

Assignment 6

Posted: May 25, 2012
Due: June 1, 2012 at 12pm (noon)
Topic: Modeling and Simulation in SystemC

1. Setup:

We will use the SystemC version 2.2.0 installed on the server at:

```
/opt/pkg/systemc-2.2.0/
```

In order to use `turnin` to submit your deliverables, create a new directory named `hw6` (next to your `hw5` directory) and work there:

```
mkdir hw/hw6  
cd hw/hw6
```

2. Reference Example

We will use the simple FIFO example presented in Lecture 8 as reference model. Please refer to the presentation by Stuart Swan:

```
Lecture8_SystemC_Intro.pdf
```

The source code for this example is available in the SystemC package installed on our server:

```
cp /opt/pkg/systemc-2.2.0/examples/sysc/simple_fifo/simple_fifo.cpp .
```

You can compile the example with the regular GNU C++ compiler and simulate it by regular execution, as follows:

```
g++ simple_fifo.cpp -I/opt/pkg/systemc-2.2.0/include  
-L/opt/pkg/systemc-2.2.0/lib-linux -lsystemc  
-o simple_fifo  
./simple_fifo
```

If the output shown is not what you expect from reading the code, then read the code again. Pay attention to the non-determinism present in this example you will likely see that it is indeed correct.

3. Instructions

To keep the amount of recoding in this assignment to a level suitable for a one-week homework assignment, we will reuse the simple producer/consumer model of Assignment 1. For your reference, we have provided a solution file.

```
cp ~eecs222/EECS222A_s12/ProdCons.sc .
```

The task of this assignment is to translate this SpecC model into an equivalent SystemC model. Since much of the code can be reused, you may want to start with copying the SpecC code as a starting point for your SystemC model.

```
cp ProdCons.sc SystemC_ProdCons.cpp
```

Edit the SystemC model such that it follows the SystemC language rules and guidelines and can be compiled and simulated with the SystemC library. For grading purposes, be sure to keep the same names of the objects and variables whenever appropriate. For example, there should be a module named `Prod` instantiated as `prod`, equivalent to the producer behavior in the SpecC model.

With the exception of the SystemC header message and any non-deterministic ordering of producer and consumer messages, the output of your SystemC model should match the output of the given SpecC model. To demonstrate that your program compiles and simulates as expected, compile and run it. Copy the output of the compiler and simulator into a file named `SystemC_ProdCons.log`.

4. Submission:

For this assignment, submit the following deliverables:

```
SystemC_ProdCons.cpp
SystemC_ProdCons.log
```

To submit the deliverables, change into the parent directory of your `hw6` directory and enter `turnin`. As in the previous assignments, the `turnin` command will locate the files listed above and allow you to submit them.

Remember that you can use the `turnin` tool to submit at any time before the deadline, *but not after!* Since you can submit as many times as you want (newer submissions will overwrite older ones), it is highly recommended to submit early and even incomplete work, in order to avoid missing the deadline.

Late submissions cannot be considered!

--

Rainer Doemer (EH3217, x4-9007, doemer@uci.edu)