BSUCA12 High Performance Adder

Must be submitted electronically before Midnight on Friday, March 16

In this part of the project, you will implement a 16-bit 2's complement adder. This adder will be one of the components used in your implementation of the ALU unit. The adder design should use a two-level carry-lookahead (CLA) technique as discussed in class. The adder takes a single-bit input (`subtract`) which, if true, denotes that subtraction should be performed instead of addition (`sum = inputA-inputB`). The Figure below illustrates the adder, including the exact signal names that you must use in your design (to facilitate testing). Note that the adder also produces the result of logical AND and logical OR; these results (`AandB, AorB`) are produced as part of addition, when computing the propagate and generate signals (i.e., you don't need to add extra AND and OR logic to your adder). If the `subtract` signal is high, then the values of `AandB` and `AorB` are 'don't cares' (i.e., you don't have to worry about them). After designing your adder, you must test it thoroughly to demonstrate that it works correctly. Your test should show me how fast your adder can produce the results. Use a gate level delay of 5 ns to calculate the critical path delay. Accordingly, do not ignore slow circuit paths through your adder. The high level module must be named `claAdder` and file name must be `claAdder.v` Names of lower-level components are unrestricted. Requirements for submitting this part of the project as you did in part 01.