Digital Systems Design
2017
Project 1 – Computer and RPN Calculator

Professor Jonathan H. Manton

Aims

1. To learn how to design your own computer from first principles.
2. To learn how to program your own computer in machine code.

Digital designs often include a microprocessor. Perhaps the fastest way of becoming familiar with microprocessors, and how to program them, is to build a simple microprocessor from scratch.

Prior Preparation

Start as early as you can!

Be familiar with how to use ModelSim to debug your design.

If you have not had the pleasure of owning your own RPN calculator, find out how an RPN calculator works. (RPN calculators are normally faster to use than ordinary calculators. They are also simpler to implement because there is no need for brackets.)

Broad Outline

The hardware side consists primarily of designing a module called CPU that implements a simple computer. The CPU will interface to the switches, pushbuttons, LEDs and 7-segment displays on the DE1-SoC board via auxiliary modules. This is Part 1 of the project.

Part 2 of the project will be the implementation of a Reverse Polish Notation (RPN) calculator: you will program (in machine code) the computer you made in Part 1 to function as an RPN calculator.

Remark

In the real world, specifications are unlikely to be perfect. If there is vagueness in any of the specifications, it is ultimately up to you to decide the exact functionality. Your decisions should be guided by two factors: which functionality is sensible, and which functionality is easier to implement.

Co-operation

The aim of this project is for you to learn. Discussing your project with others will facilitate the learning process and you are encouraged to discuss freely. Needless to say, co-operation is very different from copying: generally very little learning takes place when copying. (You must also comply with University regulations, including those concerning plagiarism.)

Project Report

You are to write a project report in conjunction with developing a Quartus Project. The project report should serve as a record of how you went about your project: the reasons for the design decisions you made, the tests that you carried out on your design, and anything else of relevance.

You can write in dot point form to save time, and you can cut-and-paste screen shots.

The report is not meant to slow you down, but rather, to help you remember what you did.

You are also encouraged to consult the Marking Scheme when deciding what information to put into your report.

A relatively short but informative report is ideal.

Assessment

You will be assessed three times. (Note the weeks are in "Lecture Weeks" and not "Lab Weeks", so Week 12 is the final week of lectures.)

Week 10	5 min (2 marks)	During designated lab session
Week 11	5 min (2 marks)	During designated lab session
Week 12	10 min (6 marks)	During designated lab session OR in after-hours lab

It is your responsibility to ensure you have booked timeslots with your lab demonstrator for these three weeks.

For each marking session, you should be ready for your demonstrator to look at your project report and your Quartus project (including the configured DE1-SoC board). Your demonstrator will look at your report and project, and ask questions to test your understanding. Assessment is largely based on the degree of understanding you have.

For the final marking session (Week 12), you must also have ready for your demonstrator the following material.

1. A zip file (e.g., on a USB memory stick) containing
    a. Your project report (in PDF format).
    b. Your Quartus project.
2. A written statement (printed out on paper) of the number of marks you believe you should receive for this part of the assessment (out of 6), and why, with reference to the Marking Scheme below.

The name of the zip file should include your surname and student id.

Your report should contain a discussion of the design decisions you made, and evidence of tests that you performed along the way.

Note that since time is limited, any delay in answering questions or transferring the files to the demonstrator might cause marks to be deducted. (Marks are awarded based on the positive aspects you show to the demonstrator; if there is not enough time to show the demonstrator your full project, it will not be possible to be awarded full marks.)

Marking Scheme

Since the aim of this project is for you to learn, you will be judged primarily on your level of understanding. You will receive close to full marks if you have a good understanding of most of the project but your final design does not work, while if you have a functioning RPN calculator but are unable to answer questions about how your design works you may receive only very few marks.

Week 10

Marks	Requirement
1	Understanding of timing issues (debouncing, synchronisers etc)
1	Progress to date (based primarily on the level of your understanding and the quality of your code and report)

At the very least, you should have completed Stages 1 to 4 of the project, although ideally you will have progressed considerably further (e.g., Stage 8), for otherwise you will have difficulty finishing the project on time.

Week 11

Marks	Requirement
1	Understanding of Instruction Pointer, registers and instructions
1	Progress to date (based primarily on the level of your understanding and the quality of your code and report)

You should aim to have completed Part 1 of the project (Stages 1 to 12).

Week 12

Marks	Requirement
½	Fully functional computer
½	Fully functional RPN calculator
2	Understanding of the microprocessor architecture
2	Understanding of how the machine code works
1	Overall quality of Verilog code and project report

You should have completed your project (Parts 1 and 2) and report.

Your project report should demonstrate your level of understanding: if you know how to design and implement a program, then your report should document the process you went through in creating the RPN calculator, for example.

In Week 12, it is up to you to convince the demonstrator, in writing[1], that you have met the objectives. For example, if you claim that you have a fully functional computer, but have not succeeded in writing an RPN Calculator, then you will need to have implemented a program that tests out the various aspects of your computer and/or have included comprehensive test bench results. Remember time is limited during the marking session, so plan ahead.

[1] If your verbal responses are too verbose and vague during the marking session then your marks will suffer: prepare your case beforehand then let the demonstrator check your documented evidence.