Digital Design and Computer Architecture: Lab Report		
Lab 5: Implementing an ALU		
Date		Grade
Names		
		Lab session / lab room

You have to submit this report via Moodle.

Use a zip file or tarball that contains the report and all other files you used for the report, i.e., the entire Verilog project folder and/or all schematics you drew. If any files are missing, it may negatively impact your grade. No shortcuts/links will be accepted.

Only one member from each group should submit the report. All members of the group will get the same grade.

The submitted file's name should be *Lab5_LastName1_LastName2.zip* (or *.tar*), where *LastName1*, *LastName2* are the last names of the members of the group.

The deadline for the report is a hard deadline and it will not be extended.

Exercise 1. Replacing the Adder with your Adder from Lab 2

In the lab Manual, you learned how to design an ALU and how to find out about its area. In this exercise, instead of using + for the adder in your ALU, **use the adder you designed in Lab 2 (i.e., with explicit gate-level description)**. To do so, you first need to build a 32-bit adder using the instances of the 4-bit adder you have built. Notice that you have to come up with such a 32-bit adder module to get points.

As you read in Chapter 5 of H&H book, hardware description languages provide the operation to specify a carry propagate adder. Modern synthesis tools select among many possible implementations, choosing the cheapest (smallest) design that meets the speed requirements. This greatly simplifies the designer's job.

Compare the area of the ALU in this exercise with the area you obtained from the ALU you designed in the manual and write about your conclusions based on this observation.

Note: For lab 6, you need to keep using the ALU you designed in the manual, not in this report.