|  |
| --- |
| EE/CprE 465 -- Fall 2014Lab 1: Function Design and Simulation The first lab is a warm-up lab, you will perform Verilog coding of a mathematical function circuit and Verilog function simulation with ModelSim. The objective of this lab is to give you hands-on experiences with:   1. Verilog coding 2. Test bench construction 3. Function simulation   The target mathematical function is:  oRESULT= iA0\*iB0+iA1\*iB1+ iA0\* iA1\* iB0\*iB1 when iSEL=0  oRESULT= iA0\*iB0+iA1\*iB1 when iSEL=1  where signals starting with *i* stand for input and those starting with *o* stand for output.  The inputs and outputs are defined below:   * iCLK — input clock signal, 50MHz * iRST\_N — input reset signal, active Low * iA0, iB0, iA1, iB1 — 8-bit binary unsigned integer input signals * iSEL — input selection signal * oRESULT — 17-bit binary unsigned integer output signal   Note that the final result may be more than 17 bits but we ignore any overflow issue in this lab. The whole circuit should be like this:    To launch ModelSim, please copy the associated file, ModelSim\_env.txt, to your home directory and source it. Then type vsim to launch it. To learn to use ModelSim, you may find information at the ModelSim tutorial posted. In step 12 of the tutorial, please make sure "Enable optimization" is un-marked.  The ModelSim tutorial already has an example of Verilog code and its testbench. You may follow it to learn Verilog coding and simulation. Please ask your TA for further questions. If you are new to Verilog, please find another 2 sets of Verilog codes and corresponding testbenches. The first one is Serial\_parallel.v and Serial\_parallel\_tb.v, which describes a parallel-in and serial-out module. It is the easier one. The second one is a controller (control.v and control.mif) and its testbench (t\_control.v). This controller takes a 6-bit opcode and translates it into 9 bits of control signals. The translation from the opcode to the control signals is stored in a memory, which is initialized by a memory initialization file (.mif).  **Lab Tasks:**   1. Verilog coding: Implement the circuit shown above using Verilog. You may use functional description or structural description. Please report your code. 2. Testbench building: Build a testbench to verify your design in ModelSim. Please take a look at the posted testbenches for reference. Furthermore, please use as many of the following functions as possible in your testbench to make it more automatic.   $display, $monitor — display and monitor the simulation process   1. Function simulation: Please report your simulation result. |