UNIVERSITY OF TEHRAN
Electrical and Computer Engineering Department
Digital System Design with Hardware Description Languages
Fall 1391 / Test 1

Computer Account#_____

First Name:_____

Last Name:_____

Number:_____

Signature:_____

Grade:

Problem 1. _____/30

Problem 2. _____/20

Problem 3. _____/20

Problem 4. _____/15

Problem 5. _____/15

Total: _____/ 100

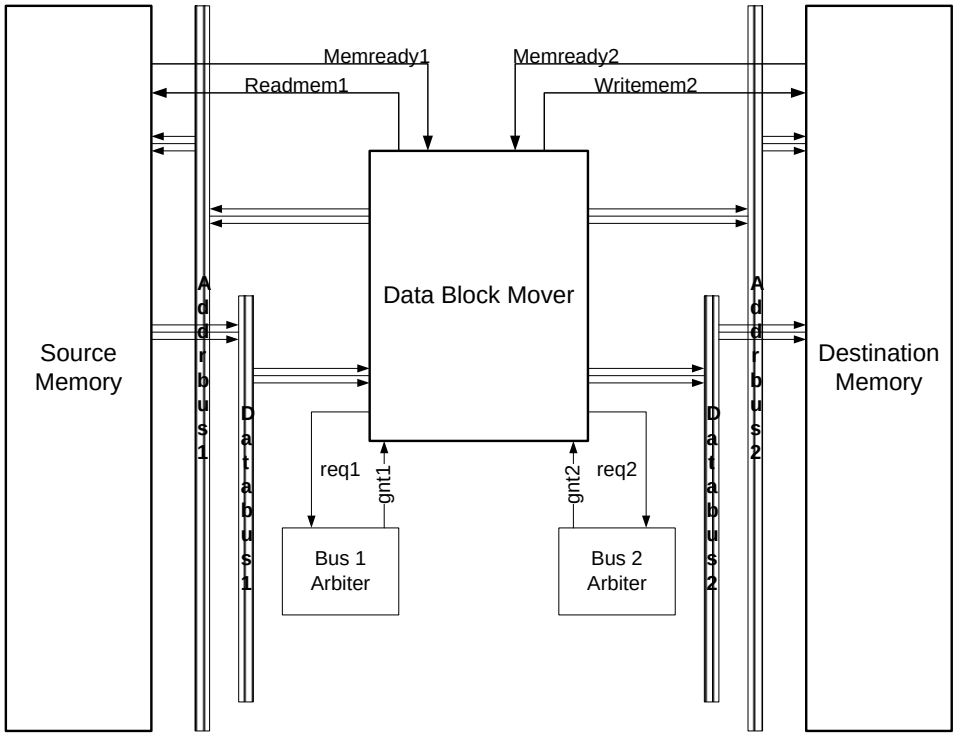
THIS IS AN OPEN BOOK OPEN NOTE EXAM
YOU HAVE EXACTLY TWO HOURS AND A HALF TO DO THIS TEST
YOU MUST SHOW COMPLETE WORK ON ALL PROBLEMS

1. **RT LEVEL DESIGN.** *RTL Design.* In this problem you are to design a complete RTL circuit with two levels of handshakings that communicates between two 64K memory systems. The block diagram of the complete system shown below, in which the circuit you are designing is called Data Block Mover (DBM). The system shown uses two shared busses and memories that are accessed by other devices as well as DBM.

   The DBM circuit moves a block of 8-bit data from a location of the source memory specified by source starting address (SSA) to the destination memory starting from the destination starting address (DSA). SSA and DSA are registers inside the data block mover that are accessible by a processor that will set the register contents to the desired locations. In addition, DBM has another 16-bit register that specifies the number of data bytes to be moved, NDB.

   The data moving process begins when DBM receives a 1 on its *Startmoving* input. After this, the DBM has to get permission from both the source bus arbiter and destination bus arbiter to use the two busses. When both permissions are granted, DBM starts from SSA in the source memory and moves NDB bytes from there to the destination memory starting from DSA. For each byte to be read from the source memory, *Readmem1* has to be issues and must be kept asserted until *Memready1* is issued by the timer of the source memory. The same handshaking procedure must be followed for writing to the destination memory.

   A) Show the complete datapath of DBM. SSA, DSA, and NDB are registers that DBM can read, but cannot be altered or written into. Clearly show schematic of all data components and their control wires that the controller controls. For each component, show enough details that the schematic can easily be turned into a VHDL code if needed. Since the busses are tri-state busses, make sure you provide proper drivers for them.
   B) Write complete VHDL code of the controller.
   C) Show the complete design of DBM by wiring its datapath and controller together. Show all connecting signals.


Source Memory
Addrbus1
Memready1
Readmem1
Memready2
Writemem2
Data Block Mover
Databus1
req1
gnt1
gnt2
req2
Bus 1 Arbiter
Bus 2 Arbiter
Databus2
Addrbus2
Destination Memory

2. **BEHAVIORAL COMPONENT MODELING.** A) Write VHDL for a complete memory model including read or write request and memory ready signals. Write this memory such that it would fit the memories required by both 64K memories of Problem 1. B) Write behavioral code for an n-port arbiter, and write a configuration declaration that would turn this arbiter into a 4-port arbiter as needed by the arbiters of Problem 1.

3. **ITERATIVE HARDWARE.** Using a generate statement generate a system that consists of four DBM of problem 1. The SSA and DSA registers of these four units must be accessible from outside. You need to write a VHDL Entity-Architecture pair that encloses these four units.

4. **COMPLETE SYSTEM INTERCONNECTIONS.** Wire the components of Problem 2 and the four-DBM system of Problem 3 to build a system that has four data transfer modules that can be programmed and activated independently.

5. **TESTBENCH.** Write a testbench to program the four ports of the quad transfer system of Problem 4 and start their transfers. Keep transferred data in separate parts of the memory. Your testbench plays the role of a CPU that initiates the transfers.