Lab Reports for Electronics for Embedded Systems

Only one report is required for each group. Please write your names and ID numbers in the first page of the report. To submit the report, please upload a PDF file to the "Elaborati" section of the web site of the course on the student portal. Name your file as "`group`*`Xyy`*`_lab`*`n`*`.pdf`" (example: `groupB05_lab1.pdf`). Groups of Monday have the letter X=A followed by the desk number, groups of Tuesday have the letter X=B followed by the desk number. If you are including screenshots, please resize them so that they don't make the pdf file too large. The report should normally be no less than 4-5 pages, and no more than 8-10 pages. If you want, you can also handle me a printed version in addition to the upload. Reports of the first 3 laboratory sessions are due by Christmas, while the last 2 reports are due one week before the first exam in the January/February session.

What to write in the report for the first laboratory: this session is mainly to learn how to use Quartus Prime for FPGA programming, while at the same time see the effects of various parameters on the final result of the logic synthesis process. The report should be a sort of diary of your lab session, and should enable the reader to repeat your experiments and obtain the same results.

Among the most important results to show, there is the timing analysis data. For the AND gate you can list the obtained worst case t_{pd} delay and the final placement of the logic element in the floorplan in various combination of the following assignments:

- With or without pin assignment (when not using pin assignment, check which pins were selected by the tool)
- With or without timing constraints (maximum delay, which I call t_{pd} in this text)
- With or without a placement assignment (with drag&drop on the chip planner)

When using a fixed pin assignment, start with a loose (i.e. high) timing constraint, and then decrease it till you find one for which the timing analysis fails (i.e. at least a constraint was not satisfied), and whether it is only for the Slow model or for the Fast model, as well. In the case of the AND gate you can try several of the above combinations, while for more complex circuits it might be too long (too many pins to change, for instance), so include less cases. If you want to see very large delays, try assigning pins of the AND gate to three different sides of the FPGA, so that they are far, and drag the logic element close to the fourth side, so that it is far from all pins; the long interconnections will cause very long delays. For instance, I managed to get delays as low as 5.5ns to as high as 15ns, by just playing with the pin assignment and the location assignment of the logic element. For sequential circuits, instead of the t_{pd} delay you should check the maximum clock frequency f_{max} and see if setup times are satisfied. Note that violations of timing constraints are shown in red in the Compilation Report under the Timing Analyzer folder.

You have certainly written some VHDL files. You can include them entirely in the report, or <u>better only relevant parts</u> of them. If you include code, use a smaller font compared to normal text. Another important part to include is the outcome of simulation. It might be useful to show the code for the testbench along with the screenshot of the simulation.

Always check the Floorplan of your design, although for small designs it is not really significant. However, enabling Routing Details (see the Layer Settings, you must zoom in a little bit to actually show routing, and you may want to disable Unused Resources to show only those that are actually used) allows understanding why certain solutions are slower than others, due to the much greater length of interconnections. You can even estimate the wire length by looking at the floorplan for certain connections.

Whenever you include some code, a table, a picture or a screenshot, be sure to refer to it in the text and explain why you include it and what information it delivers. ***<u>The best thing to do is to put ALL tables and figures at the end of the report</u>***, number them and refer to them in the text. The text should be readable without looking at the figures. Both in the reports and in the written exam, you think that a picture is worth a thousand words, and therefore you write 0 words to explain it. That's really bad in technical writing, you must really write those thousand words (or at least some of them). Having all figures and tables at the end also allows you to see how much text you wrote: if you are left with a single page of text, then you probably need to write more.

For the actual hardware part, you don't have probably much to say, other than if it worked or not. You may have a picture of the working board, but if you state that it worked without showing a picture I will (probably) believe you. Please explain any strange behaviour of the hardware. If you take measures using the oscilloscope, then you can include

a picture of the waveforms (you can save pictures to an USB memory plugged into the oscilloscope, or take a picture with your smartphone). Sometimes you make a video of what happens in the lab (or at home); if you want you can put a link to the video and I will go and look at it when evaluating the report.

For the adders, timing analysis shows the delays due to the combinational circuit, as long as delays due to interconnections and pads, and can often be misleading (i.e. the ripple carry adder may look faster than carry look ahead). The suggestion is to put the entire adder between registers, making a sort of pipeline stage, like in Figure 1 (which shows a 4-bit ripple carry adder, from a top level view on the left, and a more detailed view on the right). Now the maximum propagation delay within the adder determines the minimum clock period, and hence the maximum clock frequency. So check the clock frequency in the timing analysis, as well.

Please feel free to drop me an email if you have any question regarding the lab session and the report.

Claudio Passerone

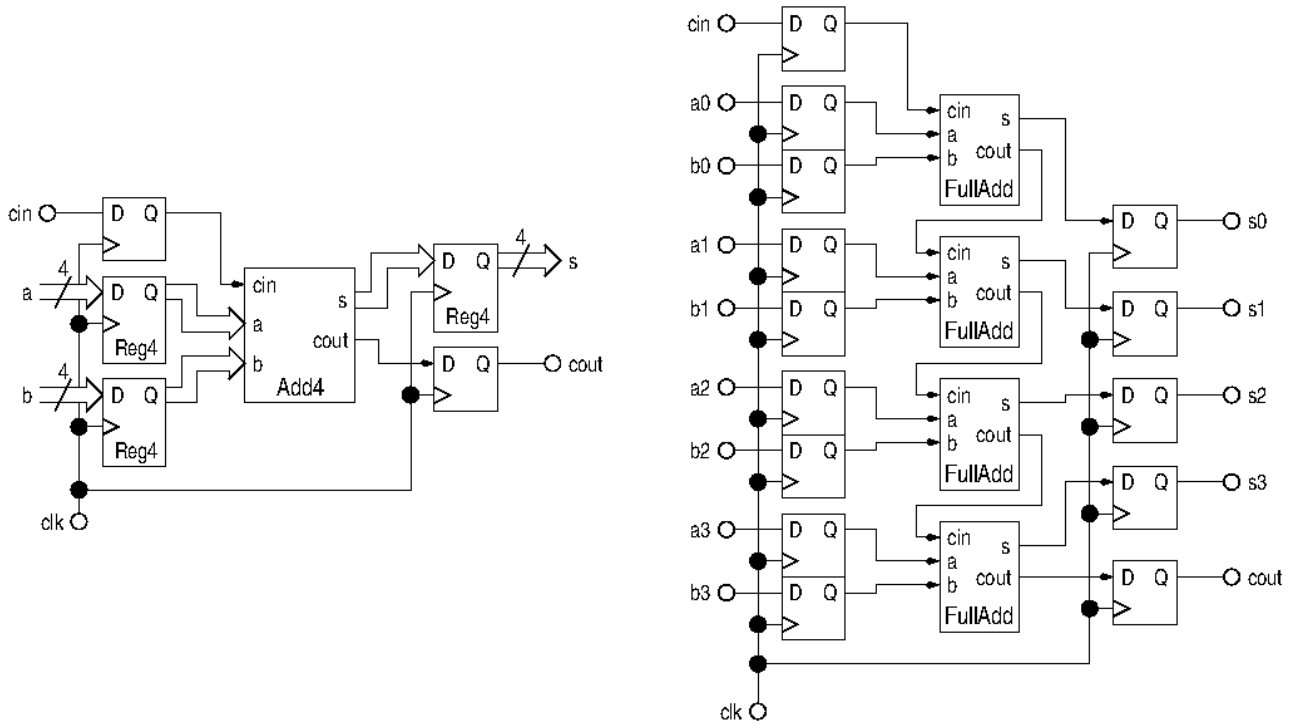

Figure 1: the 4-bit adder between input and output flip-flops, like in a pipeline stage; the circuit on the right is a more detailed representation of the circuit on the left