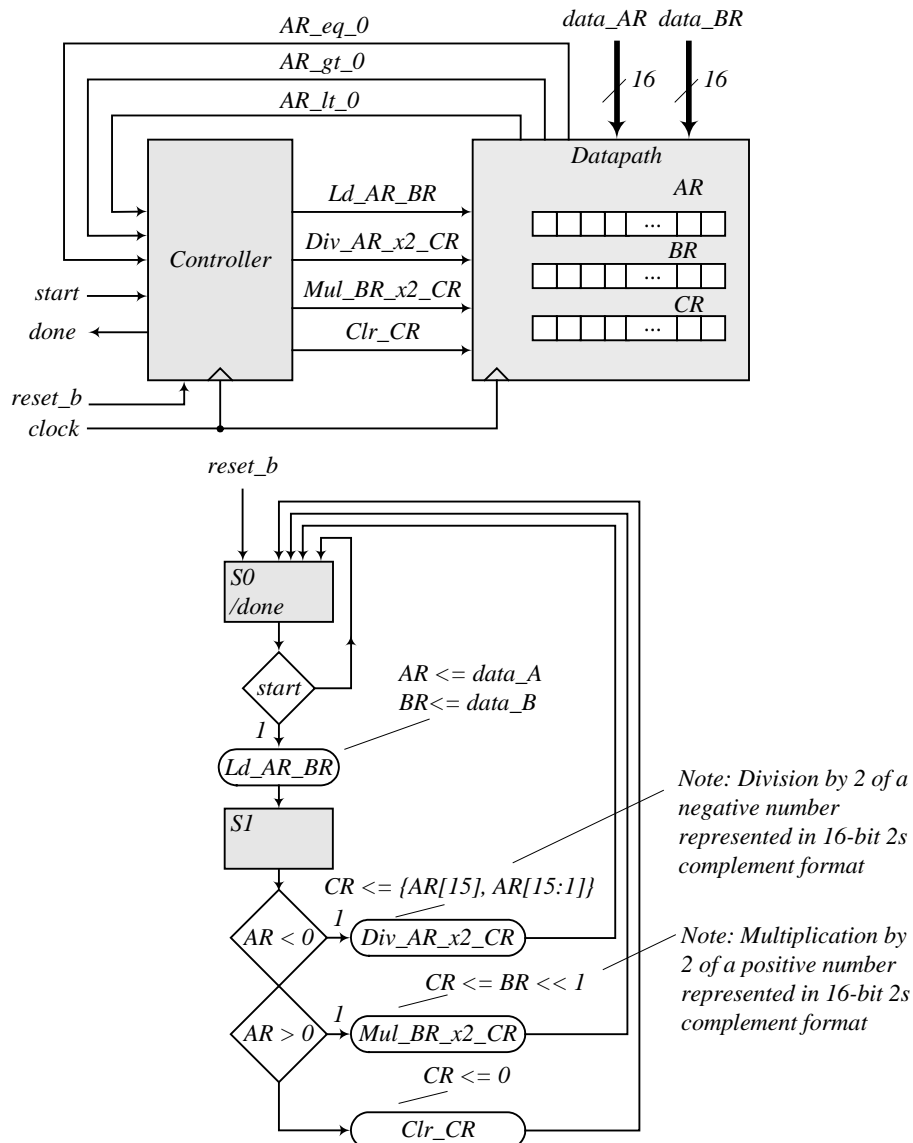


CSE 40462 VLSI Design
Introduction to Modeling with Verilog
Due Monday, Sept. 25, 2006

1. The figures below show a datapath and controller for a circuit that: (a) transfers two 16-bit signed numbers in 2's complement representation into registers *AR*, and *BR*, (b) if the number in *AR* is negative, divides the number by 2 and transfers the result to register *CR*, (c) if the number in *AR* is positive but nonzero, multiplies the number in *BR* by 2 and transfers the result to register *CR*, and (d) if the number in *AR* is 0, clears register *CR* to 0. Write, verify, and synthesize a Verilog model of the circuit (Provide Verilog code, annotated results of simulation, and schematics of the top level, datapath unit, and control unit of the circuit. Note: the register operation for sign extension is shown on the ASMD chart, as an alternative to the arithmetic right shift operator).



2. A digital circuit is to multiply two binary words by the repeated addition method. For example, to multiply 5×4 the circuit evaluates the product by adding the multiplicand four times: $5 + 5 + 5 + 5 = 20$. This is a basic, but not very efficient, way to implement multiplication (see the lecture slides for a more efficient implementation), but it serves well as an introductory assignment for designing a datapath controller using an ASMD chart.

Specifications:

The datapath unit is to have a multiplicand register, *BR*, a multiplier register *AR*, and a product register *PR*. A control unit directs the addition of *BR* to *PR*, and decrements *AR*. The primary (external) inputs are *Start*, a signal that launches the activity of the machine, *clock*, and *reset_b* (active-low, asynchronous reset). The machine is synchronized by the positive edge of *clock*. The primary outputs are *Done*, a signal that asserts while *PR* holds a valid product, and *PR*, the product. The data is to be acquired from 8-bit data busses *data_BR* and *data_AR*. A status signal, *Zero*, is to indicate that the multiplier is 0, and is to be used to terminate multiplication early, if possible. Use a one-hot state code for the controller. Clearly identify any assumptions that you have made.

Deliverables:

- Block diagram showing inputs, outputs, interface signals, and datapath registers (do not draw the architecture of the datapath)
- Complete ASMD chart
- Verilog models of the controller, datapath, and testbench
- Annotated simulation results demonstrating that the machine operates correctly
- RTL schematics obtained by synthesizing the multiplier with the Mentor Graphics *Precision* tool. Provide a separate schematic for the top level of the design, the datapath unit, and the control unit
- A brief discussion of the results