CS 525V: Warming Up with CTL and SMV

due: Jan 22, 2002; in class and electronically

These exercises will give you some practice modeling designs, specifying properties, and using SMV. Turn in your solutions in accordance with the instructions for documenting SMV assignments on the course assignments page. As with all assignments, follow the collaboration policy provided on the course web page.

Exercise 1 (**More Practice with CTL**) From the "Exercises in CTL" handout distributed in class, do the following problems: 1-1, 1-6, 2-4, 2-6, 2-10, 2-14, 3-4, 3-5.

Exercise 2 (**Arithmetic Shifter**) An *arithmetic shifter* stores a bit-vector of some specified length. It has two control signals: shift-left and shift-right. When shift-left is asserted, the bits in the vector are shifted one bit to the left, with a 0 shifted into the rightmost bit. When shift-right is asserted, the bits in the vector are shifted one bit to the right, with a 0 shifted into the leftmost bit. For example:

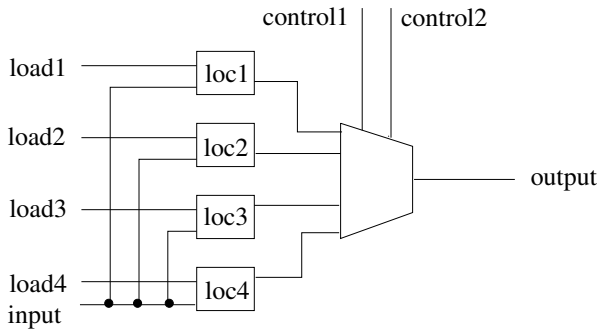
Contents of vector	Operation	New contents of vector
1 1 0 1	shift-right	0 1 1 0
1 1 0 1	shift-left	1 0 1 0

1. Write an SMV specification of a 4-bit arithmetic shifter; call the file *shifter.smv*. Use a style similar to that used for the counter in the SMV-CTL manual (*i.e.*, make a cell module, where each cell corresponds to a single bit of the shifter).
2. Determine the properties you need to test in order to verify the shifter. Argue why the set of properties you propose is sufficient.
3. Use SMV to verify the properties given in response to part 2.

Exercise 3 (**Simple Memory**) Suppose we have a simple memory with a single input bus and four storage locations. Each storage location has its own load signal; when the load signal is asserted, the value currently on the input bus is stored in that memory location. There is a single output bus for the memory. The value on the bus is controlled (via a multiplexer, if you're familiar with that term) by a pair of output control signals. The values on these signals indicates which location's contents should be placed on the bus, according to the following table:

Control1	Control2	Location
0	0	1
1	0	2
0	1	3
1	1	4

The layout of this memory appears as follows:



The desired properties of the memory are:

- Values are properly stored in the memory based on the load signals.
- Values are routed from the locations to the output bus as specified by the above table.
- Once a value has been loaded into a location, the location retains the value until the next load to that location occurs.

Download the file *simplemem.smv* from the assignments page and use SMV to verify the above properties against that design. Turn in your modified *simplemem.smv* file.