will neither give nor receive unpermitted aid in examinations or other course work that is to be used by the instructor as a basis of grading.

Disability Accommodation Policy:

To request academic accommodations for a disability, students must contact Disability Resources located in The Drahmann Center in Benson, room 214, (408) 554-4111; TTY (408) 554-5445. Students must provide documentation of a disability to Disability Resources prior to receiving accommodations.