UNIVERSITY OF TEHRAN
Electrical and Computer Engineering Department
Digital System Design with Hardware Description Languages
Fall 1392 / Test 1

Computer Account#_____

First Name:_____

Last Name:_____

Number:_____

Signature:_____

Grade:

Problem 1. _____/60

Problem 2. _____/40

Total: _____/ 100

THIS IS AN OPEN BOOK OPEN NOTE EXAM
YOU HAVE EXACTLY TWO HOURS AND A HALF TO DO THIS TEST
YOU MUST SHOW COMPLETE WORK ON ALL PROBLEMS

1. **MIXED LEVEL MODELING.** Several components that are described at the RT level (RTLC) are to share a bussing system based on their priorities. Each device is identified by its 4-bit generic parameter (i). When such a device (RTLCi) is to use the bus, it first looks up its priority from a priority Dispatcher circuit. It then passes the priority to the arbiter and the arbiter issues the id of the granted device. The output from the arbiter also selects the bus on behalf of the device that is granted the use of the bus.

   The Priority Dispatcher circuit has Pi inputs from various RTLCi circuits and assigns a 4-bit priority to the requesting device. The priority assigned by the Priority Dispatcher is based on usage, time of request, and several other parameters that are not of our concern here. When a Pi from a priority requesting device is issued, the Priority Dispatcher returns a 4-bit priority number for that specific device. The requesting RLTC sees this number on its port and stores it to be used for arbitration. Issuing the priority by the Priority Dispatcher is instantaneous and can be read immediately by the requesting device. This circuit works like a lookup table.

   An RTLC circuit that needs to use the bus, after receiving its priority, will send the priority and its Ri request to the arbiter. Based on the priority and request, the arbiter will issue a grant (Gi), and will put the id of the requesting RTLC (the 4-bit generic) on its output on the bus. The requesting device (RTLCi) will use the bus while holding it request asserted, and will de-assert its request when done.

   A) Write VHDL description for the Priority Dispatcher circuit. As input, the circuit has an unconstrained array of 1-bit Pi priority requests from several devices. The output of the circuit is an unconstrained array of 4-bit priority codes that are assigned to the requesting devices. Write a process statement that assigns arbitrary priorities to the requesting devices as soon as a request is issues.
   B) Show the RTL datapath and the controller for an RTLC device, for the part that corresponds to priority lookup and arbitration. Show a partial state diagram for this part and the corresponding registers for this. Provide mechasi8m for requesting use of the bus from the arbiter. Fake the operation of RTLC by keeping the request active for a few clock cycles before de-asserting it. Provide a go signal for each device that starts its operation after seeing a complete pulse. Also, provide a generic for the number of clocks the device holds its request active after it has been granted the bus. This number with be set to an arbitrary number between 3 and 9 by the testbench.
   C) For the arbiter, write a resolution function that issues grant to a requesting device based on its priority. The granting is issued by the arbiter outputting the id of the granted device.
   D) Build a complete system that has eight RTLC devices.
   E) Write a testbench that tests system of Part d with various timings for the go signals and different values for the operation of each unit.

2. In this assignment you are to implement a 36-bit synthesizable sequential residue circuit. A) First, write VHDL code for a modulo-7 adder which is actually a 6-bit residue circuit. Call this module *residue7*. The module takes two 3-bit inputs and produces a 3-bit result that is modulo-7 of the add result. Adding 101 plus 011 results in 001; adding 111 plus 111 results in 000; adding 110 plus 100 results in 011; adding 111 plus 000 results in 000. Use VHDL concurrent signal assignment statements to describe *residue7*. Figure below show how a combinational residue 7 circuit that has a 36 bit input. This circuit calculates the divide by 7 remainder of a 36-bit number. B) Show the RT level design of a sequential residue circuit that uses only one residue7 circuit of Part A. Show the datapath and the controller for this RT level circuit. C) Show the VHDL code of Part B. D) Add handshaking to your controller such that the input data is accepted after a complete pulse on *dataready*, and when the output becomes ready a pulse appears of *outready*. E) Write a testbench for this circuit to test it for several sets of data.