EECS 222A
System-on-Chip Description and Modeling
Spring 2012

Assignment 1

Posted: April 13, 2012
Due: April 20, 2012 at 12pm (noon)

Topic: Introduction to SpecC Compiler and Simulator

1. Setup:

A Linux account has been created for you and the login and password information have been sent to your UCI email address. If you have problems with accessing the account, please contact the course instructor by email.

We will use the following host as server:

```
alpha.eecs.uci.edu
```

In order to connect to the server remotely, you will need to use the secure shell protocol (SSH) via some client software. Note that in the beginning we will use only the command line interface, so a simple terminal program (e.g. Putty) will be sufficient. For later assignments, we will also use graphical tools for which you need X client software (e.g. Xming).

After logging in, you will need to setup your Linux environment in order to use the SpecC tools, as follows:

```
source /opt/sce-20100908/bin/setup.csh
```

This will setup your execution path to the latest stable version of the SpecC compiler, simulator, and other tools.

2. Task A: Simple Examples

As an initial step to get familiar with the SpecC compiler and simulator, we will examine, compile, and run some very simple examples, as follows:

```
mkdir simple_tests
cd simple_tests
cp /opt/sce-20100908/examples/simple/* .
ls
man scc
vi HelloWorld.sc
scc HelloWorld -vv
./HelloWorld
ls
...
```

Inspect, compile, and simulate also the other examples!

3. Task B: Producer-Consumer Example

As deliverable for this first assignment, write and simulate a SpecC program of a simple Producer-Consumer example.

Your program should contain two parallel behaviors named **Prod** and **Cons** that communicate via a channel **C**. The producer should send the string **“Apples and Oranges"** to the consumer. The communication should be character by character (a single byte at a time should be sent/received through the channel), and the consumer should print the received characters to the screen. After the entire message is communicated, both consumer and producer should end so that the main program cleanly completes.

This is how your program output should look like:

```
Main starts.
Producer starts.
Producer sends 'A'.
Consumer starts.
Consumer received 'A'.
Producer sends 'p'.
Consumer received 'p'.
Producer sends 'p'.
Consumer received 'p'.
Producer sends 'l'.
Consumer received 'l'.
Producer sends 'e'.
Consumer received 'e'.
```

```
[...]
Producer sends 's'.
Consumer received 's'.
Consumer done.
Producer done.
Main done.
```

To setup this task, create a new directory named **hw1** and work inside it. Name your source code file **ProdCons.sc**. Type the file name exactly as shown (you can't submit it otherwise!).

To write your program, take a look at slide 25 of Lecture 2. Your program should be very similar to the example shown there, with only a few modifications and some additional statements (e.g. to print the lines shown above).

To demonstrate that your program simulates as expected, compile and run it. When running, redirect the output into a file named **ProdCons.log** (Again, use exactly this filename!).

3. Submission:

For this assignment, submit the following deliverables:

```
ProdCons.sc
ProdCons.log
```

To submit these files, change into the parent directory of your **hw1** directory and enter **turnin**. The **turnin** command will locate the deliverables the current assignment asks for and allow you to submit them, as follows:

```
doemer@mu.eecs.uci.edu:22 > ls
ProdCons*    ProdCons.h    ProdCons.o   ProdCons.si
ProdCons.cc  ProdCons.log  ProdCons.sc  simple_tests/
doemer@mu.eecs.uci.edu:23 > cd ..
doemer@mu.eecs.uci.edu:24 > turnin
==============================================================
EECS 222A Spring 2012:
Assignment "hw1" submission for doemer
Due date: Fri Apr 20 12:00:00 2012
** Looking for files:
**   ProdCons.sc
**   ProdCons.log
==============================================================
* Please confirm the following:                              *
* "I have read the Section on Academic Honesty in the        *
* UCI Catalogue of Classes (available online at              *
* http://www.editor.uci.edu/catalogue/appx/appx.2.htm#academic) *
* and submit my own original work accordingly."              *
Please type YES to confirm. YES
```

```
==============================================================
Submit ProdCons.sc [yes, no]? y
  File ProdCons.sc has been submitted
Submit ProdCons.log [yes, no]? y
  File ProdCons.log has been submitted
==============================================================
  Summary:
==============================================================
Submitted on Fri Apr 13 11:06:44 2012
You just submitted file(s):
  ProdCons.sc
  ProdCons.log
```

Note 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)