1. **VHDL Design Modeling.** You are to implement a 1011 Moore sequence detector in VHDL. In addition to detecting the sequence, the circuit keeps track of modulo-256 count of the 1011 sequences ever detected. When the correct sequence is detected, the w output becomes 1 and at the same time an 8-bit counter is incremented. Include an asynchronous active-high reset input for all clocked components.
   A. Show the state diagram for this circuit.
   B. Write a behavioral VHDL description of this sequence detector using VHDL PROCESS statements (no gate-level implementation is needed). Use Huffman model.
   C. Write a testbench for the circuit in VHDL that tests the correctness of your circuit.

2. **C/C++ RTL Design Modeling.** An 8-bit, 4-function ALU is to be designed. The table below lists functions that the ALU can perform and their corresponding opcodes. The ALU inputs are data A[7:0] and B[7:0], and the 2-bit function F[1:0]. The ALU outputs are W[7:0], carry (c), and greater (g), where W is defined in the table below, c is the carry output for arithmetic operations, and g becomes 1 when A is greater than B.
A) Show the block diagram of this circuit using adder, multiplexer, other RTL components discussed in lectures.
B) Write a C++ description that corresponds to the block diagram of Part A. Use the *bus* library discussed in lectures (Chapter 3).

Opcode	Function
00	W = A + B
01	W = Max (A, B)
10	W = 0
11	W = A | B

3. SystemC RTL Modeling. A processing element for approximate calculation of *1/(1-x)* is provided below. The input range is $0 \leq x \leq 0.5$. The circuit data input and output are 18-bit fixed-point numbers with two integer bits and sixteen fractional bits. The circuit receives its fractional-only data input via its 18-bit *XBus* input. This happens after the circuit receives a complete positive pulse on the *go* input. When the calculation is complete, the circuit issues a *ready* pulse. This happens after sixteen iterations.

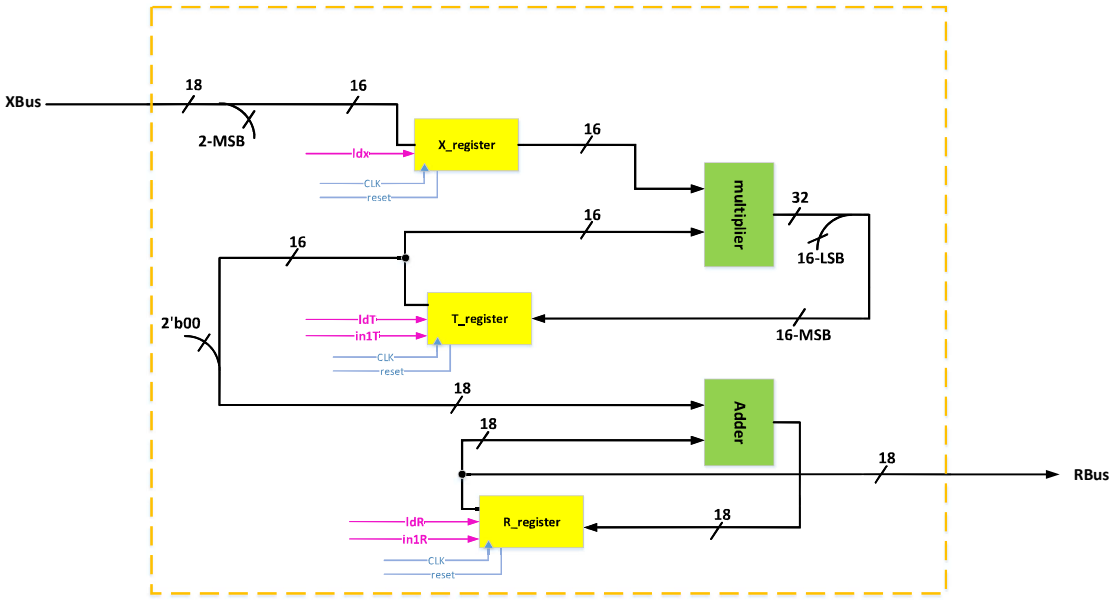
$$\frac{1}{1-x} = \sum_{n=0}^{\infty} x^n = 1 + x + x^2 + x^3 + \cdots$$

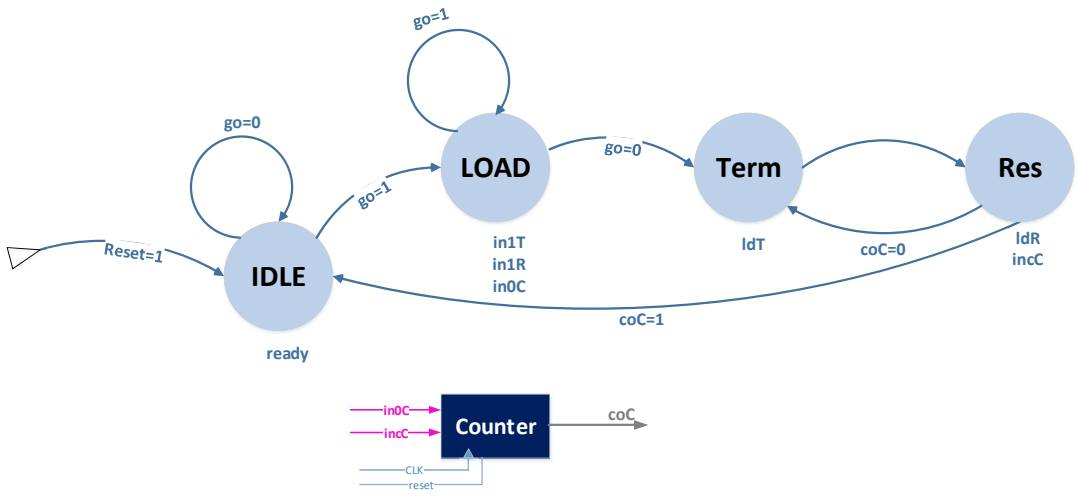
RTL Structural Modeling

A. Develop the complete controller (CU) using SystemC. Use Huffman model.
B. Implement the datapath (DP) of the circuit using SystemC.
C. Complete the design of the circuit by wiring CU and DP in a top-level module.
D. Write a testbench and verify your design using different scenarios. Show VCD viewer results.

Functional Modeling

A. Develop a cycle-accurate behavioral model (BFM:Bus Functional Model) for the circuit.
B. Write a testbench and verify your design using the same scenarios as in Part D.





Attention:

The midterm covers the following topics on Canvas:

“Topic 0.1 - Logic review”
“Topic 0.2 - HDL Background”
“Topic 3 - RT level modeling with C++ procedural language”
“Topic 4 - SystemC RTL Modeling”

Deliverables:

All your VHDL, C++, and SystemC codes and a complete report containing all parts of the exam. Make sure that your deliverable follows all mentioned rules in the assignment part of the "*ECE5723 - Comprehensive Syllabus 2020.pdf*" file. The file has been posted on Canvas.
