SOCGEN APPLICATION NOTE

Getting Started.

Installing and using SOCGEN on a Ubuntu 12.10 Linux system.

The SOCGEN project is an opensource design-for-reuse toolset that provides tool flows for building a System-on-chip (SOC) design and verifying that everything properly functions. Current flows include:

- Icarus Verilog Simulation
- Verilator Simulation
- Code coverage using covered
- Rtl Linting using Verilator
- Fpga synthesis if Xilinx ISE is installed

A socgen component is created by adding ip-xact metadata files to each component. The tool flows will read these files and use them as an electronic data sheets for each and every component.

The socgen project is hosted on opencores.org and consists of a set of tool scripts along with some sample opencores projects that have been converted into socgen files.

The first step to install socgen on a fresh Ubuntu12.10 system is to download some needed applications:

```
user@system:~$ sudo apt-get  install -y subversion;
user@system:~$ sudo apt-get  install -y iverilog;
user@system:~$ sudo apt-get  install -y verilator;
user@system:~$ sudo apt-get  install -y covered;
user@system:~$ sudo apt-get  install -y gtkterm;
user@system:~$ sudo apt-get  install -y gtkwave;
user@system:~$ sudo apt-get  install -y crasm;
user@system:~$ sudo apt-get  install -y libverilog-perl;
user@system:~$ sudo apt-get  install -y libxml-libxml-perl;
```

Socgen is then downloaded from opencores.org (Login required)

```
user@system:~/Desktop/$  svn co --username <<>> --password <<>>
http://opencores.org/ocsvn/socgen/socgen/trunk socgen;
```

I found that 12.10 required some admin mucking to stop error messaging popups

```
user@system:~$  sudo sed -i  "s/enabled=1/enabled=0/g"
/etc/default/apport
```

The directory ./socgen/tools/bin should be chained into your $PATH variable

Install or32-elf toolchain if you want to use the or1k examples and Xilinx webpack 13.3 if you want to build fpgas.

Once installed all tool flows are run from the top level socgen directory. The first step is to build a workspace by selecting the library containing the designs top level from the socgen/projects directory.

user@system:~/Desktop/socgen$ ./tools/sys/workspace /projects/www.6502.org/Arlet_6502 /work

This creates a workspace image in ./socgen/work that is used all the generated results.

If the design uses software then it is compiled with the command

user@system:~/Desktop/socgen$ make build_sw

The hardware is created with the command.

user@system:~/Desktop/socgen$ make build_hw

All of the verification tool flows are run using

user@system:~/Desktop/socgen$ make run_sims
user@system:~/Desktop/socgen$ make run_coverage

Fpgas are synthesized with

user@system:~/Desktop/socgen$ make build_fpgas

Finished results are displayed with

```
user@system:~/Desktop/socgen$     make check_sims
user@system:~/Desktop/socgen$     make check_fpgas
```

Individual results are available under each component in the /work directory. For example to run and debug an individual simulation.

```
user@system:~/Desktop/socgen/work/www.6502.org/Arlet/ip/cpu/sim/icarus/inst_1_test$     make sim

user@system:~/Desktop/socgen/work/www.6502.org/Arlet/ip/cpu/sim/icarus/inst_1_test$     gtkwave -a wave.sav -f TestBench.vcd
```