UNIVERSITY OF CALIFORNIA

Santa Barbara

Interconnect Fabric Reconfigurability for Network on Chip

A Thesis submitted in partial satisfaction of the

requirements for the degree Master of Science

in Electrical and Computer Engineering

by

Omri Almog

Committee in charge:

Professor Malgorzata Marek-Sadowska, Chair

Professor Li-C. Wang

Professor Yuan Xie

June 2015

The thesis of Omri Almog is approved.

_____

Li-C. Wang

_____

Yuan Xie

_____

Malgorzata Marek-Sadowska, Committee Chair

June 2015

Interconnect Fabric Reconfigurability for Network on Chip

by

Omri Almog

ACKNOWLEDGEMENTS

I would like to acknowledge the Semiconductor Research Corporation for supporting my thesis. This thesis would not have been accomplished without their financial support. I would like to express my special appreciation and thanks to my advisor Professor Malgorzata Marek-Sadowska, you have been a tremendous mentor for me. I would like to thank you for encouraging my research and for allowing me to grow as a research engineer. Your advice on research has been priceless. I would also like to thank my committee members, Professor Wang and Professor Xie for serving in my committee. I also want to thank you for letting my defense be an enjoyable moment, and for your brilliant comments and suggestions.

A special thanks to my family. Words cannot express how grateful I am to my mother, and father for being so encouraging and supportive throughout my life and college career. I owe a big thanks to their selfless love, caring and countless sacrifices they have made in order to allow me to be where I am today.

I would also like to thank my colleagues Ping-Lin Yang and Vivek Nandakumar for their support throughout my studies. Thank you to the GPGPUSIM support team for their continuous support during my project.

Special thanks to all of my friends who supported me in writing, and incented me to strive towards my goal.

VITA OF OMRI ALMOG
June 2015

EDUCATION

Master of Science in Electrical and Computer Engineering, University of California, Santa Barbara, June 2015 (expected)
Advisor: Prof. Malgorzata Marek-Sadowska

Bachelor of Science in Electrical and Computer Engineering, Oregon State University, June 2013

PROFESSIONAL EMPLOYMENT

2014-2015: Graduate Student Researcher, VLSI CAD Lab, University of California, Santa Barbara, CA

Summer 2011, 2012, 2013: Intern, Intel, Hillsboro, OR

Summer 2010: Intern, FLIR, Wilsonville, OR

Winter 2014: Teaching Assistant, Dept. of Physics, University of California, Santa Barbara, CA

2010-2012: Teaching Assistant, Dept. of Electrical and Computer Engineering, Oregon State University, Corvallis, OR

PUBLICATIONS

Almog, O., Meier, R., Kelly, N., & Chiang, P. (2014).A Piezoelectric Energy-Harvesting Shoe System for Podiatric Sensing Engineering in Medicine and Biology Society (EMBC), 2014 36th Annual International Conference of the IEEE.

ABSTRACT

Interconnect Fabric Reconfigurability for Network on Chip

by

Omri Almog

Microprocessor architectures are evolving at a pace greater than ever before. To meet the industry's stringent power, performance and cost demands there is a rising trend towards building heterogeneous processors with both CPU cores and off-chip components on the same chip. This is known as a System on Chip. These systems show promising solutions including chip interconnects consisting of Network on Chips (NoCs). These NoCs are composed of routers that control traffic, and channels used to connect different components of the chip itself together. Depending on the processor core's type, specifications, and technology used, the NoC fabrics may consume anywhere ranging from 28% to 40% of the total system power.

To reduce this significant power consumption, various solutions were proposed targeting CMOS technology. In this work we focus on NoC topology improvements and reconfigurability using novel VeSFET technology. The work deploys tools used to simulate full systems, such as GPGPUSIM, to evaluate the possible performance/power gains of a hybrid CMOS-VeSFET system. This hybrid system includes CMOS core and memory layers, while the NoC layer is made up of VeSFET transistors. This allows for shorter wire

lengths between routers and cores, as well as it permits for extra area to include network reconfigurability features.

The necessary modifications to build this hybrid system are area changes due to VeSFET additional layer, routing length changes, pipelining changes, and VeSFET technology parameter additions. The tools modifications necessary to include this system are described in further details in this thesis. The gathered data indicates great promise for the hybrid reconfigurable CMOS-VeSFET system over the conventional non-reconfigurable CMOS system. It is demonstrated that the hybrid VeSFET system has both a power decrease of approximately 57.0% and a performance increase of approximately 50.2%.

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1 - Introduction

1.1 Overview

With recent advancements of technology, and rising scale of on chip integration, it is possible to integrate a complete electronic system onto one chip. This is known as a System on Chip (SoC). The most promising solutions for chip interconnects are the Networks on Chip (NoCs). They are composed of routers and channels used to connect different components such as Cores, Memory, or other blocks [5]. Depending on the processor core's type, specifications, and technology used, the NoC fabrics may consume 28% to 40% of the total system power [4][6][19]. To reduce this significant power various solutions were proposed targeting CMOS technology [14][16]. In [15] a hybrid CMOS-VeSFET system was proposed. It was studied with emphasis on various features of the architecture. In this work we focus on the NoC topology improvements and reconfigurability features of that same architecture from [15].

Our aim is to evaluate the possible advantages in implementing the NoC using a new transistor technology called Vertical Slit Field Effect Transistors (VeSFETs). VeSFETs are novel twin gate and junctionless devices with terminals accessible from both sides of the device. VeSFET technology offers an attractive solution for 3D integration [12][15][17][18].

This thesis reports cycle by cycle simulation of power and performance performed on heterogeneous systems. There are four general system configurations that will be discussed in the following sections, each of which is of a different size. This is to show the scalability of the proposed improvements. Each of these system configurations is simulated for a CMOS NoC and a VeSFETs reconfigurable NoC. The reconfigurability of the VeSFET NoC

is implemented using switches in the network, giving the ability to change the topology of the network at runtime [14]. All system components are simulated using 65nm technology. The data is then extracted for the NoC allowing for the comparison of performance and power between the CMOS and VeSFET networks.

Due to cores and memory having a highly optimized modern design flow, it is a good idea to implement the NoC layer as VeSFET. When introducing the VeSFET hybrid system, we were able to decrease wire lengths not only between routers but also for router-to-core. [15] When creating this extra VeSFET NoC layer, we also observed that there is much extra area due to VeSFET routers consuming less area than CMOS routers. With this motivation, we were able to add some extra features to our NoC layer, and reconfigurability looked promising. Thus we decided to look into improvements of VeSFET NoC with reconfigurability included.

This thesis is organized as follows:

- In Chapter 2, we establish the basis of VeSFET technology and explain how it is possible to stack device layers manufactured in this technology. We then explain the benefits of including this technology into a reconfigurable 3D hybrid CMOS-VeSFET system. To accomplish this system we then explain the design behind reconfigurability and how by adding switches into the NoC layer we are able to create a reconfigurable topology network.
- In Chapter 3, we go into detail about the effects modeled in the simulators in order to establish the hybrid system. These effects include area changes due to the additional VeSFET layer as well as routing and pipelining changes. We then explain the systems modeled to collect the data necessary for comparison, as well as details about the applications run on the simulators.

- In Chapter 4, we provide the collected data of the experiments run, including tables and charts depicting these results. We then present the findings of the study.
- Chapter 5 concludes the thesis and presents future research directions.

CHAPTER 2 - NoC related improvements due to emerging trends in 3D stacking VeSFET technology

2.1 VeSFET Transistor and VeSFET based Circuits

2.1.1 The vertical slit transistor

The reconfigurable NoC layer in the studied architecture uses VeSFET transistors [6]. VeSFET is a square-shaped, twin gate, junctionless device that can be manufactured with silicon-on-insulator (SOI)-like process using conventional CMOS manufacturing steps [2].The unique geometry of VeSFET is shown in Figure 1. The diagonally positioned gate terminals on the opposite sides of a vertical slit region control the current flowing between the other two terminals, the source and the drain. VeSFETs can be of n- and p-type and can be used to construct CMOS-like ICs. Compared to a bulk 65nm CMOS transistor, VeSFET has a smaller driving current, smaller transistor capacitance, and lower power consumption[13].VeSFETs are manufactured as arrays of geometrically identical devices.

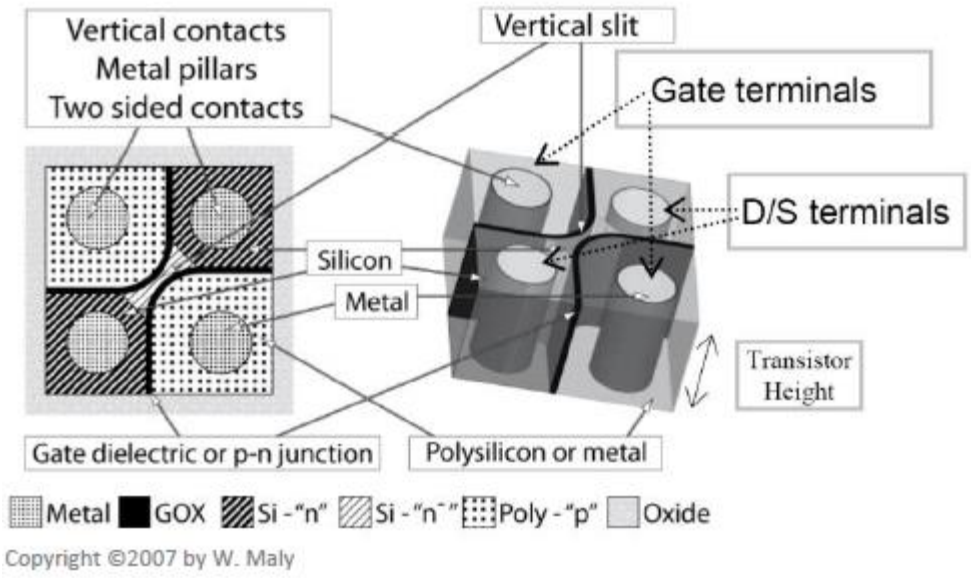

Figure 1. The VeSFET geometry [13]

2.1.2 3D stacking using VeSFET

For a given throughput, it is beneficial to shorten wires to reduce the number of pipeline stages. This can be accomplished with 3D chip architectures where the memory resides on top of the microprocessor layers [3][7]. These improvements require dense vertical communication at gate or transistor level as opposed to block level. In CMOS technology it is only possible to use two-layer face-to-face (F-to-F) 3D integration [10].Stacking more layers in a face-to-back (F-to-B) form would require very small pitch, high density and high yielding through silicon vias (TSVs) which is unfeasible today as discussed in [15]. Here, we study a 3D integrated hybrid circuit composed of two CMOS layers and one VeSFET device layer. This VeSFET device layer will include the switches needed for reconfigurability. We have a typical CMOS 2D architecture as the base case to compare the improvements of the reconfigurability.

2.1.3 CMOS - VeSFET hybrid 3D circuit

Figure 2 (a) shows the floorplan of a 2D CMOS implementation. Figure 2 (b) shows our studied 3D architecture that was proposed by [15]. The CMOS processor and memory nodes are on the top and bottom of the VeSFET NoC layer. In this implementation the router-to-router distance d_H is less than d, the router-to-router distance in the 2D_CMOS. In this architecture the three active layers are integrated without using TSVs. The intermediate VeSFET layer can make F-to-F connections to both the top and bottom layers as shown in Figure 2 (c).

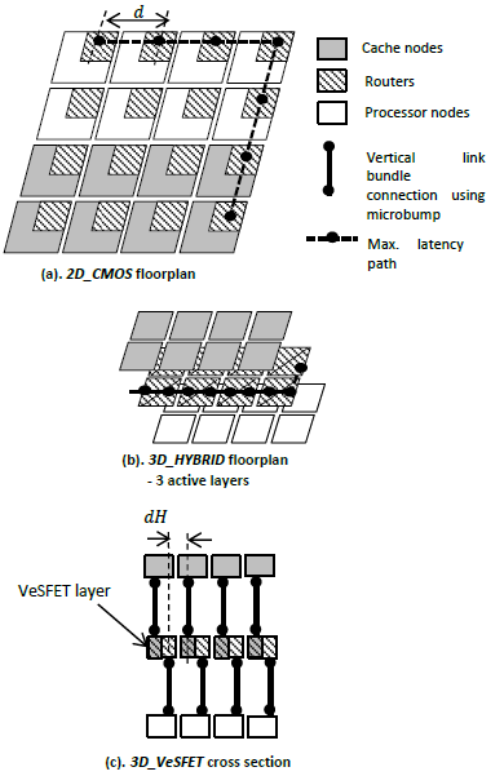

Figure 2. (a) CMOS Layout. (b) 3D Hybrid VeSFET Layout top down. (c) 3D Hybrid VeSFET Layout side view[15].

2.2 Reconfigurable network on chip

2.2.1 Switches - VeSFET& CMOS

Figure 3 shows a schematic and layout of both a simple VeSFET Switch as well as a CMOS Switch. Each switch can be configured to connect any two terminals together. This allows for the reconfiguration of the flow in the network, giving the ability to have different topologies reconfigured. The VeSFET switch is built from AND type transistors. When both gate pillars of an AND transistor are high for a p-type and both are low for an n-type the current flows between the source and drain. In the switch, only when N1 is high will the East node be connected to N5, and only when N3 is low will the South node be connected to N5. In this way we can configure the switch with N1-N4 to allow passage between the N, S, E, W nodes. In the case of the CMOS, this is achieved in the same manner but using six configuration transistors (1-6). As induced by the above design, not only will these switches be more compact when using VeSFET, they also only take 4 bits to configure.

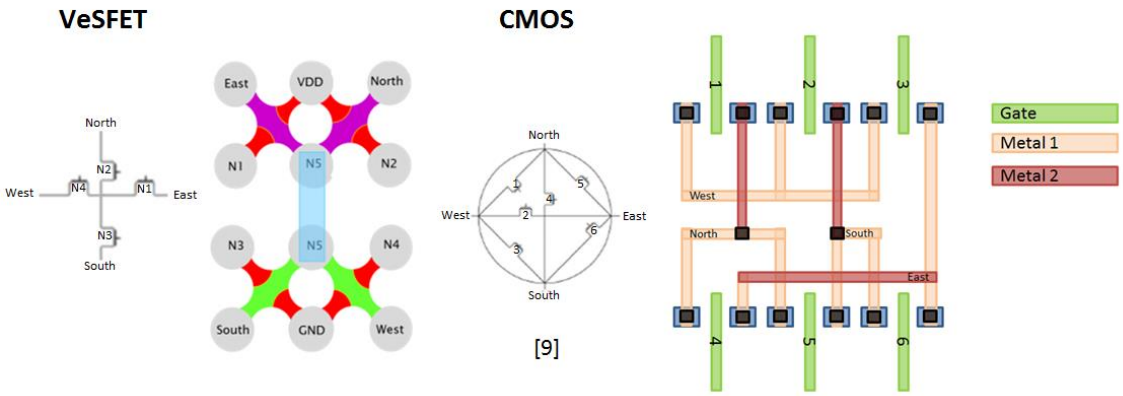

Figure 3. VeSFET Switch Layout Vs. CMOS Switch Layout

2.2.2 Reconfigurable topologies

With these switches it is possible to configure the network to have multiple topologies. Figures 4, 5, and 6 show the three different configurations we experimented with. Although the network topologies are shown for a 4x4 case, they can be extrapolated to a smaller or larger network. The added routing length between routers is only affected by the added area of the switches, and not by any detour path needed to get to the switches. The routing configuration is almost identical to the typical topology configuration without the switches added.

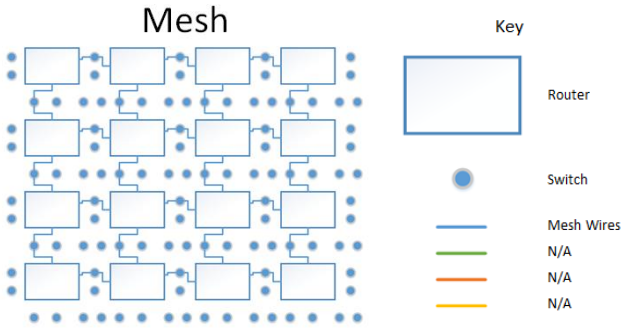

Figure 4. Mesh Reconfigurable Layout

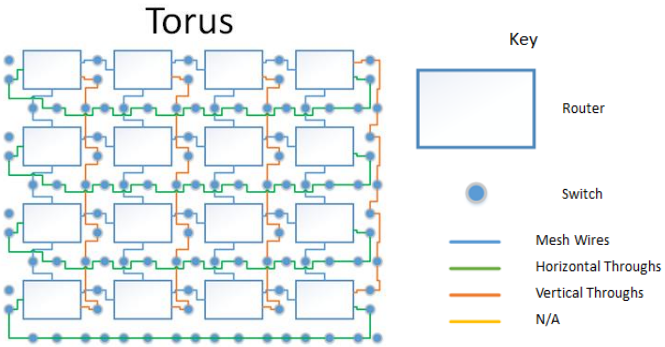

Figure 5. Torus Reconfigurable Layout

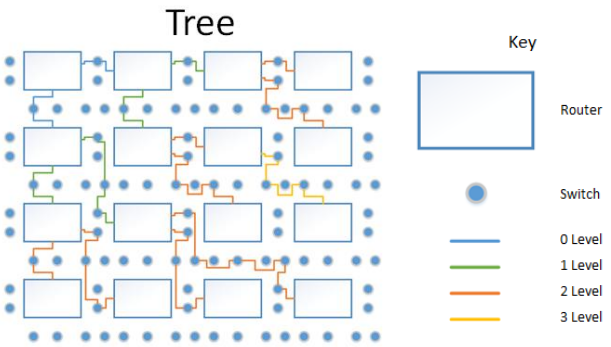

Figure 6. Tree Reconfigurable Layout (Levels referring to the tree depth levels)

With an algorithm that chooses the optimal topology for the current application, it is possible to reconfigure the network to that optimal topology. Much work has been done on choosing the topology for certain application types, and it has been proven that topology of the NoC affects the performance, and certain topologies are better for certain applications[5][9][14].

The objective of our work is to compare a 2DCMOS non-configurable and 3D CMOS-VeSFET hybrid reconfigurable implementations of the same system. We will assess the feasibility of adding reconfigurability to the VeSFET NoC and check if the hybrid implementation offers any advantages over the static CMOS network. Multiple NoC configurations will be tested with multiple applications. The experiment will also be testing scalability and topology advantages over a variety of applications.

CHAPTER 3 - Simulation and analysis of reconfigurable VeSFET NoC and non-reconfigurable CMOS NoC

3.1 Simulator changes to account for VeSFET network

The implementation of this experiment was done using a tool called GPGPUSIM [1]. This tool integrates GPUWattch[11], booksim[8], and cuda-sim to simulate applications running on a predefined system. GPUWattch is used to estimate the dynamic power of the system while it is running cycle by cycle, while booksim is used to do performance evaluation on the network aspect of the system. GPUWattch uses McPAT – an early stage design exploration tool for large multi-core processors to build the floorplans and estimate the power consumption.

A wrapper was also implemented around the tool to allow the user to choose what type of simulation is requested. The user has the option of choosing from an application integrated into the tool: topology to simulate, CMOS/VeSFET-Reconfigurable network, and system types integrated into the tool. All these configurations can be chosen from and the simulation will run and print out the performance/power results to a file.

3.1.1 Additional layer area changes

We study a hybrid 3D chip with VeSFET[13] implemented NoC as shown in Figure 8. Moving the routers and crossbars onto the VeSFET layer decreases the chip's footprint and reduces the distances between the NoC routers. The core section area of the system is reduced by approximately a 28% to 40%, and on the NoC layer, approximately 54% to 73% of the area would be unused in case of a static network implementation. In Figure 2 we can see that the VeSFET area of the core layer has shrunk with respect to the CMOS

configuration. Figure 8 visualizes the extra area we gain from moving the routers onto another layer. We can see that the NoC layer is smaller than the overall core and memory layers. The extra area around the routers can be utilized for reconfiguration of the NoC. Small switches added to the network allow for the connections between routers to be reconfigured at runtime[14]. This area is also used for connecting the routers and control signals going to the switches. As discussed in [14], it is also necessary to add storage space to keep the configuration information of the switches.

One possible control scheme for the switches can be implemented as depicted in Figure 7. This shows how the control unit sends out two control lines to the storage space of each switch for one of the routers. These signals are stored and decoded into the four control lines sent to each switch locally. There are other ways this can be designed, for example it is possible to send each of the control lines straight from the control unit to the switches and not have the storage distributed amongst the NoC.

One last note about the control signals is that this is shown for the VeSFET switches that require only 4 control signals. Not only does this require less wiring than CMOS, but it also requires less storage. In order to control the CMOS switches there will be more control signals required since there are six transistors involved with the CMOS switches.

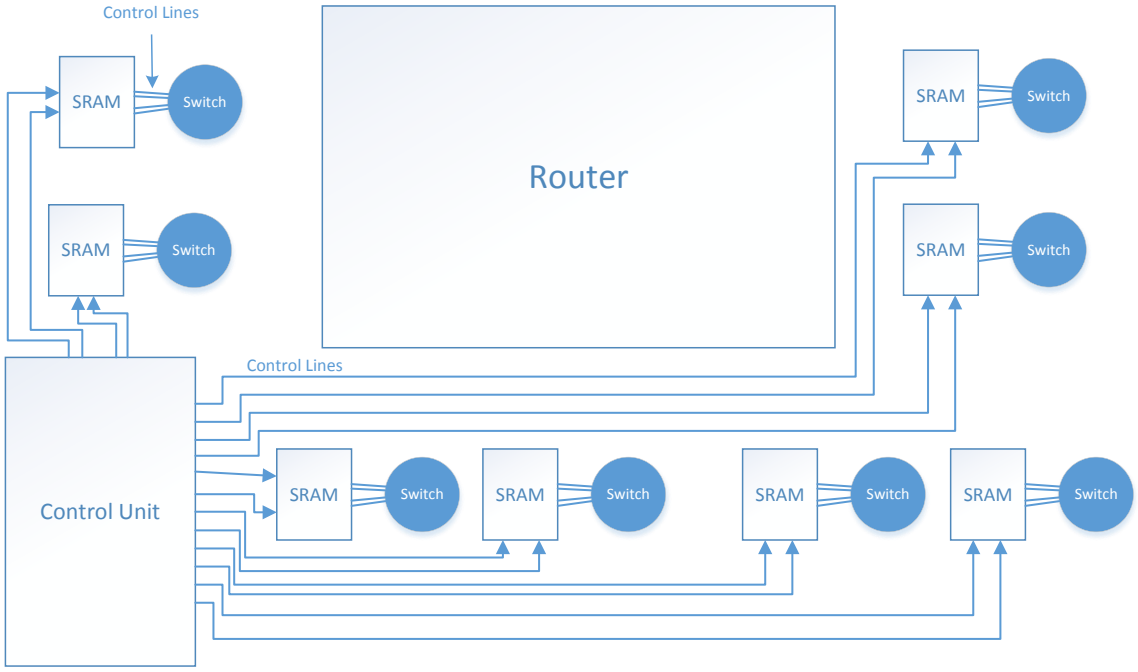

Figure 7. Potential Switch Control Scheme

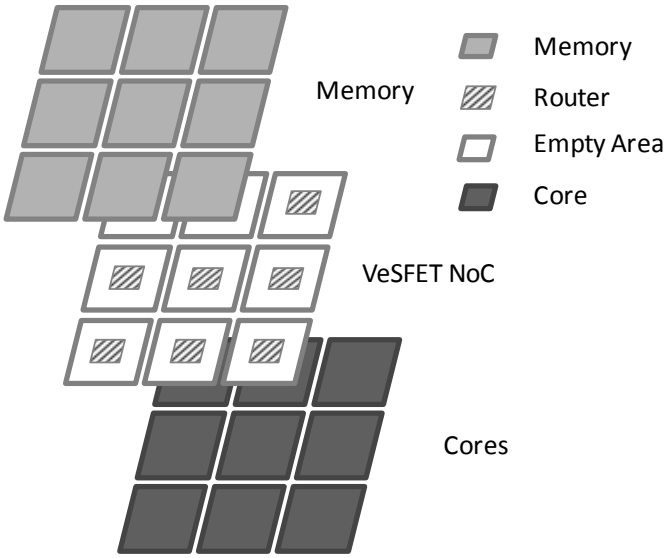

Figure 8. Rough visualization of extra area provided by VeSFET NoC

This VeSFET portion also has an additional overhead introduced to cover the additional area/power/length associated with adding the necessities to allow for reconfigurability of the network. To allow for a Mesh/Torus/Tree reconfigurable network, a corridor of width 2switches has been used in the network [14]. This corridor of width 2 can be visualized in Figure 9. This means that there are two switches in between every router. In [14], the authors analyzed networks including different variants of the reconfigurable network with switches. They conclude that reconfigurability with a corridor width of 2 requires approximately 68% area overhead [13– Figure 7]. As discussed previously, when a static VeSFET NoC is implemented, there is empty space that can be utilized for extra features. Since the VeSFET NoC has this extra space, we are able to negate this overhead by filling in the void with these required reconfigurability switches and wiring.

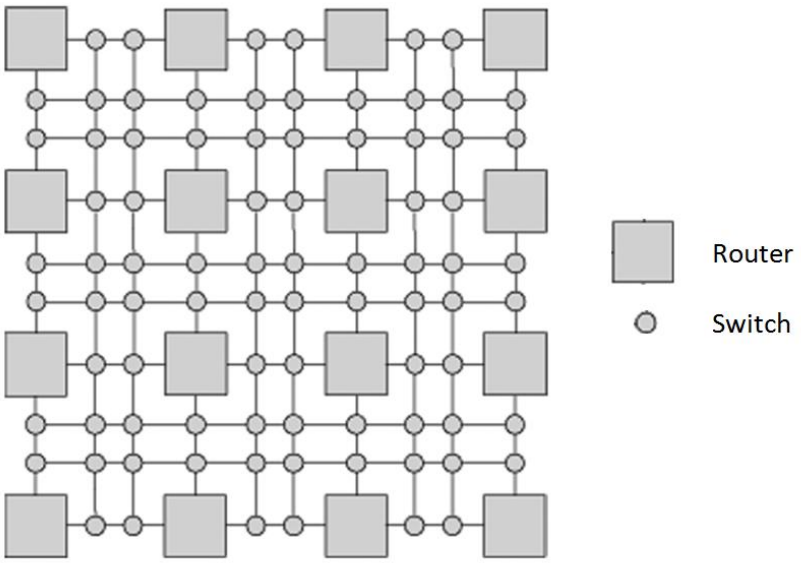

Figure 9. Visualization of the 2 corridor width added switches [14].

3.1.2 Routing changes &Pipelining changes

The following discussion has been taken from [15] and built upon in this thesis. Each of our routers in the four configurations has 5 ports. Four of the ports are for the router-to-router communication channels (North, South, East, West) and one of the ports is for the router-to-node communication. Each of these includes a 5x5 crossbar. The specifications of the routers are summarized in Table 1. We assume a common flit width of 128 bits. The routing wires of the 2D CMOS architecture are all in the horizontal plane, as shown in Figure 10 (a). In the 3D VeSFET-CMOS Hybrid system the router-to-node channel is vertical as shown in Figure 10 (b)[15].

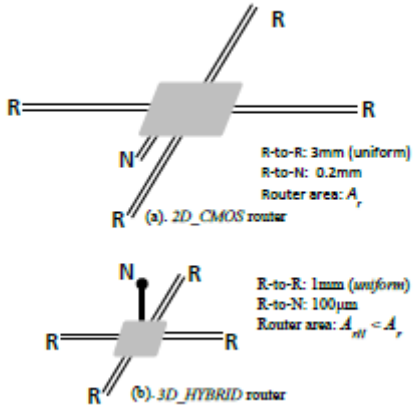

Figure 10. Router configurations [15]

The critical components of the router: input buffers, crossbar, router logic and arbiter are synthesized using RTL specifications from Stanford's Booksim group [8]. Synopsys IC compiler with 65nm CMOS technology is used to obtain delays of 75ps, 82ps, 65ps and 92ps for the above components respectively. The maximum target bandwidth is set to be 1.25Tbits/s for the NoCs that corresponds to a frequency of 10GHz and channel width of

128. The number of pipeline stages is determined for the inner router link using HSPICE simulations. More details can be found in [15].

Interconnect reduction with respect to 2D CMOS obtained by 3D VeSFET-CMOS Hybrid is reflected in the experiments. The reduced wire length between routers in 3D VeSFET-CMOS Hybrid not only translates to fewer pipeline stages but also reduces wire power. In all simulations we use 65nm technology parameters for CMOS and VeSFET device with pillar radius of 50 nm technology node. The horizontal router to router distance used for the base CMOS system is 3mm. The corresponding 3D VeSFET Hybrid distance used is d_H – 1mm. The router-to-node distance is also modified from the base CMOS system of 0.2mm to the 3D VeSFET Hybrid of 100um.Since $d_H < d$, there are fewer pipeline stages in the 3D Hybrid VeSFET NoC compared to our base 2D CMOS system as shown in Figure 11[15] to achieve a particular target frequency.

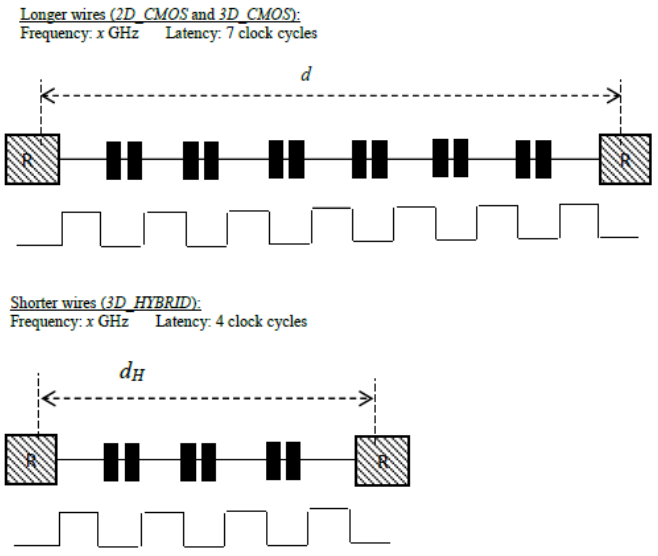

Figure 11. Pipeline stage reduction in 3D Hybrid VeSFET NoC[15]

3.2 Simulated systems

The simulated systems include 4 modified Sun Niagara II with additional cores/memory to show the scalability aspect of the experiment. Table 1 shows the various configurations of the experiment.

3.2.1 System parameters

TABLE 1. SYSTEM CONFIGURATIONS

Processor parameters:	
Case	**1/2/3/4**
Type	Sun Niagara II
Cores	**2/10/20/34**
Cores	3.16GHz
L1 cache	32KB dedicated; 4 way
L2 shared cache	2MB/cache tile
Memory Tiles	**1/3/8/15**
Router parameters:	
Ports	5
Technology	65nm
Flit width	128
Input buffer	Type: SRAM: 128x16; delay: 75ps
Network parameters:	
Type	2D Mesh/ 2D Torus/ 2D Tree
Size	**2x2/4x4/6x6/8x8**
Bandwidth	2.5 Tbits/sec

3.2.2 System configurations

The studied configurations begin with a 2x2 matrix consisting of 2 cores and 1 memory block with 2 memory sub-portions, and go up to an 8x8 matrix including 34 cores with 15 memory blocks. Each of these configurations also has six different NoC variants. These include CMOS -Mesh/Torus/Tree and VeSFET Hybrid -Mesh/Torus/Tree. This is to show the performance/power measurements for the reconfigurability. The VeSFET network

reconfigurability is included in this network. The CMOS does not provide for reconfigurability as it would increase the overall area of the chip. The VeSFET NoC layer does not incur an area increase due to reconfigurability. This is so because VeSFET layer has an extra area available for such features to be included.

Our architecture includes four multi-core processors including 2/10/20/34 cores. In all of these cases we use identical tiles of Sun Niagara II processors. Each of the cores in the base CMOS system includes a dedicated L1 cache. All core tiles share a common L2 cache. The architectural specifications of the processors are summarized in Table 1.

3.2.3 Applications

We simulated 2D CMOS non-reconfigurable networks for all the applications and topologies. These are used to compare to the reconfigurable VeSFET-CMOS hybrid. When comparing the two, first the size of the network is chosen. Then there are two options: one is to use the average performance/power over CMOS topologies for a group of applications, or to choose the best performance/power CMOS topology for each application. Once that topology is chosen for CMOS, the numbers for performance/power are extracted from the simulations. Once that is selected, an application to be compared is chosen.

The comparison is then made between the selected CMOS topology performance/power, and the best case VeSFET performance/power of the three topologies. The best case of the three topologies is chosen since it is possible to reconfigure the topology to any of the three. The best case of the three topologies can be chosen for the VeSFET system for each application, but once the CMOS topology is chosen, it is used for all the application comparisons since it cannot be reconfigured. Table 2 shows the current applications implemented into the tool with an explanation of the application and a description included.

TABLE 2. APPLICATIONS

Application	Functionality
Templates	**This sample is a templatized version of the template project. It also shows how to correctly templatize dynamically allocated shared memory arrays.**
vectorAdd	**Basic sample that implements element by element vector addition**
scalarProd	**Calculates scalar products of a given set of input vector pairs**
AtomicIntrinsics	**A simple demonstration of global memory atomic instructions**
matrixMul	**This sample implements matrix multiplication**
MultiGPU	**Use the new CUDA 4.0 API for CUDA context management and multi-threaded access to run CUDA kernels on multiple-GPUs.**
MonteCarloMultiGPU	**This sample evaluates fair call price for a given set of European options using the Monte Carlo approach, taking advantage of all CUDA-capable GPUs installed in the system.**
threadFenceReduction	**This sample shows how to perform a reduction operation on an array of values using the thread Fence intrinsic to produce a single value in a single kernel**

CHAPTER 4 - Simulation Results

4.1 Effects of interconnect reduction

Interconnect reduction with respect to the 2D CMOS system obtained by the 3D hybrid system is significant. The reduced wire length between the routers in the 3D hybrid system translates into fewer pipeline stages. This wire reduction comes from moving the NoC onto its own layer in a 3D stack. This horizontal distance between the routers and the nodes turns out to be approximately 100μm compared to 0.2mm in the 2D CMOS. The router-to-router length also gets compacted since there are no longer cores in between the routers. This translates to the 3D hybrid case containing 1mm router-to-router length whereas the CMOS case includes 3mm distance [15]. This interconnect reduction is part of the reason for the VeSFET power reduction and performance increase. The other main reason for this is the reconfigurability of the NoC.

4.2 Effects of reconfigurability

With the extra area provided by the NoC layer in the hybrid system, we are able to add the reconfigurable features to the system. With traditional non-reconfigurable networks, the set topology of the NoC cannot be changed as it is in physical hardware, but when we add the proposed solution of the switches, it is possible to reconfigure the topologies. We see the gains of this feature as it allows the NoC to reconfigure into the most optimized topology for the application that will run. These gains include both power and performance increases for the overall system due to the fact that different topologies are more optimized for certain applications compared to others.

4.3 Result tables

TABLE 3. MESH PERFORMANCE RESULTS

CASE1: 2X2 – 2 CORES 1 MEMORY
CASE 2: 4X4 – 10 CORES 3 MEMORY
CASE 3: 6X6- 20 CORES 8 MEMORY
CASE 4: 8X8 – 34 CORES 15 MEMORY

Mesh Ave # of cycles	Templates	vector Add	scalar Prod	AtomicI ntrinsics	matrix Mul	MultiGPU	MonteCarlo MultiGPU	threadFence Reduction
CMOS 2x2	13.37	28.81	28.72	80.70	40.10	21.76	60.61	26.32
VeSFET 2x2	10.61	19.08	18.30	58.22	27.89	13.34	54.18	16.00
CMOS 4x4	20.93	117.92	116.92	305.41	71.07	82.06	79.19	89.14
VeSFET 4x4	16.32	94.56	84.88	220.90	51.75	56.70	69.19	62.19
CMOS 6x6	24.68	190.35	173.88	483.52	105.85	51.68	94.88	55.91
VeSFET 6x6	19.09	152.28	132.65	363.85	77.21	31.64	72.18	34.80
CMOS 8x8	34.28	237.40	257.62	808.32	113.20	44.05	105.11	42.57
VeSFET 8x8	26.32	207.64	197.21	650.39	84.44	32.41	84.62	30.61

TABLE 4. TORUS PERFORMANCE RESULTS

Torus Ave # of cycles	Templates	vector Add	scalar Prod	AtomicI ntrinsics	matrix Mul	MultiGPU	MonteCarlo MultiGPU	threadFence Reduction
CMOS 2x2	15.13	30.46	31.91	102.71	42.55	23.71	61.86	28.35
VeSFET 2x2	12.44	20.02	21.71	81.61	31.80	16.19	55.27	18.61
CMOS 4x4	20.09	128.29	74.85	345.71	69.42	64.14	109.48	72.46
VeSFET 4x4	16.38	100.40	60.37	288.42	48.12	47.97	67.45	53.44
CMOS 6x6	26.41	103.64	95.66	593.63	100.40	45.30	77.31	50.06
VeSFET 6x6	21.46	94.82	68.25	483.92	69.29	31.16	61.81	34.82
CMOS 8x8	30.16	163.71	141.01	893.88	90.55	38.42	74.09	38.11
VeSFET 8x8	24.44	115.80	99.19	750.00	69.52	33.56	66.88	31.59

TABLE 5. TREE PERFORMANCE RESULTS

Tree Ave # of cycles	Templates	vector Add	scalar Prod	AtomicI ntrinsics	matrix Mul	MultiGPU	MonteCarlo MultiGPU	threadFence Reduction
CMOS 2x2	14.91	35.65	40.40	78.27	42.74	36.34	67.04	40.81
VeSFET 2x2	11.81	21.73	24.25	61.61	31.35	17.97	62.37	21.44
CMOS 4x4	14.91	106.14	95.59	324.26	58.25	65.02	72.65	72.89
VeSFET 4x4	11.81	96.99	62.63	269.66	41.23	39.27	60.62	45.15
CMOS 6x6	14.91	104.11	107.58	582.28	55.60	22.38	46.48	25.26
VeSFET 6x6	11.81	71.90	74.41	484.12	41.81	16.40	29.17	19.82
CMOS 8x8	14.91	104.58	86.20	1000.89	48.73	16.67	35.05	17.14
VeSFET 8x8	11.81	76.73	58.62	838.96	36.35	12.77	21.72	13.14

TABLE 6. MESH ENERGY RESULTS

Mesh Energy (pJ/bit)	Templates	vector Add	scalar Prod	AtomicI ntrinsics	matrix Mul	MultiGPU	MonteCarlo MultiGPU	threadFence Reduction
CMOS 2x2	0.35	0.72	0.74	2.02	1.00	0.56	1.52	0.67
VeSFET 2x2	0.11	0.20	0.19	0.59	0.28	0.14	0.55	0.17
CMOS 4x4	0.54	2.97	2.93	7.66	1.80	2.06	1.99	2.24
VeSFET 4x4	0.16	0.95	0.85	3.21	0.52	0.57	0.70	0.62
CMOS 6x6	0.63	4.78	4.37	10.10	2.66	1.31	2.38	1.41
VeSFET 6x6	0.20	1.52	1.34	5.65	0.78	0.32	0.73	0.35
CMOS 8x8	0.87	5.94	6.45	16.23	2.84	1.12	2.64	1.08
VeSFET 8x8	0.27	3.08	1.98	7.51	0.85	0.33	0.85	0.31

TABLE 7. TORUS ENERGY RESULTS

Torus Energy (pJ/bit)	Templates	vector Add	scalar Prod	AtomicI ntrinsics	matrix Mul	MultiGPU	MonteCarlo MultiGPU	threadFence Reduction
CMOS 2x2	0.39	0.77	0.82	2.58	1.07	0.61	1.56	0.71
VeSFET 2x2	0.13	0.21	0.22	0.82	0.32	0.17	0.56	0.20
CMOS 4x4	0.51	3.23	1.88	8.65	1.74	1.61	2.76	1.83
VeSFET 4x4	0.17	1.01	0.60	3.88	0.48	0.49	0.68	0.54
CMOS 6x6	0.66	2.60	2.39	19.85	2.53	1.13	1.94	1.28
VeSFET 6x6	0.22	0.96	0.69	5.85	0.70	0.32	0.62	0.35
CMOS 8x8	0.77	4.10	3.54	25.37	2.28	0.98	1.87	0.96
VeSFET 8x8	0.25	1.17	1.00	9.50	0.70	0.34	0.68	0.32

TABLE 8. TREE ENERGY RESULTS

Tree Energy (pJ/bit)	Templates	vector Add	scalar Prod	AtomicI ntrinsics	matrix Mul	MultiGPU	MonteCarlo MultiGPU	threadFence Reduction
CMOS 2x2	0.38	0.90	1.03	1.96	1.07	0.93	1.69	1.02
VeSFET 2x2	0.12	0.22	0.24	0.62	0.32	0.19	0.63	0.22
CMOS 4x4	0.38	2.67	2.41	8.11	1.46	1.64	1.82	1.83
VeSFET 4x4	0.12	0.98	0.63	2.70	0.42	0.40	0.62	0.46
CMOS 6x6	0.39	2.62	2.70	14.56	1.41	0.57	1.17	0.65
VeSFET 6x6	0.13	0.73	0.74	9.85	0.43	0.17	0.30	0.21
CMOS 8x8	0.39	2.63	2.17	25.04	1.23	0.43	0.89	0.45
VeSFET 8x8	0.12	0.78	0.59	11.40	0.37	0.13	0.22	0.14

TABLE 9. MESH POWER RESULTS

Mesh Ave Power (% NoC of full Chip)	Templates	vector Add	scalar Prod	AtomicI ntrinsics	matrix Mul	MultiGPU	MonteCarlo MultiGPU	threadFence Reduction
CMOS 2x2	0.91	7.81	6.29	22.01	1.70	9.76	1.49	11.43
VeSFET 2x2	0.51	4.43	2.79	15.36	0.84	2.00	0.11	0.87
CMOS 4x4	0.55	11.12	9.24	19.05	4.76	13.35	2.26	14.42
VeSFET 4x4	0.29	6.52	4.66	13.30	1.27	3.98	0.36	1.81
CMOS 6x6	0.35	11.85	9.68	14.78	5.13	14.44	2.42	16.72
VeSFET 6x6	0.19	7.08	5.06	10.35	1.64	5.10	0.64	2.68
CMOS 8x8	0.23	11.71	9.93	11.18	6.57	14.85	1.89	16.70
VeSFET 8x8	0.13	7.16	5.32	7.46	2.20	5.16	0.63	2.77

TABLE 10. TORUS POWER RESULTS

Torus Ave Power (% NoC of full Chip)	Templates	vector Add	scalar Prod	AtomicI ntrinsics	matrix Mul	MultiGPU	MonteCarlo MultiGPU	threadFence Reduction
CMOS 2x2	0.91	7.81	6.27	18.95	1.70	9.73	1.49	11.47
VeSFET 2x2	0.51	4.47	2.78	12.30	0.84	1.99	0.11	0.85
CMOS 4x4	0.55	11.18	9.62	20.11	5.04	13.93	2.29	15.33
VeSFET 4x4	0.30	6.47	4.72	13.00	1.19	4.03	0.37	1.83
CMOS 6x6	0.35	12.31	10.11	12.53	5.33	14.80	1.95	17.28
VeSFET 6x6	0.20	7.20	5.20	8.22	1.73	5.18	0.54	2.72
CMOS 8x8	0.23	12.09	10.24	12.67	6.15	15.70	1.97	17.73
VeSFET 8x8	0.13	7.39	5.51	8.25	2.15	5.35	0.65	2.88

TABLE 11. TREE POWER RESULTS

Tree Ave Power (% NoC of full Chip)	Templates	vector Add	scalar Prod	AtomicI ntrinsics	matrix Mul	MultiGPU	MonteCarlo MultiGPU	threadFence Reduction
CMOS 2x2	0.91	7.81	6.25	23.39	1.68	9.65	1.51	11.30
VeSFET 2x2	0.51	4.43	2.76	15.36	0.84	1.96	0.11	0.82
CMOS 4x4	0.51	11.11	9.33	23.63	4.81	13.81	2.30	15.36
VeSFET 4x4	0.29	6.53	4.78	15.33	1.27	4.31	0.38	2.11
CMOS 6x6	0.36	12.22	10.04	21.04	4.93	15.30	2.69	16.87
VeSFET 6x6	0.18	7.23	5.41	13.70	1.67	5.47	0.61	2.92
CMOS 8x8	0.22	12.35	10.45	19.96	6.04	15.42	2.35	17.07
VeSFET 8x8	0.12	7.72	5.89	13.17	2.27	5.56	0.83	3.08

4.4 Result charts

Performance

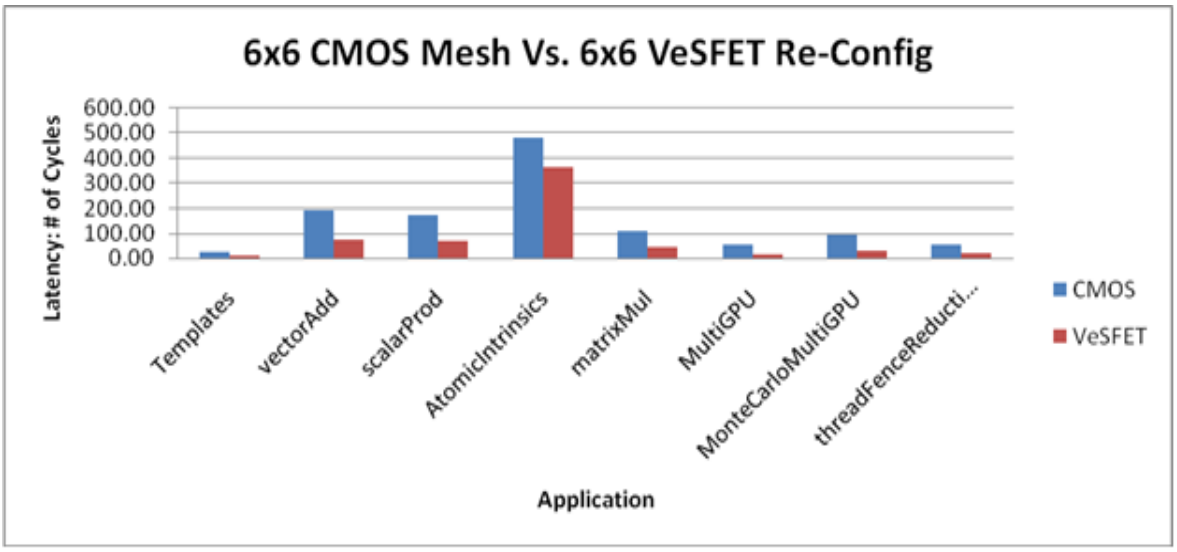

Figure 12.6x6 CMOS Mesh Vs. 6x6 VeSFET Re-Configurable Performance Chart

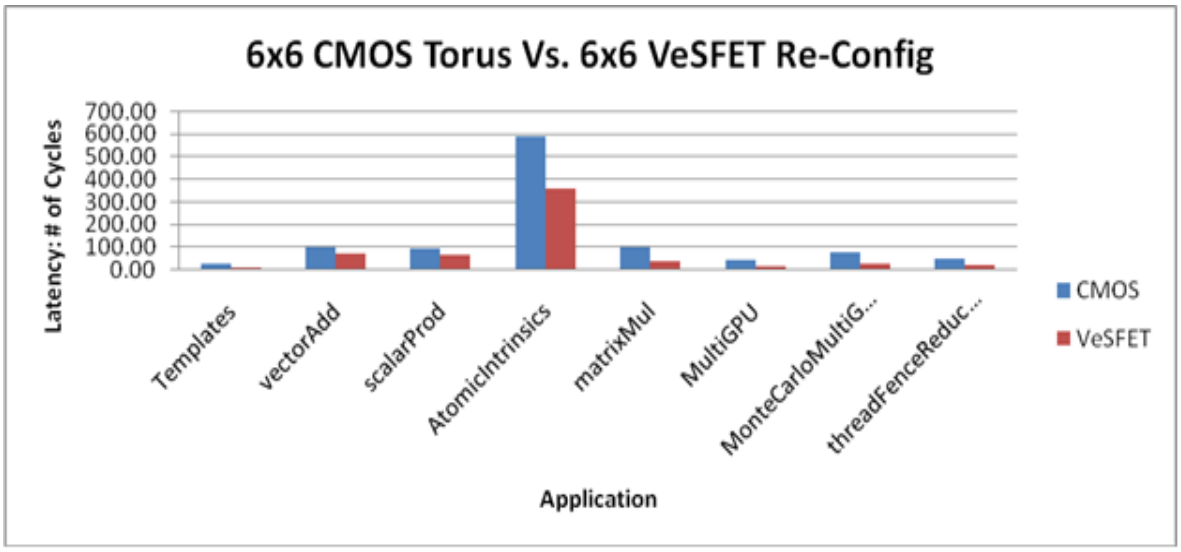

Figure 13.6x6 CMOS Torus Vs. 6x6 VeSFET Re-Configurable Performance Chart

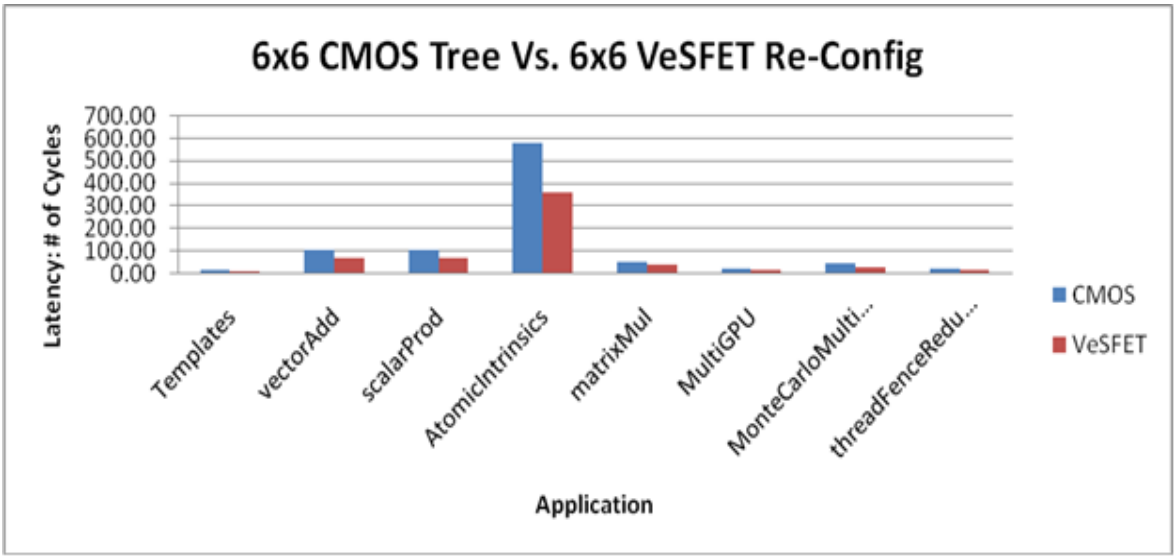

Figure 14.6x6 CMOS Tree Vs. 6x6 VeSFET Re-Configurable Performance Chart

Energy

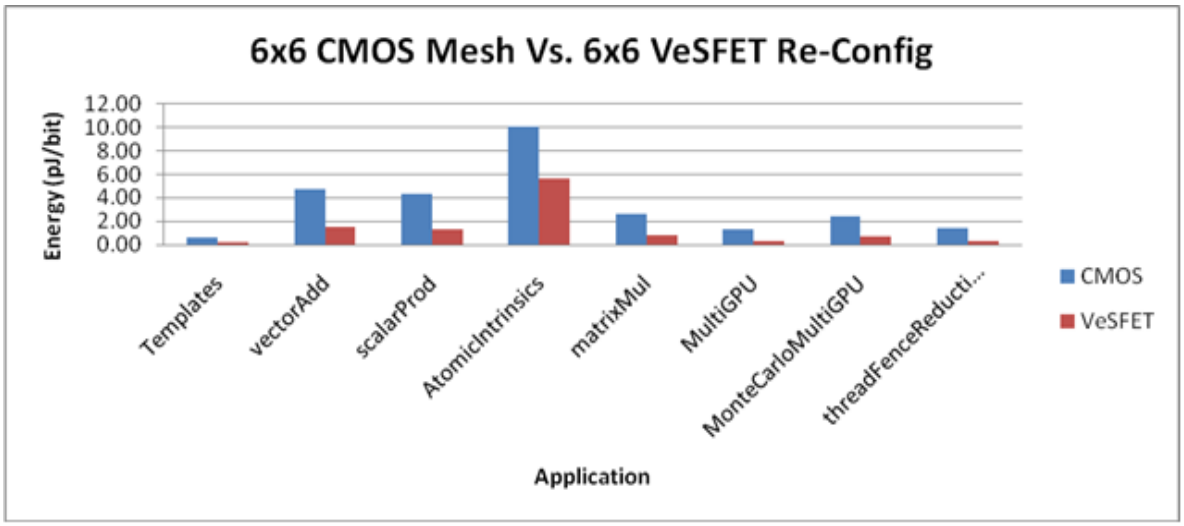

Figure 15.6x6 CMOS Mesh Vs. 6x6 VeSFET Re-Configurable Energy Chart

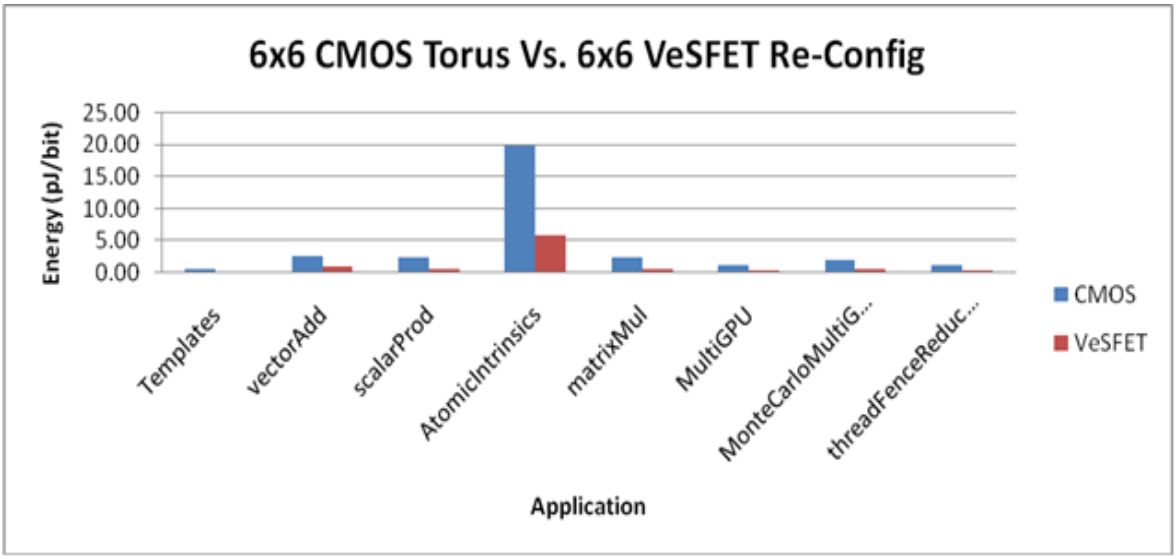

Figure 16.6x6 CMOS Torus Vs. 6x6 VeSFET Re-Configurable Energy Chart

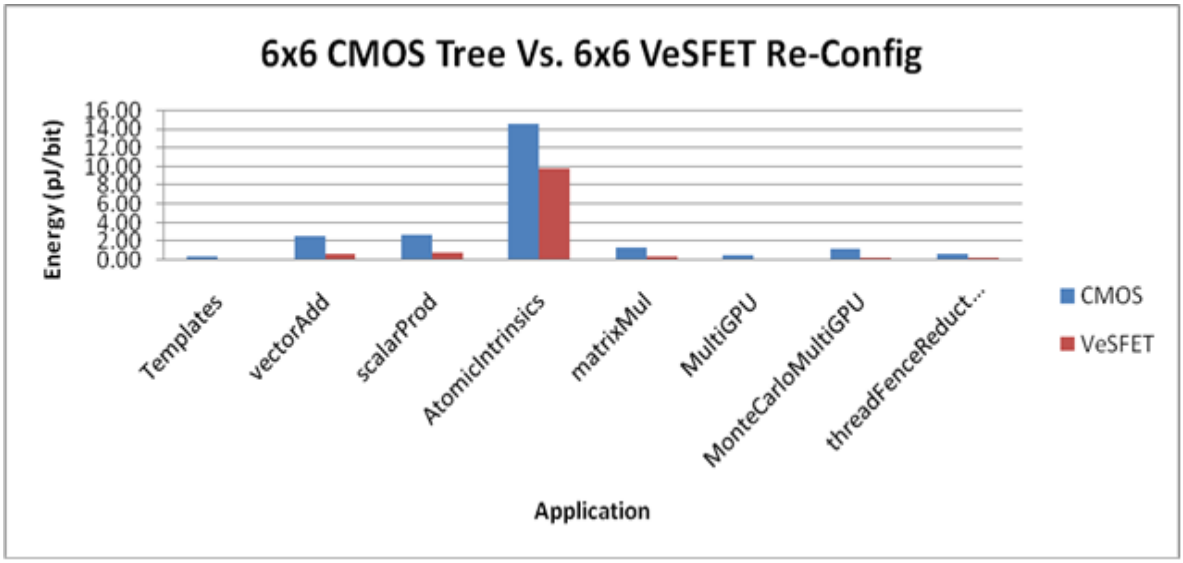

Figure 17.6x6 CMOS Tree Vs. 6x6 VeSFET Re-Configurable Energy Chart

Power

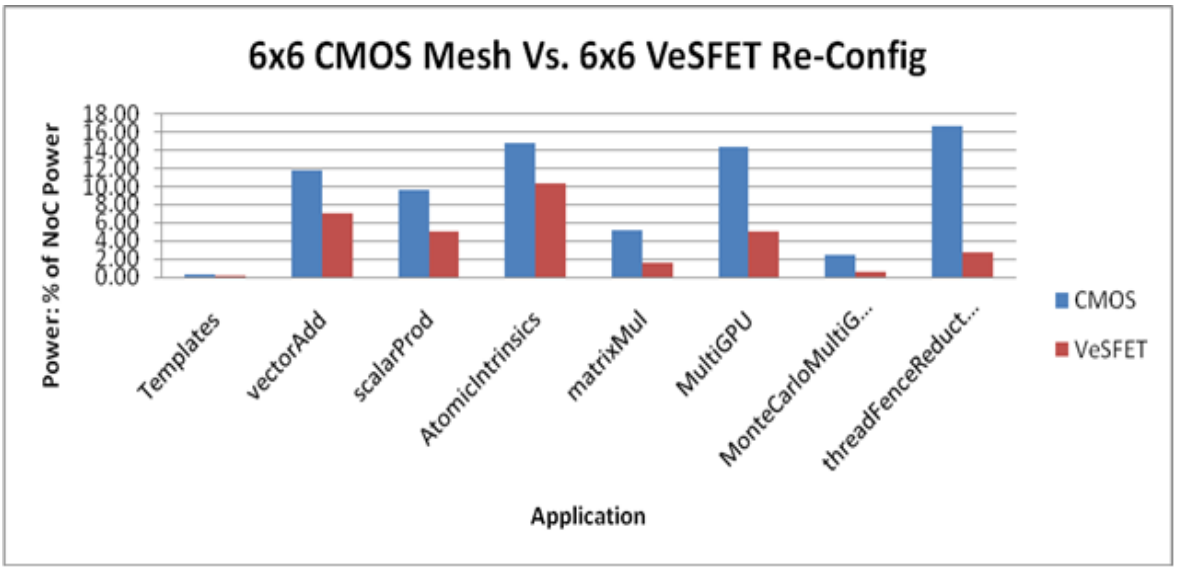

Figure 18.6x6 CMOS Mesh Vs. 6x6 VeSFET Re-Configurable Power Chart

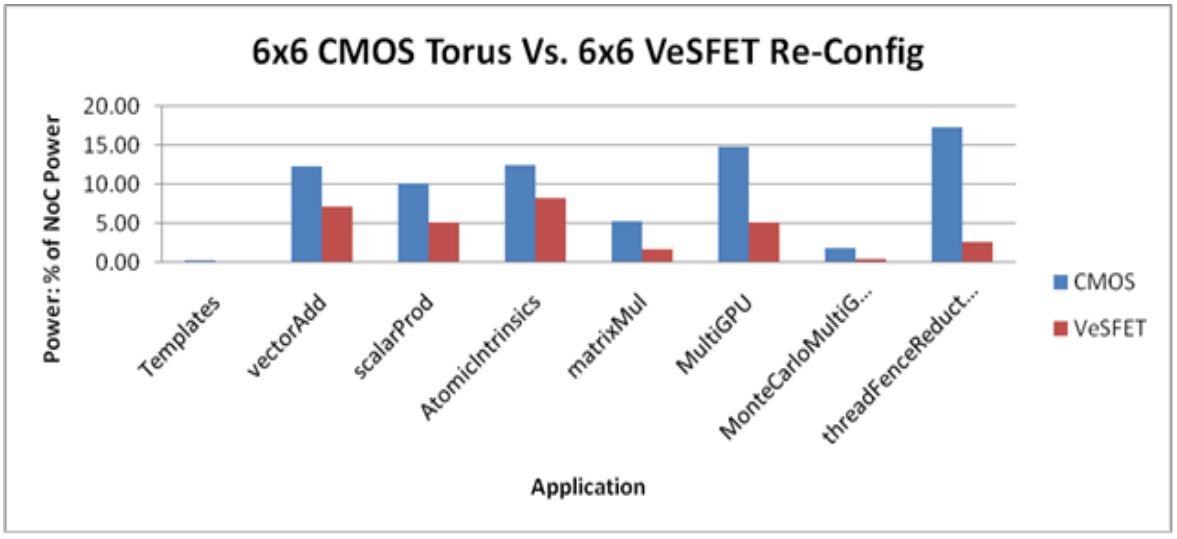

Figure 19.6x6 CMOS Torus Vs. 6x6 VeSFET Re-Configurable Power Chart

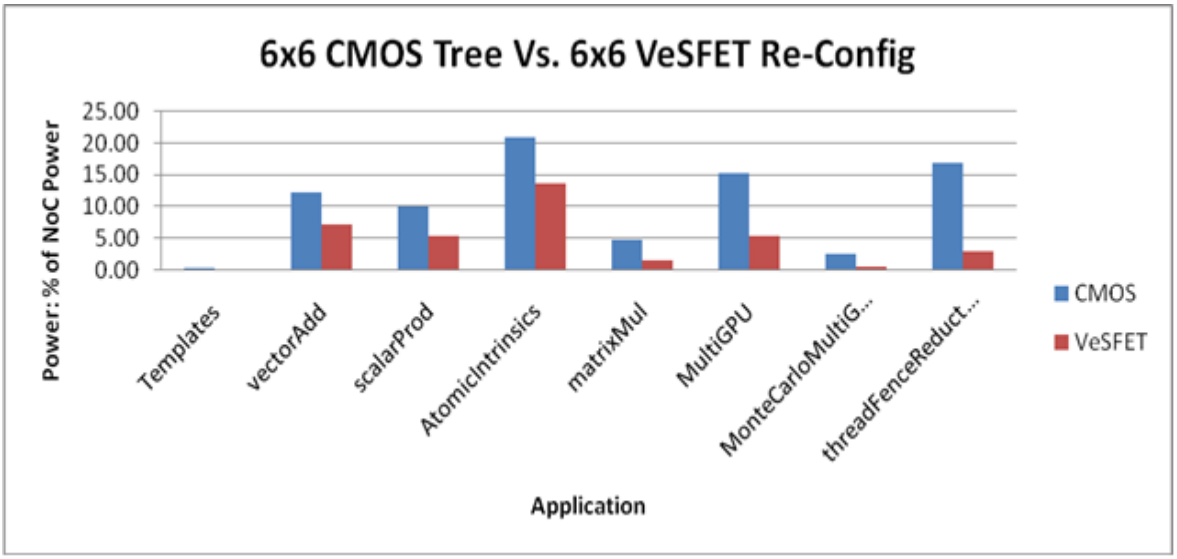

Figure 20.6x6 CMOS Tree Vs. 6x6 VeSFET Re-Configurable Power Chart

Scaling

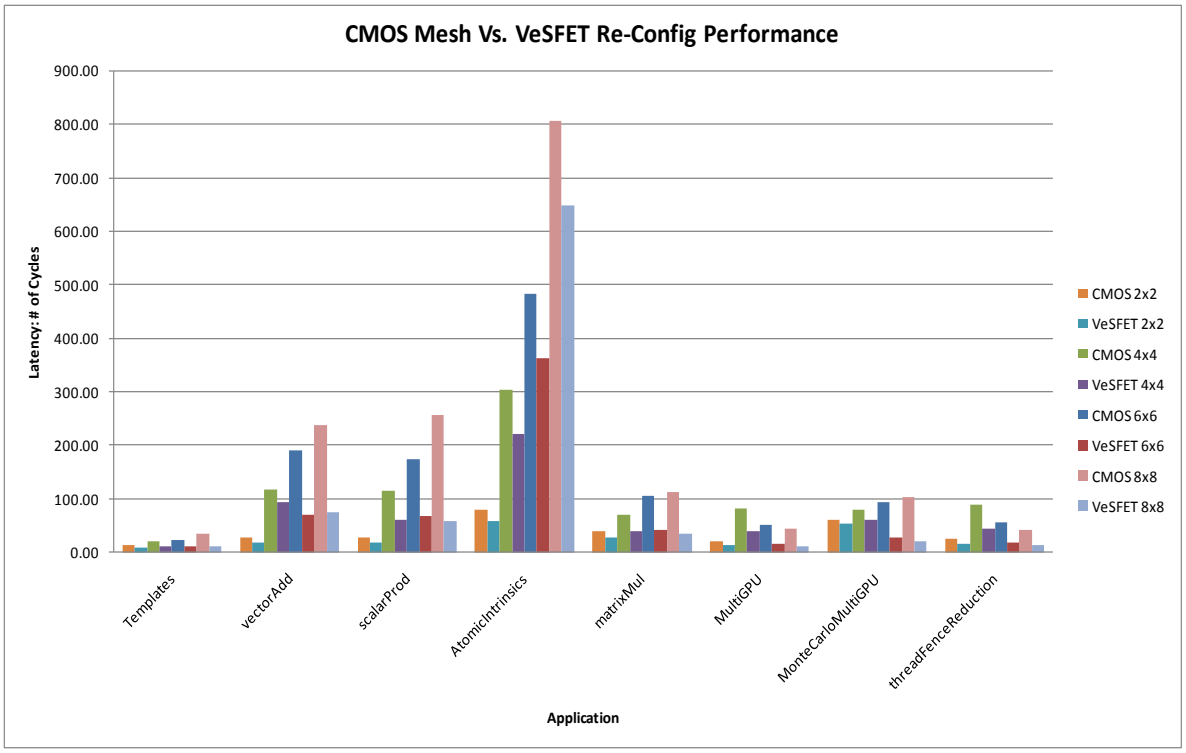

Figure 21. CMOS Mesh Vs. VeSFET Re-Configurable Performance Scalability

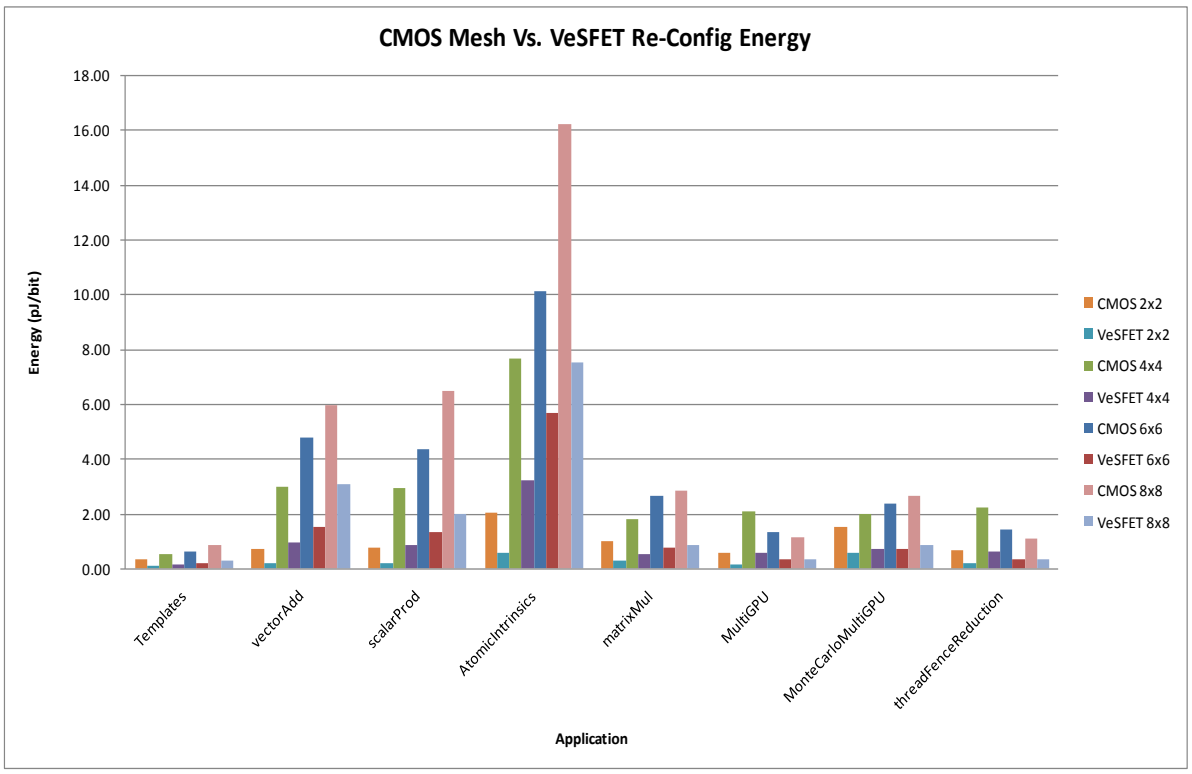

Figure 22. CMOS Mesh Vs. VeSFET Re-Configurable Energy Scalability

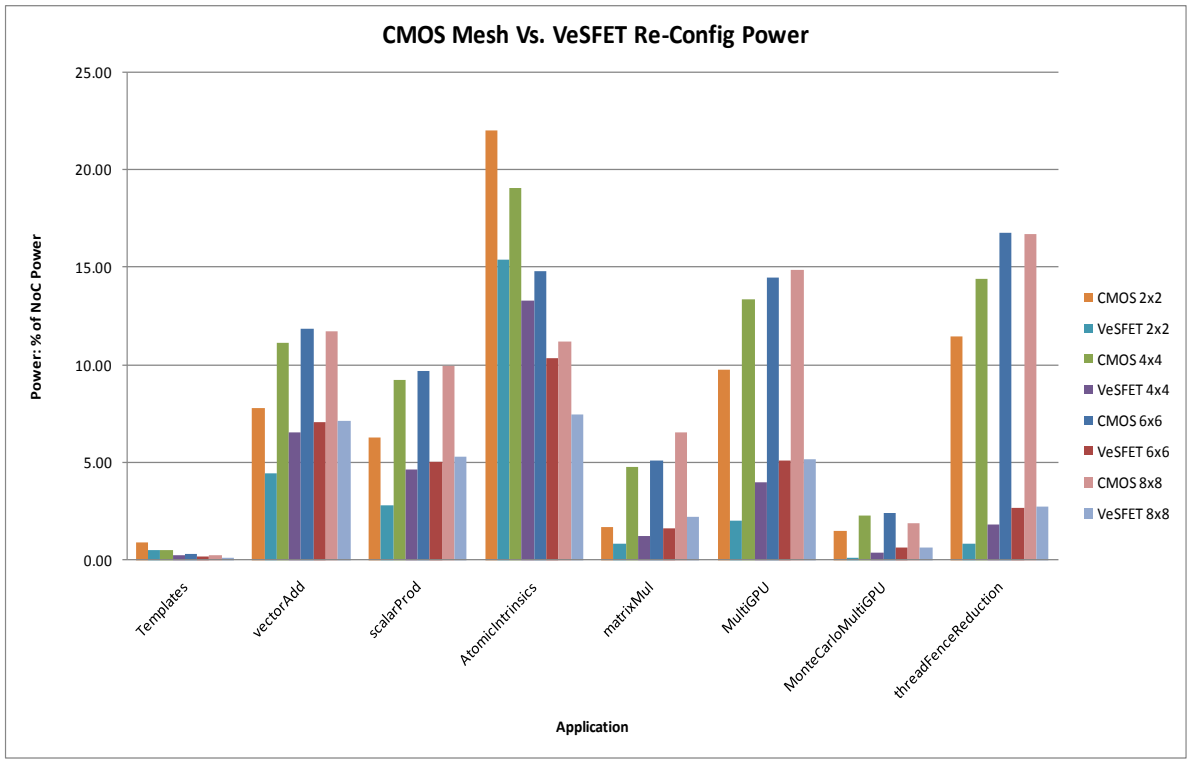

Figure 23. CMOS Mesh Vs. VeSFET Re-Configurable Power Scalability

4.5 Discussions

As demonstrated by the tables 3-5 and graphs 12-14, on average VeSFET has a 50.2% performance increase over the non-reconfigurable CMOS network. This performance increase is gained due to several factors. The VeSFET-CMOS hybrid is a 3D implementation allowing for shorter connections between the routers and shorter connections between the routers and nodes. The VeSFET reconfigurable network has the ability to match its topology to the application. It is possible to choose the best topology for the CMOS, for a certain application, but once the topology is set in the silicon it is not able to change. The VeSFET reconfigurability allows for the topology to update itself according to the application. This allows VeSFET to choose the topology that best fits the application, thus over the range of applications VeSFET has the increased performance.

These factors can also translate to the better power we are seeing. In the results presented, the power is represented as % of total NoC power. This is computed by dividing the NoC power by the total power of the system. As demonstrated by tables 6-11 and graphs 15-20VeSFET-CMOS hybrid has an overall 57.0% power decrease compared to 2D CMOS static implementation. This is caused by the system-level factors discussed above and is also caused by the VeSFET parameters themselves.

One interesting fact to note is that the AtomicIntrinsics application behaves in the opposite fashion than the other applications when looking at the overall results. A few possible factors that could lead to this behavior include the behavior of the application itself that is causing the network to be less efficient with smaller networks compared to larger networks. When implementing this application into this research, there were problems involving deadlock. These were fixed, but there could still be some underlying problems that

relate to these that are not visible. These could also be a cause of the different behavior of these results in scalability.

Charts 21-23 show the scalability of these networks. As the networks get bigger, the power and energy of the systems slightly increases as it takes more of it to move the packets through the network due to the network being larger. The performance also decreases slightly for the network due to the packets having to travel larger distances between routers/cores. Averaging the scalability of the overall CMOS system, and the overall Reconfigurable VeSFET system using result Tables 4-11, we are able to conclude that on average CMOS decreases 25.6% in performance when increasing the size of the network by 2 cores (in the x and y direction, ex: from 2x2 to 4x4). We are also able to conclude that VeSFET decreases on average 26.5%. Power on average increases 14.4% when scaling the CMOS system, and increases 14.9% when scaling the VeSFET Reconfigurable system. Overall CMOS and VeSFET scale in a similar fashion as observed in the data we have collected.

CHAPTER 5 - Conclusions

5.1 Conclusion

By analyzing the different topologies and comparing CMOS NoC layouts and3D VeSFET-CMOS hybrid with reconfigurable NoC we have demonstrated that reconfigurable VeSFET-based NoCs have an advantage over non-reconfigurable CMOS NoCs. It is demonstrated that both power and performance are improved in the VeSFET system compared to the non-reconfigurable CMOS system. These advantages are possible in the hybrid implementation with VeSFET layer allowing for face-to-face integration and

implementation of VeSFET-based reconfigurability. This is not feasible with CMOS due to large area (about 70%) overhead [14].

5.2 Future work

5.2.1 Addition of dynamic power gating

As described in the thesis, the VeSFET switches use AND-type VeSFET transistors. This means that both of the gate pillars need to be on in order for current to flow through source and drain. This shows a great promise as it is not necessary to include a power-gating transistor in order to shut off portions of the chip. It is possible to convert the NoC logic into these AND-type transistors and use one of the gates as a power-gate switch. There will be overhead involved in running the control lines to these gates. This is where the evaluation comes into place. Is it more power efficient to include large power-gating transistors in the circuit or is it more efficient to use one of the VeSFET gates in each of the logic transistors as a power-gate?

5.2.2 Additional 3D VeSFET NoC layers

In this thesis the 3D hybrid CMOS-VeSFET system included only one NoC VeSFET layer. As described in the above text, VeSFET can be stacked to multiple layers, as it is accessible via both from the top and bottom. Some current works are being done to simulate memory as a VeSFET layer. This idea of the NoC layer can be expanded to stacking multiple layers of NoC and memory. The bottom layer can still consist of CMOS, and going up from there it can alternate NoC, memory, NoC, memory etc. Each one of the NoC layers can also contain a different topology from the other. Modeling this comes to be very

difficult but it has the possibility of showing great improvements as the topology between L1 cache and L2 cache can vary within the chip itself for each application run.

5.2.3 Additional applications and topologies

This study contained a set list of applications and three topologies that were compared. This can be extended to comparing many more applications with different traffic loads as well as compare additional topologies to allow for greater increase in performance/power as the topologies can be tailored specifically to the applications. This allows the reconfigurable network to have topologies tailored to the applications whereas the traditional CMOS layer will still only be able to choose one optimal topology for the overall application list.

References

[1] A.Bakhoda, G. Yuan, W. W. L. Fung, H. Wong, T. M. Aamodt, "Analyzing CUDA Workloads Using a Detailed GPU Simulator," in IEEE International Symposium on Performance Analysis of Systems and Software (ISPASS), Boston, MA, April 19-21, 2009.

[2] L. Barbut, D. Bouvet, and J-M. Sallese. "Towards fabrication of Vertical Slit Field Effect Transistor (VeSFET) as new device for nano-scale CMOS technology." IEEE International Semiconductor Conference (CAS), Vol. 2., 2011.

[3] J. Bautista, "Tera-scale computing and interconnect challenges," Proceedings of the 45th annual Design Automation Conference. ACM, 2008.

[4] F. Clermidy, C. Bernard, R. Lemaire and J. Martin, "Low-power processors & communication" in ISSCC, 2010.

[5] N. Concerand M. A. Zamboni, "Design and Performance Evaluation of Network-on-Chip Communication Protocols and Architectures," PhD thesis, University of Bologna, 2009.

[6] Y. Hoskote, S. Vangal, A. Singh, N. Borkar and S. Borkar, "A 5-Ghz Mesh Interconnect for a Teraflops Processor," IEEEMICRO, 2007.

[7] Ph. Jacob, et al. "Mitigating memory wall effects in high-clock-rate and multicore CMOS 3-D processor memory stacks." Proceedings of the IEEE, 97.1 (2009): 108-122.

[8] N. Jiang, G. Michelogiannakis, D. Becker, B. Towels and W. J. Dally, "BookSim 2.0 User's guide", 2010.

[9] N. K. Kavaldjiev, "A run-time reconfigurable Network-on-Chip for streaming DSP applications." PhD thesis, University of Twente, 2007.

[10] S. J. Koester, A. M. Young, R. R. Yu, S. Purushothaman, K-N. Chen, D. C. La Tulipe, N. Rana, L. Shi, M. R. Wordeman and E. J. Sprogis, "Wafer-level 3D integration technology," IBM *Journal of Research and Development*,vol 52, no. 6, pp. 583-597, Nov 2008.

[11] J.Leng, T. Hetherington, A.ElTantawy, S. Gilani, N. Sung Kim, T. M. Aamodt, V.J.Reddi, "GPUWattch: Enabling Energy Optimizations in GPGPUs,"Proc. of the ACM/IEEE International Symposium on Computer Architecture (ISCA 2013), Tel-Aviv, Israel, June 23-27, 2013.

[12] W. Maly, Y.-W. Lin and M. Marek-Sadowska, "OPC-Free and Minimally Irregular IC Design Style," Design Automation Conference, pp.954-957, 4-8 June 2007.

[13] W. Maly, N. Singh, Z. Chen, N. Shen, X. Li, A. Pfitzner, D. Kasprowicz, W. Kuzmicz, Y. W. Lin and M. Marek-Sadowska, "Twin Gate, Vertical Slit FET (VeSFET) for Highly Periodic Layout and 3D Integration," MIXDES 2011, pp. 145–150

[14] M. Modarressi, A.Tavakkol, and H.Sarbazi-Azad. "Application-aware topology reconfiguration for on-chip networks." IEEE Transactions on Very Large Scale Integration (VLSI) Systems, 19.11 (2011): 2010-2022.

[15] V.S. Nandakumar, "Physically-Aware Architectural Exploration and Solutions for Heterogeneous Processors,"PhDThesis,Universityof California, Santa Barbara, 2014.

[16] Ch. Nicopoulos, "Network-on-Chip architectures: A holistic design exploration,"PhD thesis, The Pennsylvania State University, 2007.

[17] X. Qiu, M. Marek-Sadowska, “Assessing Circuit-level Properties of Vesfet-based ICs” 2013. Ph.D. Thesis, University of California at Santa Barbara, Santa Barbara, CA, USA

[18] V.M. Srivastava, N.Saubagya, G. Singh, "Circuit Design with Independent Double Gate Transistors," International Conference on Advances in Computer Engineering (ACE), pp.289-291, 20-21 June 2010

[19] M. Taylor, J Kim, J. Miller, D. Wentzlaff, F. Ghodrat, B. Greenwald, H. Hoffman, P. Johnson, J-W. Lee, W. Lee, A. Ma, A. Saraf, M. Senseski, N. Shnidman, V. Strumpen, M. Frank, S. Amarasinghe and A. Agarwal, “The raw microprocessor: a computational fabric for software circuits and general-purpose programs”,IEEEMICRO, IEEE, 2003.