Electrical and Computer Engineering
ECE 2612 Digital Circuit Design Syllabus – Fall 2012

This course considers binary number systems, codes, truth tables and the fundamental operations and descriptions of digital logic circuits. The implementation of combinational and sequential digital logic is taught using the Verilog hardware description language. Complex digital logic, state machines, hierarchical design and verification methods are covered. The designs are developed using commercial design tools. They are simulated and implemented in programmable gate array hardware.

Textbooks:

- *Digital Design with RTL Design, VHDL, and Verilog*, by Frank Vahid, 2nd Ed., ISBN 9780470531082, Wiley Publishing;
- [Optional] *Verilog HDL Synthesis, A Practical Primer,* by J. Bhasker - ISBN 978-0965039154, Star Galaxy Pub. (Three reference copies available in Engineering Library.)

Lecture Time/Place: TR 9:30-10:50am/EA 308

Lecturer: Frank P. Higgins, PhD

Lecturer office: EA 702B (near the rear stairway)

Lecturer office hours: T 11:00am-2:30pm; R 11:00am-2:30pm

Lecturer email: fhiggins@temple.edu

Course website: https://sites.google.com/a/temple.edu/ece2612/

Blackboard: Digital Circuit Design – 2612 – Fall 2012

Accommodation: Any student who has a need for accommodation based on the impact of a disability should contact me privately to discuss the specific situation as soon as possible. Contact Disability Resources and Services at 215-204-1280 in 100 Ritter Annex to coordinate reasonable accommodations for students with documented disabilities.

Grading Policies:

- Quizzes – 20%
- In-class exams – 40%
- Final exam – 40%
- Extra credit project – 1-10%

Homework will be assigned (through blackboard), but not graded. The homework problems are selected to re-emphasize concepts covered in the lectures. Unannounced in-class quizzes (10-15 minutes) will cover these and be graded. There will be no make up quizzes. The lowest quiz score will be discarded from your quiz average. There will be three in-class exams (see schedule below). There will be no make

up exams. An unexcused absence will result in a zero for that exam. If an exam is missed because of an excused absence, the score for that exam will be the lower of the other two exams. If a second exam is missed, that score will be zero. The final exam will follow the university schedule.

All quizzes and exams will be closed book, closed notes, no calculators, unless otherwise noted. Any special tables, code examples, etc. will be provided as part of the exam.

Students who have adequate amounts of free time can select and execute a special project on a topic of interest, for extra credit. The subject and amount of extra credit will be negotiated on an individual basis with the course lecturer, and must be agreed to in writing before the work can proceed. Example projects would be extensions of designs done in class or the lab. Projects commonly involve a high level block diagram, a test bench, the design, its simulation and implementation. A short four-page paper summarizing the design will formalize the completion of the project.

Schedule:

Week No.	Subjects Covered
1	**Introductory concepts:** Applications, representations and implementations of digital design – (block diagrams, schematics, state machines, truth tables, timing diagrams); Number systems – binary, decimal, hexadecimal, conversion examples; ASCII code; Boolean logic – introduction to basic notation and gates; Introduction of Verilog as a hardware description language; define and illustrate simulation and synthesis; Introduce lab design project; Top level design concepts; Hierarchy and reuse.
2	**Combinational logic:** Logic operations on number systems – addition, subtraction, 2s complement; XOR gates and their relationship to adders. Boolean logic and its application to combinational logic.
3	Boolean algebra, axioms, operator precedence, DeMorgan's theorem; Developing combinational logic from truth tables – Sum of Products; Encoding and decoding; Multiplexers; Binary Coded Decimal; Relationship of Verilog logic symbols to gate level schematics.
4	Verilog representations of combinational logic – in class exercises; Verilog instantiations, design hierarchy, alternative port styles and instantiation styles – in class exercises; Creating and interfacing self-checking test benches with a top level design – in class exercises; **Exam 1 (*Introductory concepts* through *Combinational logic*).**
5	**Synchronous logic:** RS latch description, implementation and examples; Extension to D flip flops; Verilog representations; Clocks and timing issues; Implications of gate delay; D-ff timing characteristics – setup time, hold time, clock to Q; Implications to synthesis timing closure; Basic register – storing multiple bits.

6	Finite State Machines – formalisms, state diagrams, capturing system behavior; Putting combinational and synchronous logic together to implement FSMs; Implementations and examples in Verilog; coding styles – in class exercises;
7	Common counter designs; Count down dividers/timers; Single versus multiple clock domains; Asynchronous versus synchronous reset; Count load and enable signals; Schematic representations; Button debouncer design; Implementations and examples in Verilog – in class exercises;
8	Verilog representations/examples of FSM's – in class exercises; Verilog representations/examples of counters – in class exercises; **Exam 2 (*Synchronous logic*).**
9	**Datapath components:** N-bit registers; Shift registers; Register files; Extension to Random Access Memory; Verilog representations and examples – in class exercises;
10	N-bit adders; N-bit comparators; N-bit subtractors; Arithmetic Logic Units; Verilog representations and examples – in class exercises;
11	Parallel to serial/serial to parallel conversion; Verilog representations and examples – in class exercises;
12	Data paths and busses – multiplexing versus tri-stating; Verilog representations and examples – in class exercises; **Exam 3 (*Datapath components*).**
13	**Advanced topics:** Integrating the lab design project; Verilog in class exercises; [Fixed point multiplication; Floating point arithmetic;] [3 Tap FIR filter (Vahid, Example 5.10)] [6 instruction programmable processor (Vahid, Section 8.4)] [4 bit programmable processor (gnome4)]] [SPI Bus] [I2C Bus] [IEEE-488 GPIB] TBD …
14	… Physical implementations of a digital design; Review for Final Exam.