ECE1373S Project Requirements and Report

Paul Chow

May 5, 2004

1 Project Demo

About a week after the report is handed in, I'd like to have a demo of your project by the group. The delay is to give me a bit of time to read the report first.

Your demo should include the following:

- overview of your design environment – The structure of your overall design environment is important
- file system structure
- how you did your testing/verification, both the original code and the running system
- example of using TestBuilder
- downloading your system
- running your system

2 Report

The rest of this is a guideline to help you organize your report with a list of the major sections that you should include. You should follow the structure as outlined as much as possible.

The LaTeX source for this document is available at

`www.eecg.toronto.edu/~pc/courses/vlsi/2004/`

if you want to use it as a starting point.

I would like the reports by Friday, June 4, 2004 (negotiable, but this is a good target).

3 Introduction

This is an introduction to the design problem. You do not have to be too detailed, just give a basic overview of what you are doing, and cite the appropriate references. This is mainly so that your report can stand on its own.

Some higher level block diagrams would be useful to describe the overall functionality. This is probably useful for later when you need to talk about parts of the design.

4 Current Status

Briefly give the current state of the design.

- What has been done.
- What remains to be done.

5 Architectural Design

- discuss the steps/process you took to define the overall architecture and come up with the initial specifications that you used to start your design implementation
- If you were given some documents to work from, you should include them as an appendix and refer to them here.
  - assumption here is that the specs may have later changed, but talk about what you did to arrive at your original specs
- describe the architecture you developed

6 Specifications

- Outline the specifications you developed.
- What problems did you have with the specifications as the design progressed?
- What changes did you later make to the specifications and why?

7 Methodology

Describe your methodology of design and design strategy. How did you attack the problem? Include here descriptions of any tools you had to develop.

7.1 Design Environment

- describe your design environment here
- tools used
- file structure/organization
- source code control
- how you handled multiple users accessing files
- etc.

7.2 Partitioning

- How did you partition the design?
- Why did you do it this way?

7.3 Physical Design Strategy

Describe your layout and circuit design strategy, if any, i.e., did you do any floorplanning or anything like that to get better performance in your design?

7.4 Simulation and Verification

- Describe your simulation, and verification strategy.
- How did you verify the operation of your HDL modules?
- Indicate, if any, possible holes in your strategy that may not cover certain problems.

8 Contributions

Each member should contribute a section here (labelled with their name) that briefly describes what they did in the project. Be sure to include all things for the course, such as setting up tools, scripts, etc. Also, indicate what code you wrote and where I might look to see what it looks like.

9 Design Characteristics

9.1 Architecture

Give some more detailed descriptions of the chip architecture.

- What are all the pieces? It would be helpful to indicate which HDL modules correspond to these pieces.
- Briefly what do they do.
- How do they go together?

9.2 Chip Characteristics

- utilization of chip(s) – PAR report would be sufficient

9.3 Computing System

- comments on requirements of the computing system
- disk utilization – how much disk did you need
- CPU requirements – memory requirements and performance

- How long did typical jobs take (synthesis, P&R, etc.) for various aspects of the project. Give a table if you can and specs on the machine. You don't have to be really accurate (approximate hours is fine).
- anything else?

9.4 Where the Time Went

If you need I can lend you the progress reports so that you can get the actual numbers.

- Give an analysis of how you spent your time.
- What were the major bottlenecks in your design.

10 Problems

Discuss problems you had.

- Were there management problems?
- Were there interfacing problems (between partitions)?
- Did a particular tool cause a problem? Which one? What was the problem?
- Was there just not enough time, not enough people?
- Known problems in the design and suggest fixes.
- Anything else?

11 Conclusions, Suggestions, Comments

- What do you think you got out of the course, if anything?
- What did you learn about systems design?
- Comments on the format of the course – likes and dislikes
- Suggestions for improvement
- Anything else you care to say?

Appendix: Initial Descriptions

Original handouts that were given to you about what to build.

Appendix: Design Specifications

This should be like a spec sheet about the chip and how it is to be used. There should be enough information for someone who is taking over the design to be able to figure out what you did.

Appendix: Testing Strategy

- You should describe the procedure for testing the design.
- You should give the various tests that are to be used, where to find them, what each test is testing, what is to be expected.

Appendix: Documentation

Where to find the files, and your directory structure. This should provide enough documentation so that someone else can find all your stuff and continue with the project.

Appendix: Schematics and other Details

Include here any schematics and other details about your design that you feel would be appropriate for anyone that needs to understand the details of what you did.

Appendix: Hints for the Next Time

Please provide any general comments that you feel would be helpful to future users regarding the tools, design environment, computing environment, disk space required, etc.