ECE 448
FPGA and ASIC Design
with VHDL

Spring 2013
ECE 448 Team

Course Instructor:  Kris Gaj  
kgaj@gmu.edu

Lab Instructors (TAs):

Wednesday & Thursday sections:  
Umar Sharif  
malik.umar.sharif@gmail.com

Help with the development of new labs:  
Rabia Shahid  
rabia.shahid4@gmail.com
A few words about You

5 BS EE students

23 BS CpE students
Undergraduate Computer Engineering Courses

Color code:
- BS EE
- BS CpE
Computer Engineering Course Progression

This is not a suggested schedule. It only illustrates dependencies and shows courses in earliest possible semester.

*) Math 203 can be taken concurrently with ECE 220
Math 203 can be taken concurrently with ECE 220

This is not a suggested schedule. It only illustrates dependencies and shows courses in earliest possible semester.

Electrical Engineering Course Progression

PHYS 161 → PHYS 160 → MATH 114 → ECE 201 → CS 222
PHYS 261 → PHYS 260 → MATH 213 → MATH 214 → MATH 203 → STAT 346 → ECE 220 → ECE 280
PHYS 263 → PHYS 262 → MATH 213 → MATH 214 → MATH 203 → STAT 346 → ECE 220 → ECE 280
ECE 201 → CS 112

Prerequisite
Must be taken in sequence

Co-Requisite
Should be taken concurrently but not earlier

Co-Requisite +
Suggested to be taken in sequence

Semester
Courses between dashed lines can be taken concurrently

*) Math 203 can be taken concurrently with ECE 220
Digital system design technologies coverage in the CpE & EE programs at GMU

Microprocessors

ECE 445: Computer Organization

ECE 447: Single Chip Microcomputers

FPGAs

ECE 448: FPGA and ASIC Design with VHDL

Digital Circuit Design

ECE 431

ASICs

ECE 545: Digital System Design with VHDL

ECE 645: Computer Arithmetic

Real-Time Embedded Systems

ECE 612

ECE 511: Microprocessors

ECE 611: Advanced Microprocessors

ECE 586: Digital Integrated Circuits

ECE 681: VLSI Design for ASICs
Course Hours

Lecture:
Monday, Wednesday
1:30-2:45 PM, Enterprise Hall, room 275

Lab Sessions:
Wednesday, Thursday
7:20-10:00 PM, The Nguyen Engineering Bldg., room 3208
Tuesday session canceled!

Lab sessions start this week!!!
It is very important that you attend the first lab session!
ECE 448 Section Assignment Rules

• You should do your best to attend all lab meetings of the section you are registered for.

• If you have missed a meeting of your section please attend a meetings of the other section, but give preference in access to the lab computers to the students attending their own section.

• All lab assignment **demos** should be normally done **exclusively during the class time of your section**.

• Any requests for exceptions to these rules (due to illness, accident, etc.) should be well documented and presented to the TA & primary instructor for approval.
You are welcome to attend all office hour sessions! You can direct your questions regarding lab assignments to the TA and myself.

Umar Sharif, Engineering 3208
• Tuesday, 3:00-4:00pm, 6:30-7:30pm
• Wednesday, 3:00-4:00pm
• Thursday, 11:00am-12:00pm

Kris Gaj, Engineering 3225
• Monday, 3:00-4:00pm, 6:30-7:30pm
• Wednesday, 3:00-4:00pm

Do your best to avoid “chasing” the TA outside of his office hours! He has other jobs to do!
Lab Access Rules and Behavior Code

Please refer to

Computer Engineering Lab website

and in particular to

Access rules & behavior code
Grading criteria

First part of the semester (before the Spring break)

Lab experiments - Part I
20%

Quizzes & homework: 5%

Midterm exam for the lecture: 10%
Midterm exam for the lab: 15%

Second part of the semester (after the Spring break)

Lab experiments - Part II
20%

Quizzes & homework: 5%

Final exam
25%
Tentative Grading Scheme for the Labs
(the exact point amounts may still change)

Lab 1: Developing Effective Testbenches (Parts a & b) – 4 points
Lab 2: Implementing Combinational Logic in VHDL – 5 points
Lab 3: Implementing Sequential Logic in VHDL – 5 points
Lab 4: State machines – 6 points
Lab 5: VGA display – 6 points
Lab 6: DSP & FPGA Embedded Resources – 6 points
Lab 7: PicoBlaze & Serial Communication – 6 points
Lab 7a: Logic Analyzer – 2 points
Penalties and Bonus Points

Penalties:

one-week delay: 1/3 of points

i.e., you can earn max. 4 out of 6 points

No submissions or demos will be accepted more than one week after the assignment is due!

Bonus points:

Majority of labs will have opportunities for earning bonus points by doing additional tasks
Flexibility in the Second Part of the Semester

Schedule A:

Lab 5: VGA display (2 weeks) – 6 points
Lab 6: DSP & FPGA Embedded Resources (2 weeks) – 6 points
Lab 7: PicoBlaze & Serial Communication (2 weeks) – 6 points
Lab 7a: Logic Analyzer (in class) – 2 points

Total: 20 points

Schedule B:

Lab 5: VGA display (3 weeks) – 6 points
Lab 6: DSP & FPGA Embedded Resources (3 weeks) – 6 points
Lab 7a: Logic Analyzer (in class) – 2 points

Total: 14 points
Flexibility in the Second Part of the Semester

Schedule A+:

• Intended for students who do exceptionally well in the first part of the semester (≥ 90% of points for Labs 1-4)
• An open-ended project proposed by students, the TA, or the instructor
• Can be done individually or in groups of two students
• Schedule: Detailed Specification (1 week)
  Milestone 1 (2 weeks)
  Milestone 2 (2 weeks)
  Final Report & Deliverable (1 week)

Total: 25 points
Required Textbook

Pong P. Chu,
*FPGA Prototyping by VHDL Examples: Xilinx Spartan-3 Version*,

Recommended Textbook

Stephen Brown and Zvonko Vranesic,
Basic Textbook

Part I Basic Digital Circuits
   - combinational
   - sequential
   - state machines and ASM charts

Part II I/O Modules
   - video
   - serial communication
   - keyboard
   - mouse

Part III PicoBlaze Microcontroller
   - block diagram
   - instruction set
   - I/O interface
   - interrupts
ECE 448, FPGA and ASIC Design with VHDL

Topics

VHDL:

- writing synthesizable RTL level code in VHDL
- writing testbenches

FPGAs:

- architecture of FPGA devices
- embedded resources (memories, DSP units)
- tools for the computer-aided design with FPGAs
- current FPGA families & future trends
High-level ASIC Design:
- standard cell implementation approach
- logic synthesis tools
- differences between FPGA & standard-cell ASIC design flow

Applications:
- basics of computer arithmetic
- applications from communications, digital signal processing, cryptography, etc.

Platforms & Interfaces:
- FPGA boards
- I/O modules (VGA controller, serial communication modules)
- microprocessor board–FPGA board interfaces (USB, PCIe)

New trends:
- microprocessors embedded in FPGAs (PicoBlaze, ARM)
- using high-level programming languages to design hardware
Tasks of the course

**Advanced course on digital system design with VHDL**
- writing VHDL code for synthesis
- design using division into the datapath & controller
- testbenches

**Comprehensive introduction to FPGA & front-end ASIC technology**
- hardware:
  - Xilinx FPGAs,
  - Altera FPGAs,
  - Library of standard ASIC cells
- software:
  - VHDL simulators,
  - Synthesis tools,
  - Implementation Tools

**Testing equipment**
- oscilloscopes
- logic analyzer
Levels of design description

1. Algorithmic level
2. Register Transfer Level
3. Logic (gate) level
4. Circuit (transistor) level
5. Physical (layout) level

The Level of description most suitable for synthesis is the Register Transfer Level.
Register Transfer Level (RTL) Design Description

Registers → Combinational Logic → Combining Logic → ...
What is an FPGA?

- Configurable Logic Blocks
- I/O Blocks
- Block RAMs
Two competing implementation approaches

**ASIC**  
Application Specific Integrated Circuit

- designed all the way from behavioral description to physical layout
- designs must be sent for expensive and time consuming fabrication in semiconductor foundry

**FPGA**  
Field Programmable Gate Array

- no physical layout design; design ends with a bitstream used to configure a device
- bought off the shelf and reconfigured by designers themselves
FPGAs vs. ASICs

**ASICs**
- High performance
- Low power
- Low cost (but only in high volumes)

**FPGAs**
- Off-the-shelf
- Low development costs
- Short time to the market
- Reconfigurability

Low power
FPGA Design process (1)

Design and implement a simple unit permitting to speed up encryption with RC5-similar cipher with fixed key set on 8031 microcontroller. Unlike in the experiment 5, this time your unit has to be able to perform an encryption algorithm by itself, executing 32 rounds……

Library IEEE;
use ieee.std_logic_1164.all;
use ieee.std_logic_arith.all;
use ieee.std_logic_unsigned.all;

entity RC5_core is
  port(
    clock, reset, enparator: in std_logic;
    data_input: in std_logic_vector (31 downto 0);
    data_output: out std_logic_vector (31 downto 0);
    out_full: out std_logic;
    key_input: in std_logic_vector (31 downto 0);
    key_read: out std_logic;
    out AES.Core;
  );
end RC5_core;

Specification (Lab Assignments)
On-paper hardware design
(Block diagram & ASM chart)

VHDL description (Your Source Files)

Functional simulation

Synthesis

Post-synthesis simulation
FPGA Design process (2)

- Implementation
- Configuration
- Timing simulation
- On chip testing
Simulation Tools

ISim

Active-HDL™

ModelSim®
FPGA Synthesis Tools
architecture MLU_DATAFLOW of MLU is

signal A1:STD_LOGIC;
signal B1:STD_LOGIC;
signal Y1:STD_LOGIC;
signal MUX_0, MUX_1, MUX_2, MUX_3: STD_LOGIC;

begin
  A1<=A when (NEG_A='0') else not A;
  B1<=B when (NEG_B='0') else not B;
  Y<=Y1 when (NEG_Y='0') else not Y1;

  MUX_0<=A1 and B1;
  MUX_1<=A1 or B1;
  MUX_2<=A1 xor B1;
  MUX_3<=A1 xnor B1;

  with (L1 & L0) select
    Y1<=MUX_0 when "00",
    MUX_1 when "01",
    MUX_2 when "10",
    MUX_3 when others;

end MLU_DATAFLOW;
FPGA Implementation

- After synthesis the entire implementation process is performed by FPGA vendor tools
InputFile = c:/Documents and Settings/Milind Parekar/My Documents/ECE_449/ALU/implement/xie0.ini
Executing C:\Xilinx\bin\nt\ngdbuild.exe -p 2S100TQ144-6 -sd "c:\Documents and Settings\Milind Parekar\My Documents\ECE_449\ALU\synthesis" -sd "c:\Documents and Settings\Milind Parekar\My Documents\ECE_449\ALU\compile" -sd "c:\Documents and Settings\Milind Parekar\My Documents\ECE_449\ALU\src" -sd "C:\Program Files\Aldec\Active-HDL 6.2\vlib\SPARTAN2\compile" -uc "ALU.ucf" "ALU.edf" "ALU.ngd"

c:\Documents and Settings\Milind Parekar\My Documents\ECE_449\ALU\implement\verl\revl>set XILINX=C:\Xilinx

c:\Documents and Settings\Milind Parekar\My Documents\ECE_449\ALU\implement\verl\revl>set PATH=C:\Xilinx\bin\nt
Xilinx FPGA Tools

ECE Labs

Xilinx ISE Design Flow

- Xilinx ISim or
- Mentor Graphics ModelSim SE
- Xilinx XST or
- Synopsys Synplify Premier DP
- Xilinx ISE Design Suite (IDE)

simulation
synthesis
implementation

Aldec Active-HDL Design Flow

- Aldec Active-HDL (IDE)
- Xilinx XST or
- Synopsys Synplify Premier DP
- Xilinx ISE Design Suite
Design Process control from Active-HDL
Digilent Nexys3 FPGA Board

- New board used for the first time this semester
- 40 boards purchased by the department
- Distributed to students at the beginning of the semester, collected at the end of the semester
- You may be held financially responsible for any damage caused to your board
FPGA available on the board

Xilinx Spartan 6, XC6SLX16-CSG324C FPGA

- 2,278 CLB slices
- 32 BRAMs (18 kbit each)
- 32 DSP units
- 232 User pins
Why ECE 448 is a challenging course?

• need to refresh and strengthen your VHDL skills

• need to learn new tools

• need to perform practical experiments

• time needed to complete experiments
Difficulties
(based on a student survey)

• finding time to do the labs - 15

• learning VHDL – 2

• getting used to software – 1
Self-evaluation
(based on a student survey)

- 8 – worse than expected
- 3 – better than expected
- 16 – as well as expected
Why is this course worth taking?

- **VHDL** for synthesis: one of the most sought-after skills
- knowledge of state-of-the-art **tools** used in the industry
- knowledge of the modern **FPGA & ASIC technologies**
- knowledge of state-of-the-art **testing equipment**
- design portfolio that can be used during job interviews
- unique knowledge and practical skills that make you competitive on the job market