ECE 448 Team

Course Instructor:  
Kris Gaj  
kgaj@gmu.edu

Lab Instructors (TAs):

Monday and Wednesday sections: 
Umar Sharif  
malik.umar.sharif@gmail.com

Thursday section: 
Rabia Shahid  
rabia.shahid4@gmail.com
A few words about You

- 26 BS CpE
- 7 BS EE
- 2 BS CpE-CS
- 1 BS EE-Phys
Computer Engineering Course Progression

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

- 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
Electrical 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

---

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
Digital system design technologies
coverage in the CpE & EE programs at GMU

**Microprocessors**
- ECE 445 Computer Organization
- ECE 447 Single Chip Microcomputers
- ECE 511 Microprocessors
- ECE 611 Advanced Microprocessors
- ECE 612 Real-Time Embedded Systems

**FPGAs**
- ECE 448 FPGA and ASIC Design with VHDL
- ECE 545 Digital System Design with VHDL
- ECE 645 Computer Arithmetic

**ASICs**
- ECE 431 Digital Circuit Design
- ECE 586 Digital Integrated Circuits
- ECE 681 VLSI Design for ASICs

**ECE 699 Software/Hardware Codesign**
Course Hours

Lecture:
Monday, Wednesday
1:30-2:45 PM, Robinson Hall A, room 412

Lab Sessions:
Monday                             Wednesday, Thursday
4:30-7:20 PM                     7:20-10:00 PM
The Nguyen Engineering Bldg., room 3208

Lab sessions will start next week!!!
General 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 meeting of another section, but give preference in access to the lab computers to the students attending their own lab 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.
Office Hours

You are welcome to attend all office hour sessions!
You can direct your questions regarding lab assignments
to the TAs and myself

Do your best to avoid “chasing” the TAs outside of their
office hours! They have other jobs to do!

Umar Sharif
• Mon, 12:00-1:00pm, Wed, 4:00-5:00pm, ENGR 3204
• Thu, 5:00-7:00pm, ENGR 3231

Rabia Shahid
• Tue, 4:00-6:00pm, ENGR 3203
• Thu, 2:00-4:00pm, ENGR 3204

Kris Gaj
• Mon, 3:00-4:00pm, Wed, 3:00-4:00pm,
• Thu, 6:00-7:00pm, ENGR 3225
Getting Help Outside of Office Hours

piazza

• System for asking questions 24/7
• Answers can be given by students and instructors
• Student answers endorsed (or corrected) by instructors
• Average response time in Spring 2014 = 18 minutes
• You can submit your questions anonymously
• You can ask private questions visible only to the instructors
Lab Access Rules and Behavior Code

Please refer to

ECE Labs website

and in particular to

Access rules & behavior code
# Course Web Page

http://ece.gmu.edu/coursewebpages/ECE/ECE448/S15

<table>
<thead>
<tr>
<th>Organization</th>
<th>Lecture</th>
<th>Lab</th>
</tr>
</thead>
<tbody>
<tr>
<td>Instructor</td>
<td>Syllabus</td>
<td>Lab Syllabus</td>
</tr>
<tr>
<td>Teaching Assistants</td>
<td>Textbooks</td>
<td>Rules</td>
</tr>
<tr>
<td>Lecture and Lab Time</td>
<td>Lecture Slides</td>
<td>Lab Assignments</td>
</tr>
<tr>
<td>Office Hours</td>
<td>Homework</td>
<td>Lab Slides &amp; Examples</td>
</tr>
<tr>
<td>Grading</td>
<td>Past Quizzes</td>
<td>Software</td>
</tr>
<tr>
<td></td>
<td>Past Midterm Exams</td>
<td>Hardware</td>
</tr>
<tr>
<td></td>
<td>Past Final Exams</td>
<td>Useful References</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Past Lab Exams</td>
</tr>
</tbody>
</table>
Grading criteria

First part of the semester (before the Spring break)

Lab experiments - Part I
16%

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
24%+2%

Quizzes & homework: 5%

Final exam
25%
# Tentative Grading Scheme for the Labs

<table>
<thead>
<tr>
<th>Lab 1: Developing VHDL Testbenches</th>
<th>4 points</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lab 2: Combinational &amp; Sequential Logic</td>
<td>4 points</td>
</tr>
<tr>
<td>Lab 3: State Machines. Basic I/O Devices.</td>
<td>8 points</td>
</tr>
<tr>
<td><strong>Total:</strong></td>
<td><strong>16 points</strong></td>
</tr>
<tr>
<td>Lab 4: VGA Display</td>
<td>8 points</td>
</tr>
<tr>
<td>Lab 5: Computer Graphics</td>
<td>8 points</td>
</tr>
<tr>
<td>Lab 6: PicoBlaze System</td>
<td>8 points</td>
</tr>
<tr>
<td>Lab 7: Logic Analyzer (in class)</td>
<td>2 bonus points</td>
</tr>
<tr>
<td><strong>Total:</strong></td>
<td><strong>24+2 points</strong></td>
</tr>
</tbody>
</table>
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:**

<table>
<thead>
<tr>
<th>Lab 4: VGA display (2 weeks)</th>
<th>– 8 points</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lab 5: Computer Graphics (2 weeks)</td>
<td>– 8 points</td>
</tr>
<tr>
<td>Lab 6: PicoBlaze System (2 weeks)</td>
<td>– 8 points</td>
</tr>
<tr>
<td>Lab 7: Logic Analyzer (in class)</td>
<td>– 2 bonus points</td>
</tr>
</tbody>
</table>

**Total:** 24+2 points

**Schedule B:**

<table>
<thead>
<tr>
<th>Lab 4: VGA display (3 weeks)</th>
<th>– 8 points</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lab 5: Computer Graphics or Lab 6: PicoBlaze System (3 weeks)</td>
<td>– 8 points</td>
</tr>
<tr>
<td>Lab 7: Logic Analyzer (in class)</td>
<td>– 2 bonus points</td>
</tr>
</tbody>
</table>

**Total:** 16+2 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 1-3)
- An open-ended project proposed by students, the TAs, 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
Bonus Points for Class Activity

• Based on class exercises during lecture and lab sessions, as well as your activity on Piazza

• “Small” points earned each week posted on BlackBoard

• Up to 8 “big” bonus points

• Scaled based on the performance of the best student

For example:

<table>
<thead>
<tr>
<th></th>
<th>Small points</th>
<th>Big points</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Alice</td>
<td>40</td>
<td>8</td>
</tr>
<tr>
<td>2. Bob</td>
<td>35</td>
<td>7</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>26. Charlie</td>
<td>10</td>
<td>2</td>
</tr>
</tbody>
</table>
Exams

- Midterm Exam for the Lecture – 10 points
  Wednesday, March 4

- Midterm Exam for the Lab (hands-on) – 15 points
  Monday or Thursday, March 2 and 5
  Wednesday section students can write the exam on Monday or Thursday in order to avoid taking Midterm Exam for the Lecture on the same day

- Final Exam – 25 points
  Wednesday, May 6, 1:30-4:15pm
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 testbenches
- writing synthesizable RTL level code in VHDL

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, computer graphics, 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,
  - Library of standard ASIC cells
- software:
  - VHDL simulators,
  - Synthesis tools,
  - Implementation Tools

Testing equipment
- oscilloscopes
- logic analyzer
VHDL for Specification

VHDL for Simulation

VHDL for Synthesis
Levels of design description

- Algorithmic level
- Register Transfer Level
- Logic (gate) level
- Circuit (transistor) level
- Physical (layout) level

Level of description most suitable for synthesis
Register Transfer Level (RTL) Design Description
What is an FPGA?

Block RAMs

Configurable Logic Blocks

I/O Blocks

Block RAMs

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
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_unsigned.all;

entity RC5_core is
    port(
        clock, reset, encr_decr: in std_logic;
        data_input: in std_logic_vector(31 downto 0);
        data_output: out std_logic_vector(31 downto 0);
        out_full: in std_logic;
        key_input: in std_logic_vector(31 downto 0);
        key_read: out std_logic;
    );
end AES_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™
Complete FPGA Verification Environment

ModelSim®
FPGA Synthesis Tools

Xilinx®
XST

Synopsys®
Synplify® Premier
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 Parelkar/My Documents/ECE_449/ALU/implement/xie0.ini
Executing C:\Xilinx\bin\nt\ngdbuild.exe -p 2S100TQ144-6 -sd "c:\Documents and Settings\Milind Parelkar\My Documents\ECE_449\ALU\synthesis" -sd "c:\Documents and Settings\Milind Parelkar\My Documents\ECE_449\ALU\compile" -sd "c:\Documents and Settings\Milind Parelkar\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 Parelkar\My Documents\ECE_449\ALU\implement\verl\revl>set XILINX=C:\Xilinx

c:\Documents and Settings\Milind Parelkar\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)

Aldec Active-HDL Design Flow

- Aldec Active-HDL (IDE)
- Xilinx XST or
- Synopsys Synplify Premier DP
- Xilinx ISE Design Suite

- simulation
- synthesis
- implementation
Design Process control from Active-HDL
Digilent Nexys3 FPGA Board

• Used for the first time in Spring 2013

• 40 boards purchased by the department

• Distributed to students at the beginning of the semester, collected at the end of the semester

• Treat with care! 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

Block RAMs

Configurable Logic
Block slices (CLB slices)

Programmable Interconnects
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
16 – as well as expected
3 – better than 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