General notes for labs. The labs are intended to give you the opportunity to apply what you’ve learned in the course to larger problems and to gain some experience with the kind of CAD tools used by professional digital systems designers. You should strive to find the best possible solution for the given problem and to make your solutions complete, well-documented and easy for someone else to understand. The work you are asked to turn in will often include design notes, providing background explanations needed to understand your design, in addition to the design itself. These should not be just rough notes, but carefully written explanations of what you have done and why. Block diagrams should be well-organized, with all inputs and outputs labeled and components arranged so that it is easy to see what is connected to what. VHDL code should be organized neatly, with appropriate use of indenting, mnemonic signal names and comments to explain various sections of code. Simulation output should be complete and should include written comments that explain clearly how the simulation demonstrates that the circuit works correctly at every step (or if it does not work correctly, you should point that out and explain what you did to try to resolve the problem – be assured that the TAs and I will notice if your simulation results are incorrect). Unusual simulation output should be noted and explained. The labs are your opportunity to demonstrate that you have really mastered the course material and can apply it to address creative design challenges. You will be judged on the professionalism you demonstrate in the work that you turn in, not just on whether you’ve met the minimum requirements for the problem.

Individual SVN repositories have been setup for each of you. These include separate folders for each lab and studio. Please see the instructions in the left margin of the web site for instructions on how to access the repository. All files for this lab can be found in the lab1 folder in your repository. Use Project Navigator to create a new project called lab1, within the lab1 folder. Add all the provided source files to your project. You will be prompted for an “association” for each source file. The two testbench files should be given a “simulation-only” association. All others should have been given an “all” association. You will modify some of these files using Project Navigator as you work on the lab. Change only the files that you are instructed to. When you are finished, be sure to commit your changes back to the repository. Changes must be committed no later than 6:00 pm on the due date for the lab. In addition to the VHDL source files, you will find a word document that contains a template for your lab report. The template includes detailed instructions for what you should turn in. You should include your report in the committed repository, but you are also required to hand in a paper copy. The paper copy must be securely fastened together, preferably with a single staple in the top left corner. Your name should appear on the first page.
In this lab, you will be modifying the calculator circuit that was discussed in class, so that it will detect arithmetic error conditions and display an appropriate error indication when errors occur. You will be adding two signals to the calculator interface. The mode input should be added right after the add input in the entity declaration. It is a std_logic signal which determines if additions are considered to be unsigned or signed. When mode=0, operations are treated as unsigned. When mode=1, they are treated as signed. The mode affects how arithmetic errors are detected. An unsigned addition results in an arithmetic error if it produces a sum that is too large to fit in the register used by the calculator (the calculator uses a 16 bit register). A signed addition results in an arithmetic error if the two operands have the same sign, but their sum has a different sign.

Whenever an error is detected, the calculator should set an internal error bit. The error output should be equal to the value of this error bit. The error bit should be cleared when a clear operation is performed and no other operations should be allowed to occur while the error bit is set.

The output module should be modified to include an error input. Whenever the error input is high, the output module should display dashes in the bottom row, where the calculator output would normally appear.

You are also required to modify the top circuit to handle the signals added to the calculator and output module. The calculator’s mode input should be connected to swt(0) so that you can change the mode by changing the switch. The calculator’s error output should be connected to the output module’s error input. The mode and error signals should also be connected led(0) and led(1), so that their values can be observed on the board’s LEDs as well.

You will find more details in the lab report template.

For this lab, you are expected to work as individuals. While you can discuss the lab in general terms with your classmates, you are not permitted to collaborate in any way. Sharing of VHDL code, design notes or simulation results is explicitly prohibited. Students found violating these rules will have the full value of this lab deducted from their class scores.