Worcester Polytechnic Institute
Electrical and Computer Engineering Department
ECE 759D
Methodologies for System Level Design and Modeling
Fall 2014

Final Exam

Zainalabedin Navabi

---

1. Design an 8-bit ALU that performs 8 functions according to the following table. The inputs are unsigned 8-bit numbers.

s_2	s_1	s_0	Function
0	0	0	A add B + C_{in}
0	0	1	A sub B – C_{in}
0	1	0	Right-Shift A by 1, zero-fill
0	1	1	Left-Shift A 1, zero-fill
1	0	0	NOT A
1	0	1	A & B (Bitwise AND)
1	1	0	A | B (Bitwise OR)
1	1	1	Max (A, B)

a. Design this ALU using RTL C++.
b. Use *bus* class and its utilities
c. Write a *main* in C++ and verify your design.

2. In this problem you are to design and implement a circuit to find the maximum value among the incoming clocked data. This circuit has two input signals called *Start* and *Stop* and a data bus, *Databus*. After a positive pulse that is synchronized with the clock appears on the *Start* signal, the circuit starts picking up a data word on the positive edge of the clock from the *Databus*. This process continues until a positive pulse appears on the *Stop* signal. At this point the circuit should put the maximum value among all captured data into the *Max* output and issues the *Ready* output signal. Draw a schematic that shows the details of the datapath and the controller part of the circuit. Then Implement the circuit using SystemC. Separate datapath and the controller. Write a testbench that applies data to your circuit and verify your design.

3. The purpose of this problem is to model a multiport memory system using SystemC primitive channels, and develop several modules that perform read and write operations from and to this memory. Figure below shows a communication environment which contains four processes (P1- P4) and one multiport memory. Each process (each is an SC_MODULE) needs to read or write from/to the memory several times at various times. The memory has three read ports and one write port that can function concurrently. This means that three processes can read from the memory simultaneously, but only one process can write to the memory at the same time.

a. Write a SystemC code for the memory and its communication environment. Use *sc_mutex* for controlling the write port and *sc_semaphore* for controlling the read ports.

b. Develop four modules that request read and write operations at various times causing overlaps in requesting. Write a test environment that examines your concurrent processes and the memory module.

Note: The data to / from the memory is handled by a *struct* that contains both address and data. This *struct* is passed by reference.