



# UNIVERSITY OF BRIDGEPORT

School of Engineering  
Department of Computer Science and Engineering

**CpEG 447 Logic Synthesis Using FPGAs**  
Spring 2014, Tuesday and Thursday 3:00PM-4:15PM

---

**Instructor:** Miad Faezipour  
**Email:** mfaezipo@bridgeport.edu  
**Office:** Technology Building-Room 155  
**Office Hours:** Wednesdays 3:00-5:00pm or by appointment  
**Teaching Assistant:** To be announced

**Credit Hours:**  
3 semester hours:

**Pre-requisites:**  
CpE 315 (Digital System Design II with Lab) or Equivalent  
Debugging Experience with some high level programming language

**Required Reference:**

1. **Class Notes and Class Webpage:** Canvas [www.bridgeport.edu/myub](http://www.bridgeport.edu/myub)

**Bibliography and Other References:**

2. **FPGA-Based System Design** by Wayne Wolf; Prentice Hall (2004)
3. **Rapid Prototyping of Digital Systems – Quartus II Edition** by Hamblen, Hall and Furman (2006)
4. **Digital Systems Design Using VHDL** by Roth and John, 2<sup>nd</sup> edition; PWS (2008)
5. **HDL Chip Design**, by Smith, Doones Publications (1996)
6. **Verilog Styles for Synthesis of Digital Systems** by Smith and Franzon, Prentice Hall (2000)
7. **Verilog HDL** 2nd Edition, by Palnitkar (2003)
8. **Digital Logic Design, by Holdsworth and Woods, 4th Edition (2002)**
9. **Designer's Guide to VHDL**, by Ashenden, 3<sup>rd</sup> Edition (2008)

**Course Description:**

This course covers digital logic design using textual design entry and VHDL, behavioral, structural and data flow descriptions, technology-dependent vs. technology-independent designs, logic synthesis and implied hardware for HDL codes, CPLD, rapid prototyping and retargeting designs for FPGAs. A major design project is given in this course. Simulations, projects and labs are performed using commercial and/or academic CAD tools.

**Policies:**

1. All classes will be conducted as per the university regulations.
2. Extensive computer work will be required for programming assignments and labs. The class will use Altera/Xilinx Boards, Quartus II and/or ModelSim/Synopsys, Mentor Graphics synthesis and simulation CAD tools.
3. Laboratory site is Tech 211B. Students will have access to the lab in the designated lab hours.
4. Homework assignments will be assigned throughout the course. Copying is strictly prohibited. All homework assignments must be **individual** works. Cheating in assignments/exams/projects will result in automatic **F** grade.
5. Late assignments are not accepted at all.

6. Special requests must be made well ahead of time. Explicit permission of the instructor will be required for such cases.
7. Several design projects and laboratory assignments will emphasize the practical aspects of the course. Designs will be implemented using programmable logic. Design projects and lab assignments should be carried out in groups of two or more. Well documented formal written reports (using a word processor) will be required for all labs and design projects. Oral presentations and demos may be required for some projects.
8. Note that lab work is to be carried out outside lecture hours. Lab hours will be announced on the course webpage.
9. Regular attendance is recommended. Students are expected to log on to the Canvas class webpage [www.bridgeport.edu/myub](http://www.bridgeport.edu/myub) frequently in order find any current postings. You can get a UB email account by going to: [www.bridgeport.edu/ubnet](http://www.bridgeport.edu/ubnet).

**Grading Policy:**

Following is the tentative grading policy:

- 1. Homework problems** - verification of principles and software familiarization **(20%)**
- 2. Design projects** **(60%)**
  - a) Design and test of digital systems including at least two different target technologies test normally done in groups of two or more
  - b) Report presentation - oral and written includes notebook documenting all design and test
  - c) Labs (tentative)
    - i) LED Pong
    - ii) Pulse Width Modulation (PWM)
    - iii) Altera Specific Technologies
    - iv) Xilinx ISE
    - v) USRT
    - vi) Final Project Using a UART
- 3. Examination** **(20%)**

**Lecture Topics (Tentative):**

1. Introduction
  - a) Microelectronic design methodologies
  - b) ASICs - what and why
  - c) The ASIC process
  - d) FPGAs - what and why
2. Review of Verilog (Very brief – special sessions for those who need Verilog reinforcement)
  - a) Basic design features
  - b) Schematics vs. architectures
  - c) Behavioral, dataflow and structural descriptions
  - d) A few simple examples
3. System Implementation Strategies
  - a) The FPGA paradigm
  - b) Design styles - SW vs. HW paradigms
  - c) Design methodologies - structural, behavioral, physical
  - d) Technology-independent design - advantages and disadvantages
  - e) Hierarchy
4. Introduction to FPGA Architectures – FPGA Fabrics
  - a) Historical perspectives
  - b) SRAM programming technology (Xilinx)

- c) Anti-fuse programming technology (Actel)
  - d) EPROM (CPLD) programming technology (Altera)
  - e) Chip I/O
  - f) FPGA Fabrics
  - g) Commercially available FPGAs
  - h) Benchmarking
5. Review of Logical Design Aspects (Brief)
- a) Combinational design – logic design process
  - b) Delay, fanout, power dissipation
  - c) Logic optimization – both technology-independent and technology-dependent
  - d) Sequential design
  - e) State machine implementations - encoded vs. one-hot, Mealy vs. Moore, communication protocols
  - f) Clocking methodologies, clock skew, retiming
6. More Verilog
- a) The *always* block - combinational and sequential
  - b) Sequential statements
  - c) Hierarchical modeling concepts
  - d) Data types
  - e) Sequential and parallel blocks
  - f) Structural implementations - component instantiation, generate
  - g) Verilog modeling techniques for combinational and sequential logic
  - h) Hierarchical modeling
  - i) Tasks and Functions
  - j) Timing and delays
  - k) User-defined Primitives
7. Behavioral Synthesis
- a) Datapath controller architectures
  - b) Scheduling and allocation, communication
  - c) Hardware implementations ASAP and ALAP scheduling
  - d) Pipelining
8. Design Methodologies
- a) Design Processes
  - b) Design Standards
  - c) Design Verification
  - d) Case studies
9. Technology Dependent Design
- a) Synthesis and fitting
  - b) CPLDs vs. FPGAs
  - c) Case studies
  - d) Synthesis directives
  - e) Writing efficient and synthesizable VHDL descriptions
  - f) More VHDL
10. Large-Scale Systems
- a) Busses – bus protocols, logic design for busses
  - b) Platform FPGAs
  - c) Multi-FPGA systems
  - d) Novel Architectures
11. Optimizing Data Paths
- a) Pipelining

- b) Resource sharing
  - c) Case studies
11. Verilog and Testing
- a) The test bench
  - b) Assert, report, severity error
12. Additional Topics – as time permits
- a) Design reuse
  - b) Rapid prototyping
13. Software Design Tools – Altera, Xilinx, Mentor Graphics (HDLDesigner), ModelSim
- a) Design entry - schematic capture, textual description, state diagrams, truth tables
  - b) Logic simulation
  - c) Timing analysis
  - d) Compiler
  - e) Floor planner

***Interesting Web Sites:***

[www.altera.com/](http://www.altera.com/)

[www.xilinx.com/](http://www.xilinx.com/)

Additional web sites will be provided.

***Important Dates:***

**Spring Break (No class):** March 16 – March 23, 2014.

**Final Test:** Week of May 5 – May 9, 2014.

**\*Note: These descriptions and timelines are subject to change at the discretion of the Instructor.**