System-On-Chip Enviroment (SCE) Setup

Andreas Gerstlauer

1. Access and Setup

Below are the instructions for accessing and setting up your environment for running SCE on the Linux Machines in the department:

(a) The SpecC tool set is installed only on the Linux machines in the ECE LRC. Information about how to access the LRC machines is available at:
http://www.ece.utexas.edu/it/remote-linux
You can work on any of the regular 64-bit Linux servers.

(b) Once you have logged into one of the ECE LRC linux machines, you need to load the SCE module via:
% module load sce

(c) Now to launch SCE, type 'sce' at the command line
% sce
This will pop up the SCE window. You can get online help at any time through Help buttons or by selecting Help→Manual.

Documentation for SCE is installed under *$SPECC/doc*:

- Copies of the SCE Manual: *$SPECC/doc/SCE_Manual*
- SCE Tutorial: *$SPECC/doc/SCE_Tutorial*, and the latest errata for the tutorial: *$SPECC/doc/SCE_Tutorial_Errata.pdf*
- SCE Specification Reference Manual: *$SPECC/doc/SpecRM.pdf*
- SCE Database Reference Manual: *$SPECC/doc/DBRM.pdf*
- SpecC Language Reference Manual: *$SPECC/doc/SpecC_LRM.pdf*

2. SCE Tutorial

SCE includes a tutorial that demonstrates the environment as applied to the example of designing a GSM voice encoding (Vocoder) application for mobile phone applications. Below are instructions for getting started with and going through the tutorial:

(a) Do the setup of your SCE environment as described above. If the *setup_demo* command (see (b) below) does not work, you might have to reset your LRC environment:
% resetenv
Then follow the instructions and reload the *sce* module.

(b) To work through the tutorial you need about 200MB of hard disk space, i.e. you might run out of your quota in your regular ECE home directory. You can create your own temporary directory in the public */scratch* space, which is accessible from all Linux machines. Note, however, that */scratch* is not backed up and will be wiped out at the end of the semester, so do not store any important files there!

First, create a directory with your username in the */scratch* directory:

```
% mkdir /scratch/<username>
% cd /scratch/<username>
```

Next, create a working directory for the tutorial in your scratch space:

```
(/scratch/<username>)% mkdir tutorial
(/scratch/<username>)% cd tutorial
```

In this tutorial directory run the command *setup_demo*:

```
(/scratch/<username>/ tutorial)% setup_demo
```

This will create the required files for your demo:

```
(/scratch/<username>/ tutorial)% ls -F
DEMO_FINAL.tar.gz@  IP  Makefile  SCE_Tutorial@  src/  testbench.sc@
vocoder.sce
```

(c) Now you can start following the tutorial:
http://www.cecs.uci.edu/~cad/publications/tech-reports/2003/TR-03-41.tutorial.pdf
However, since this tutorial is old and not updated for the most recent version of SCE that is currently being used, please also refer to the errata document at:
http://www.ece.utexas.edu/~gerstl/ee382n_f17/docs/SCE_Tutorial_Errata.pdf

(d) If you quit the tutorial at some intermediate step, the next time you login to continue from the earlier point you need to go into the work directory which was used earlier and run *setup_demo* again:

```
(/scratch/<username>/ tutorial)% setup_demo
```

Ignore the warnings about any links already existing, none of your files will be deleted. You can continue the tutorial by launching SCE, opening the *vocoder.sce* project and then loading the last generated sir file by double-clicking on it in the project window.