

14:332:437 Digital System Design  
Fall 2009

Problem Set 3

Electrical and Computer Engineering Department  
College of Engineering  
Rutgers University

Prof. Michael L. Bushnell

Assigned: October 8, 2009

Due: October 14, 2009

STUDENTS ARE EXPECTED TO WORK INDEPENDENTLY ON BOTH HOMEWORK AND EXAMINATIONS. NO COLLABORATION IS PERMITTED, UNLESS YOU ARE OTHERWISE INSTRUCTED. YOUR WORK MUST BE YOUR OWN.

1. *State Machine Design.* Given the state machine transition graph in Figure 1, please write a Verilog description for the state transitions of this diagram. You need to provide signals for the *present* and *next* states of the machine, an module declaration for the entire machine (*clk*, input *g*, input *h*, output *Z1*, output *Z2*, and *reset*), an always block for the state transition behavior (expressed as if-then-else or case statements), and an always block for clocking and resetting the machine. Note that in the state transition graph, we show the transition behavior as  $g h = 00 / 1 0$  to indicate that the transition occurs when  $g = 0$  and  $h = 0$ , and causes *Z1* to be 1 and *Z2* to be 0. Assume that state *A* is the reset state for this machine. Turn in your Verilog behavioral description, timing waveforms for the behavioral description simulation, the Verilog synthesized logic description, and a plot of the logic schematic.
2. *Arithmetic Logic Unit Design.* Design an *arithmetic logic unit* (ALU) for a computer that operates on the input bus signals  $[0 : 15]$  *ABUS*,  $[0 : 15]$  *BBUS*, and produces an output bus signal  $[0 : 31]$  *CBUS* with the result of the ALU's computation. The operations that the ALU must do are:

Operation	<i>CODE</i>	Meaning
Subtraction	1	$CBUS [16 : 31] = A - B$
Addition	2	$CBUS [16 : 31] = A + B$
Multiplication	3	$CBUS [0 : 31] = A \times B$
Arithmetic right shift	4	$CBUS [16 : 31] = A \gg \gg B$
Left shift	5	$CBUS [16 : 31] = A \ll \ll B$
Bit clear	6	$CBUS [16 : 31] = A AND (NOT B)$

The hardware variable  $[0 : 3]$  *CODE* indicates to the ALU which operation should be done. After each operation, set the signals *ZERO* and *NEGATIVE* to indicate whether the resulting bits on *CBUS* are all zero or whether they represent a negative number, respectively.

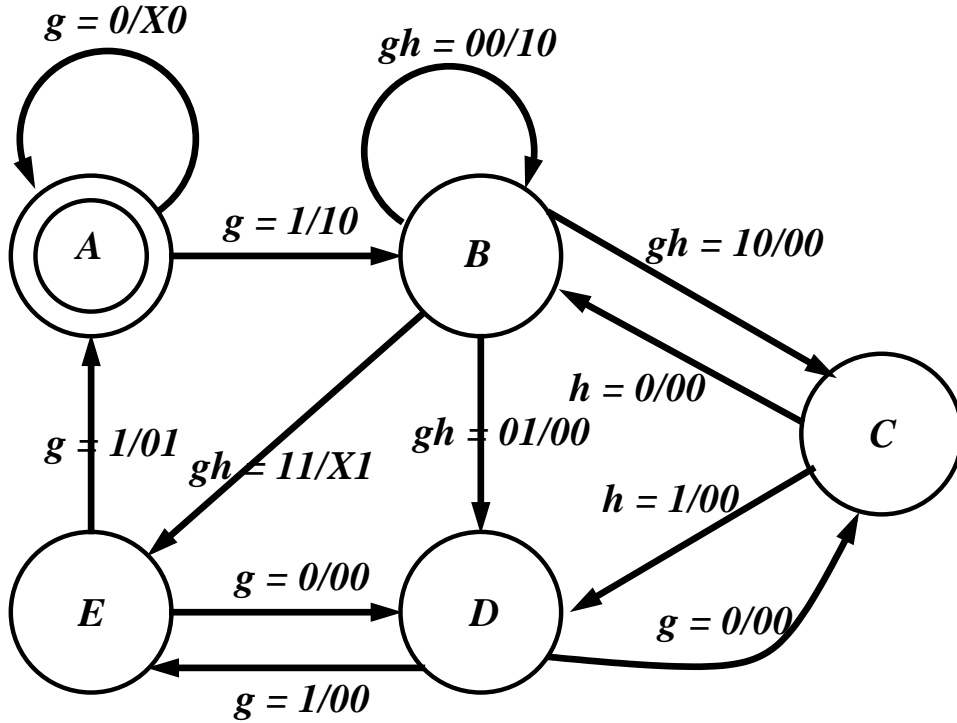


Figure 1: State Transition Diagram for Problem 1.

Set the signal *CARRY* appropriately after Subtraction or Addition to indicate whether a carry out or borrow out occurred. For all other operations, set *CARRY* to 0. Bits on *CBUS* that are not set by an operation should be cleared to 0. Turn in the Verilog code, the circuit schematic created by design\_vision, and the simulation results from vcs running the testbench.