ECE 410/510
Introduction to SystemVerilog
Summer 2011
Homework 3

Consider a simple system in which there is a single "master" processor and one or more "slave" devices. They share a synchronous parallel data bus, two control signals (WB and RB for write bus and read bus respectively), one handshaking signal (Ack), an 8-bit address that uniquely identifies the slave that is the target of the transaction, and a single parity bit that provides parity for the data. Signals are sampled on rising clock edges.

For read transactions, the master places the target ID of a device on the address lines and asserts RB. The target recognizes its ID at the next clock edge and transfers the data on the following clock edge, asserting Ack. Upon receiving Ack the master latches the data and de-asserts RB. For write transactions, the master places the target ID of a device on the address lines, data on the data bus, and asserts WB. At the next clock edge the target device latches the data and asserts Ack. Upon receiving Ack the master tri-states the data bus and de-asserts WB.

Model a system configured with a master and two slave devices. Use an interface with modports for master and slave, and provide interface methods (tasks) for reading and writing the bus so that the master and slave modules don't have to do the handshaking directly. Provide an interface function for the sender to compute the parity for the data. Provide another for the receiver to check the parity. The width of the bus should be a parameter that can be overridden when the interface is instantiated.

Can you propose a better way to handle the parity checking? You do not need to implement it.