ECE275 Lab Final Project: Pong game

Pascal Francis-Mezger
Vikas Dhiman

November 17, 2021

Final Project Overview

The final project consists of a design and implementation for the Atari pong game[1] FPGA project. You should use both what you have learned in class, labs, and additional functionality you can find in reference material, manuals, and documentation. You will do the project in a team of two. The goal of the project is to utilize Verilog to create higher level constructs and demonstrate your understanding of the base material we have covered.

Final Project Requirements

1. Must consist of substantially your own created work. Utilizing supporting code to realize your design is acceptable, but it must be *clearly demarcated* what is your code and what is supporting code you have integrated. You must clearly state the source of supporting code. Failure to follow this practice will count as academic dishonesty. Ideally, your code and supporting code will be part of different modules, but you can also borrow supporting code and comment which lines of the code are yours and what were borrowed along with the source.

2. Must integrate some functionality into your project that we have not explicitly covered in class. This could be more advanced Verilog functionality, integrating VHDL modules, utilizing some hardware on the FPGA we have not yet utilized, integrating external hardware/sensors, etc.

3. Must provide a clear design goal before implementation. It is expected your design will change as you work through implementation, but you need clearly defined goals before starting. This should consist of a block/wiring/hardware diagram, and a short writeup and/or psuedo-code describing the functionality. In your final writeup you should describe any differences between your initial goals and your finished project, and what led you to make those changes.

4. Must provide documenation of the finished project. This should consist of your initial design documents, a detailed description of the functionality of your final project, a listing of all of your code, a description of your code, and a brief conclusion covering any changes to the final project from your initial design and any future modifications you would like to make to the project. The description for the code at minimum should describe the functionality of each module, the purpose of the input/outputs for the modules, and broad strokes of how your code in the modules accomplishes your functionality.

Project design and planning document (due Nov 22)

The project design and planning document will consist of a PDF file and a Verilog file for each module. The Verilog files for unimplemented modules are expected to contain input, output and documentation.

[1] ponggame.org

1. What do you understand by the pong game implementation? What will be the final product? In a proposal, you should not specify a single outcome, but a range of projects, from minimal to ambitious. The minimal outcome should make sure that you have something to show, while ambitious outcome will be perfect project of your dreams. You should develop the timeline by making minimal outcome ready as early as possible and then work towards adding more features towards the ambitious outcome. For example
    - (Starting point): Reproduce the VGA interface demo, from DE0_demonstrations.
    - (Minimal): A moving ball demo.
    - (Stage 2): A bouncing ball demo (reacts with edges of screen)
    - (Stage 3): A ball that bounces off a paddle
    - (Stage 4): Control the paddle with human input
    - (Ambitious): Insert your crazy idea here.
2. How can each outcome be tested? What is the expected range of inputs? What is the expected output for each input?
3. Go over the Altera cookbook and demonstrations to understand what capability is available.
4. What are the input and output ports, switches, LEDs on the FPGA board that you are going to use. Are you going to develop and implement your own protocol or use existing modules for interacting with the FPGA board?
5. How will the project be broken down into smaller modules? The minimum number of modules is three: input, core, and output. The core of your implementation can be further broken down into smaller modules. Each module's signature and documentation (what it is expected to do) will be specified at this stage.
6. Which modules will be borrowed from external sources like Altera cookbook or the internet? Which modules will be programmed by you? How will you automatically test the modules programmed by you and those found from unreliable sources?
7. Anticipated timeline for the project, including research on finding the desired modules; learning new Verilog features; evaluating, adapting, testing and debugging the existing modules; anticipated man-hours for each step. Since the project is spread over 3-weeks, we expect you to spend total 5-7 hours per week per person. Since, you are new to Verilog, it is okay to anticipate your time in terms of C/C++ and then multiply it by 2.

Project progress report (due Dec 1st)

Expectations for the project progress report will evolve and will be more concretely defined by Nov 22.

1. Module code with automated testing fixture for the modules that have been implemented. Each module, successfully implemented (passes your own specification of the module), will earn you grades towards the final project.
2. Updates to the project proposal timeline. Scaling up or scaling down the project expectations.
3. Sticking points and challenges. Things to try next to address the challenges.
4. A 10-min Q&A will be scheduled with you to evaluate your understanding of the Verilog code that you wrote. The Q&A for each team will be scheduled only once between Dec 1st to Dec 10th.

Final Project report, demo and Q&A

1. A report and video demo of your project is due on before Dec 10th.
2. Sticking points and challenges. Reflections on your planning document. How could you have planned better? What are the future ideas you will like to try?
3. A 10-min Q&A will be scheduled with you to evaluate your understanding of the Verilog code that you wrote. The Q&A for each team will be scheduled only once between Dec 1st to Dec 10th.

Grading

The main purpose of the project is to create an opportunity for you to learn, apply, and demonstrate your understanding of Verilog.

1. Your grade will depend on the Verilog modules successfully implemented, their complexity, and a final project Q&A session about your implementation (total 30% of the final grade).
2. Project proposal (5% of the final grade).
3. Project progress report 1 (10% of the final grade).
4. Final report (15% of the final grade).