

# Lab 1 Introduction to Access Noxim

## I. Purpose

In this Lab, we will introduce the 3D Network-on-chip (NoC) simulation tool – Access Noxim!

Please read the background information and follow the procedure and learn running a single simulation. Hope you will enjoy using our tool!

## II. Introduction

### A. SystemC

SystemC is a set of C++ classes and macros which provide an event-driven simulation kernel in C++ (see also discrete event simulation). These facilities enable a designer to simulate concurrent processes, each described using plain C++ syntax. SystemC processes can communicate in a simulated real-time environment, using signals of all the data types offered by C++, some additional ones offered by the SystemC library, as well as user defined. In certain respects, SystemC deliberately mimics the hardware description languages VHDL and Verilog, but is more aptly described as a system-level modeling language. Wikipedia, "SystemC", <http://en.wikipedia.org/wiki/SystemC>

### B. Noxim

Noxim, the Network-on-Chip Simulator developed at the University of Catania (Italy) is developed using SystemC. Noxim has a command line interface for defining several parameters of a NoC. In particular the user can customize the network size, buffer size,

packet size distribution, routing algorithm, selection strategy, packet injection rate, traffic time distribution, traffic pattern, hot-spot traffic distribution.

The simulator allows NoC evaluation in terms of throughput, delay and power consumption. This information is delivered to the user both in terms of average and per-communication results. In detail, the user is allowed to collect different evaluation metrics including the total number of received packets/flits, global average throughput, max/min global delay, total energy consumption, per-communications delay/throughput/energy etc. “Noxim, the NoC Simulator”, <http://noxim.sourceforge.net/>

## **C. Hotspot**

HotSpot is an accurate and fast thermal model suitable for use in architectural studies. It is based on an equivalent circuit of thermal resistances and capacitances that correspond to microarchitecture blocks and essential aspects of the thermal package. The model has been validated using finite element simulation. HotSpot has a simple set of interfaces and hence can be integrated with most power-performance simulators like Wattch. The chief advantage of HotSpot is that it is compatible with the kinds of power/performance models used in the computer-architecture community, requiring no detailed design or synthesis description. HotSpot makes it possible to study thermal evolution over long periods of real, full-length applications. “Hotspot”, <http://lava.cs.virginia.edu/HotSpot/>

## **D. Access Noxim**

The Access Noxim is a co-simulation platform for 3D NoC system that couples the network model, power model and thermal model. We integrate Noxim and HotSpot, and adopt the power model of Intel’s 80-core processor. Noxim is a cycle-accurate SystemC NoC

simulator, and HotSpot provides the architecture-level thermal model. To coupling with HotSpot, the NoC simulator should convert its architecture-level floorplan and power trace to chip-level physical floorplan and power trace. We first add the model of basic 3D router and the vertical crossbar router, and we extend Noxim to be able to generate 3D architectures of NoC based on user-defined parameters of dimension. Then a module is inserted for automatically converting the architecture-level floorplan to physical floorplan. During network traffic simulation, a power trace is generated based on the power model of the NoC. The power trace and physical floorplan are used as inputs of the thermal simulation. In the proposed simulator, the tile geometry and power model are based on Intel's 80-core chip. Kai-Yuan Jheng, Chih-Hao Chao, Hao-Yu Wang, and An-Yeu Wu, "Traffic-thermal mutual-coupling co-simulation platform for three-dimensional Network-on-Chip," in Proc. int. Sym. VLSI Design Automation and Test (VLSI-DAT), pp.135-138, April 2010.

## III. Procedure

### A. Installation of SystemC

- SystemC Download link: <http://www.systemc.org/downloads/standards>
- Installation
  - Windows: [Chinese](#)/[English](#)
  - Linux: [Chinese](#)/[English](#)

### B. Installation of Access Noxim

1. Download the Access Noxim and upload to your workstation.
2. Using "tar -zxvf [*filename*] [-C *target directory*]" to extract files.
3. Modify the file, "Makefile.defs", in /bin. Put the correct path of SystemC in line 8.

```
SYSTEMC = <your SystemC install path>
```

4. Compile the program.

```
% make
```

## C. Execute the Access Noxim

1. After execute “make” command, there’s a executable file called “noxim” in the /bin directory. Execute Noxim directly with default parameters.

```
% noxim
```

2. After few seconds, the statistics results shows.

```
Noxim simulation completed.
( 10001 cycles executed)
% Total received packets: 25530
% Total received flits: 204178
% Global average delay (cycles): 5003.17
% Global average throughput (flits/cycle): 0.384796
% Throughput (flits/cycle/IP): 0.079765
% Max delay (cycles): 10000
% Total energy (J): 0.103236
% Avg power (J/cycle): 1.03236e-05
% Avg power per router (J/cycle): 4.03267e-08
% Layer average delay (cycles): 5030.93 5001.82 5009.89 4969.72
% Layer energy (J): 0.0254396 0.025568 0.026202 0.0260268
% Layer Routed flits: 355090 357010 365710 363405
```

## D. Execute Access Noxim with Parameters

1. There are many parameters can be set in Access Noxim. You can type command below to show all parameters.

```
% noxim -help
```

2. And you’ll get the help information. We list some important arguments, and you can read more explanation on screen.

Option	Sub-Option	Explanation
-help		Show this help and exit
-verbose	N	Verbosity level (1=low, 2=medium, 3=high, default off)
-trace	FILENAME	Trace signals to a VCD file named 'FILENAME.vcd' (default off)

-dimx	N	Set the mesh X dimension to the specified integer value (default 8)
-dimy	N	Set the mesh Y dimension to the specified integer value (default 8)
-dimz	N	Set the mesh Z dimension to the specified integer value (default 4)
-buffer	N	Set the buffer depth of each channel of the router to the specified integer value [flits] (default 16)
-size	Nmin Nmax	Set the minimum and maximum packet size to the specified integer values [flits] (default min=8, max=8)
-routing	TYPE	Set the routing algorithm to TYPE where TYPE is one of the following (default 0):
-sel	TYPE	Set the selection strategy to TYPE where TYPE is one of the following (default 0):
-pir	R TYPE	Set the packet injection rate to the specified real value [0..1] (default 0.01) and the time distribution of traffic to TYPE where TYPE is one of the following:
-traffic	TYPE	Set the spatial distribution of traffic to TYPE where TYPE is one of the following (default 0):
-hs	ID P	Add node ID to hotspot nodes, with percentage P (0..1) (Only for 'random' traffic)
-warmup	N	Start to collect statistics after N cycles (default 1)
-seed	N	Set the seed of the random generator (default time())
-detailed		Show detailed statistics
-volume	N	Stop the simulation when either the maximum number of cycles has been reached or N flits have been delivered
-sim	N	Run for the specified simulation time [cycles] (default 10000)

- Now, try to run a single simulation that topology of NoC is 6 by 6 by 4(H). The router buffer size is 8 and packet size is (2,8). Set simulation cycle as 50,000 with warm up cycle 20,000. Keep other parameter in default value.

```
% noxim -dimx 6 -dimy 6 -dimz 4 -buffer 8 -size 2 8 -sim 50000 -warmup 20000
```

- Compare to the default setting, what's the difference?
- Try to design another setting, and run the simulation. Fill in your command below

```
% noxim <write down your parameters>
```

- Good job! Now you understand how to use our tool to run a single NoC simulation! Move to the Problem section to learn more about the properties of NoC!

## IV. Problems

### A. Packet Injection Rate

Please try to set different packet injection rate(PIR) and discuss what's the relationship between PIR and Latency and throughput?

PIR	0.01								0.05
Latency									
Throughput									

### B. Simulation Cycle

Please try to set different simulation cycle and discuss what's the relationship between simulation cycle and Latency and throughput?

Sim. Cycle	10000								
Latency									
Throughput									

### C. Default Setting

Can you describe the default parameters of NoC that we simulated in our tool? What parameter do you interest in? (hint: you can read the NoximMain.h)

## V. Reference

- [1] Wikipedia,"SystemC", <http://en.wikipedia.org/wiki/SystemC>
- [2] "Noxim,the NoC Simulator", <http://noxim.sourceforge.net/>
- [3] "Hotspot",<http://lava.cs.virginia.edu/HotSpot/>
- [4] Kai-Yuan Jheng, Chih-Hao Chao, Hao-Yu Wang, and An-Yeu Wu, "Traffic-thermal mutual-coupling co-simulation platform for three-dimensional Network-on-Chip," in Proc. int. Sym. VLSI Design Automation and Test (VLSI-DAT), pp.135-138, April 2010.