PowerSMT Tutorial

Rogério Aparecido Gonçalves

rogerioag@utfpr.edu.br

1. Introdução

It is the first release of SS_SMT 1.0. It is a beta version to execute SS code in a SMT simulator. It is not completly reviewed and no support is provides. It is free.

This simulator was adapted by Ronaldo A. L. Goncalves using the original SimpleScalar 3.0 Tool that is a part of the SimpleScalar tool suite written by Todd M. Austin as a part of the Multiscalar Research Project.

Contact: ronaldo@din.uem.br (http://www.din.uem.br/~ronaldo)

The SimpleScalar suite is currently maintained by Doug Burger and Todd M. Austin.

INTERNET: dburger@cs.wisc.edu
US Mail: 1210 W. Dayton Street, Madison, WI 53706

To use SS SMT follow the steps (you can issue only ss_smt to view a brief description of the simulator):

1) uncompress the compressed file

2) execute make (some warnings will appear - don't panic)

3) cd tests

4) ./run_it (some tests will run)

5) To run the simulator, execute the follow command:

ss_smt <x1> <x2> {options} batch file

ss_smt 11 12 {options} RUN_SPEC95

which:

<x1> : initial line to consider from <batch file> (ignoring blank and comentaries lines)
<x2> : final line to consider from <batch file> (ignoring blank and comentaries lines)
{options} : original options from sim-outorder plus some new ones
<batch file> : text file: each line must be

a command line for one application that will
be executed for each slot

Please note, that, in order to execute more than one thread per time, your l1 cache configuration must have
the number of banks or associativity greater or equal to the number of threads. For example, to run the simulator
with 4 threads you should configurate your cache il1 like that:

-cache:il1 il1:4:xx:xx:xx:x (4 banks)
-cache:il1 il1:1:xx:xx:xx:4 (1 bank and 4 ways)
-cache:il1 il1:2:xx:xx:xx:2 (2 banks and 2 ways)

The only applications which are available are in the /tests directory
These applications are part of original SimpleScalar 3.0 tool.

==========

Parameters:
-bpred:type forced:n : force the bpred accuracy to n (%: 100 = 100%)
for example: -bpred:type forced:93 (93% accuracy)

-ruulsq:type distributed | shared : Estações de reserva (RUU) distribuídas por unidades funcionais ou compartilhadas entre elas.

-fu:type hetero | homo :

-imodules:num 2

2. **Instalação**
3. **Testes**
4. **Exemplo de Saída**
5. **Executando Benchmarks**
6. **Execução em Lote**
7. **Referências**

[1] BROOKS, D. et al. Wattch: A framework for architectural-level power analysis and optimizations. In PROC. OF 27TH ANN. INT'L SYMP. COMPUTER ARCHITECTURE, pages 83-94. IEEE Computer Society Process, Los Alamitos, USA, 2000.

[3] GONÇALVES, R. A., et al. Correlacionando Desempenho e Potência em Processadores Superescalares: Analisando Cache e Janela de Instruções In: II Encontro Paranaense de Computação, Cascavel. Unioeste, Ago. 2007.