View metadata, citation and similar papers at core.ac.uk
brought to you by CORE
provided by Infoscience - École polytechnique fédérale de Lausanne

Comparative Analysis of NoCs for Two-Dimensional Versus Three-Dimensional SoCs Supporting Multiple Voltage and Frequency Islands

Ciprian Seiculescu, Srinivasan Murali, Luca Benini, *Fellow, IEEE*, and Giovanni De Micheli, *Fellow, IEEE*

***Abstract*—In many of today's *system-on-chip* (SoC) designs, the cores are partitioned into multiple voltage and frequency islands (VFIs), and the global interconnect is implemented using a packet-switched network on chip (NoC). In such VFI-based designs, the benefits of 3-D integration in reducing the NoC power or delay are unclear, as a significant fraction of power is spent in link-level synchronization, and stacked designs may impose many synchronization boundaries. In this brief, we show the quantitative benefits of the 3-D technology on NoC power and delay values for such application-specific designs. We show a design flow for building application-specific NoCs for both 2-D and 3-D SoCs with multiple VFIs. We present a detailed case study of NoCs designed using the flow for a mobile platform. Our results show that power savings strongly depend on the number of VFIs used (up to 32% reduction). This motivates the need for an early architectural space exploration, as allowed by our flow. Our experiments also show that the reduction in delay is only marginal when moving from 2-D to 3-D systems (up to 11%), if both are designed efficiently.**

***Index Terms*—Networks on chip (NoCs), three-dimensional ICs, topology, voltage and frequency island (VFI).**

I. Introduction

THREE-DIMENSIONAL stacking is emerging as a promising integration option for systems on chip (SoCs) [1]–[3]. One of the major advantages of 3-D stacking is that global wire length could be much shorter than in 2-D systems. Long global wires in 2-D can be replaced by shorter local wires and efficient vertical interconnects, as the footprint of each layer and the distance between layers are small [1]. Networks on chip (NoCs) have recently evolved as the paradigm for designing the interconnect for both 2-D and 3-D systems [7], [12].

Distribution of clock trees is a major design challenge today. With advancing technology generations, the clock frequency and the design area increase, and only a portion of the chip can be covered in a single clock cycle [6]. For ease of design, many complex systems are partitioned into multiple *voltage and frequency islands* (VFIs). Each island is synchronous, using the same frequency and voltage lines. When the cores inside a VFI are not used for a particular application, the entire VFI can be shut down to save power. The cores in a VFI can be connected using a local interconnect (a local NoC). The local interconnects are then connected using a global NoC. The global NoC can itself be synchronous or asynchronous, and the latter case is called the *globally asynchronous, locally synchronous* (GALS) paradigm.

NoC links that cross from one VFI to another change frequency domains and, therefore, require a frequency converter. In 3-D systems, even if the whole design is synchronous, ensuring a zero-clock skew across different layers is difficult [29]. In this case, mesochronous synchronizers are needed for the vertical links. For example, in [29], an efficient design of such synchronizers for vertical links is presented.

In such VFI-based designs, the benefits of 3-D integration in reducing NoC power or delay are unclear. Earlier works either made comparisons using standard topologies (such as meshes) or did not consider VFI partitioning [12], [24], [25]. The objective of this work is to make a comparative study of NoCs for 2-D and 3-D implementation of SoCs. Our aim is to show quantitative benefits of the 3-D technology on NoC power and delay values. We show a design flow for building application-specific NoCs for both 2-D and 3-D SoCs with multiple VFIs. We present a detailed case study of NoCs designed using the flow for a mobile platform. Our results show that when the whole design is synchronous, 3-D designs give very low power savings (11%), as the mesochronous converters incur a lot of power overhead. As the number of VFIs increases, 3-D SoCs have large NoC power reductions (up to 32%) due to reduction in wire lengths. However, after a sweet spot, the gains fall again, as the wires get shorter in 2-D due to the use of more switches, and the contribution of converter power to overall power also becomes significant. Our results show the need for an early architectural design space exploration of the whole space, and our tools facilitate the same. Our experiments also show that the reduction in delay is only marginal when moving from 2-D to 3-D systems (up to 11%), if both are designed efficiently. This is because the number of links in 2-D that are long enough to require pipelining is less, and the frequency converter delay is dominant when compared with the wire delay.

Manuscript received October 26, 2009; revised February 10, 2010; accepted March 8, 2010. Date of current version May 14, 2010. This work was supported in part by the CTI under Project 10046.2 PFNM-NM and the Artist-Design Network of Excellence. This paper was recommended by Associate Editor V. Stojanovic.

C. Seiculescu and G. De Micheli are with the Integrated Systems Laboratory, Ecole Polytechnique Fédérale de Lausanne, 1015 Lausanne, Switzerland (e-mail: ciprian.seiculescu@epfl.ch; giovanni.demicheli@epfl.ch).

S. Murali is with the Integrated Systems Laboratory, Ecole Polytecnique Fédérale de Lausanne, 1015 Lausanne, Switzerland, and also with iNoCs, 1007 Lausanne, Switzerland (e-mail: murali@inocs.com).

L. Benini is with the Dipartimento Elettronica Informatica e Sistemistica (DEIS), University of Bologna, 40136 Bologna, Italy (e-mail: luca.benini@unibo.it).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCSII.2010.2047320

II. Related Work

An introduction to the NoC paradigm with its benefits is presented in [6] and [7]. The problem of synthesizing application-specific topologies for 2-D systems is investigated in [8]–[10]. Methods to shutdown VFIs, like power gating, are presented in [13]–[15]. These methods have to be used in conjunction with a topology synthesis algorithm that can generate NoCs that support VFI shutdown in order to achieve the actual shutdown of cores. Architectures for *GALS* NoC and for multisynchronous NoC are presented in [16] and [17]. In [18] and [19], methods to partition cores to VFIs and to build NoC-supporting VFIs are investigated. The design of GALS adapters for NoCs is presented in [20]. However, these works do not address 3-D design issues.

A description of 3-D integration technologies and methods for 3-D thermal-aware floor-planning are presented in [1]–[5]. Router architectures, evaluations of regular NoC topologies, and thermal-aware mapping methods on NoC topologies for 3-D ICs are described in [21]–[23]. Methods to design NoCs for 3-D ICs are presented in [11] and [12]. However, none of these works support the design of NoCs for 3-D designs with multiple VFIs.

In [24] and [25], the authors present power and latency comparisons between 2-D and 3-D NoCs. However, these works compare regular NoC topologies. While regular topologies are suitable for systems with homogeneous cores, most SoCs require application-specific custom NoC topologies with minimum-power delay overhead [10]. In [12], the authors make a comparison between application-specific NoC topologies for 2-D and 3-D ICs, but assume that the entire design is fully synchronous.

III. 3-D Architecture

We assume a 3-D manufacturing process based on the wafer-to-wafer bonding technology. Here, through silicon vias (TSVs) are used for establishing vertical interconnections. A vertical link requires a TSV macro on one of the layers (for example, the top layer), where the via cuts through the silicon wafer. In the bottom layer, the wires of the link will use a horizontal metal layer to reach the destination. For links that go through more than one layer, TSV macros are required in all the intermediate layers. However, it is important to note that the macros need not be aligned across the layers, as the horizontal metal layer can be used to reach the macro at each layer as well. Stacked TSVs are not used as the alignment of the TSVs would complicate floor planning. The area of the TSV macros for a particular link width is taken as input. For the synthesized topologies, our tool automatically places the TSV macros in the intermediate layers and on the corresponding switch ports. Our synthesis process automatically places the TSV macros at different layers for the different vertical interconnects.

IV. Design Approach

Here, we present a brief description of the method to synthesize NoCs for multiple VFIs. The algorithm takes as input the description of the application and the optimization objective and library of the area, which are the power models of the NoC components. The application description specifies information on the number of cores, their size, position, and VFI assignment. We also, optionally, take the input floor plan (without the NoC) to better estimate the power consumption and latency of wires. Our synthesis method automatically inserts the NoC components in the floor plan as needed. The application description also contains the communication description. We target embedded systems, where the tasks are usually statically mapped to physical cores. The bandwidth and latency requirements between the physical cores define the communication description, which is given as input to the synthesis algorithm. The optimization objective can be chosen between minimizing power and latency. The models of the NoC components are generated by synthesizing their register transfer level code for the targeted technology. These models are used to estimate power consumption, area, latency, and operating frequency. The design flow explores different design points by varying the number of switches and will design the best topologies that have different tradeoffs between power, area, and latency. An example of an output topology with the cores assigned to VFIs is presented in Fig. 1.

NoC architectural parameters, such as the width of the links, are varied, and the topology design process is repeated for each architectural point. In the following step, the number of switches needed to connect the cores is varied, and different topologies are synthesized. For a particular switch count, in the next steps, we determine the connectivity between the switches and the cores and the 3-D layer assignment of the switches.

The synthesis algorithm works as follows. We first find the minimum operating frequency in each VFI. Then, we calculate the minimum number of switches required in each VFI as a starting point. We vary the number of switches in each VFI and construct the best topology for each design point. To construct the topology, we assign the cores in each VFI to the switches in that VFI. When the cores are assigned to switches, the algorithm tries to assign cores in a VFI that have high bandwidth communication or tight latency constraints to the same switch. After this step, the algorithm finds paths for the flows between cores that are assigned to different switches. To allow for shutdown of VFIs, a restriction is imposed. A link between switches in different VFIs can only be opened if the source switch is in the same VFI as the core that initiates the communication flow, and the destination switch is in the same VFI as the core that is the target of the communication flow. A full description of the synthesis algorithm for 2-D is provided in [28]. After the routing step, a floor plan of the NoC is generated starting from the original positions of the cores given as input. The floor-planning routine tries to minimally affect the initial positions of the cores.

For designing 3-D NoC topologies, we also consider the number of TSVs that can be used between two layers as an additional constraint as in [11]. The algorithm only establishes as many vertical links as the maximum permitted by this TSV constraint.

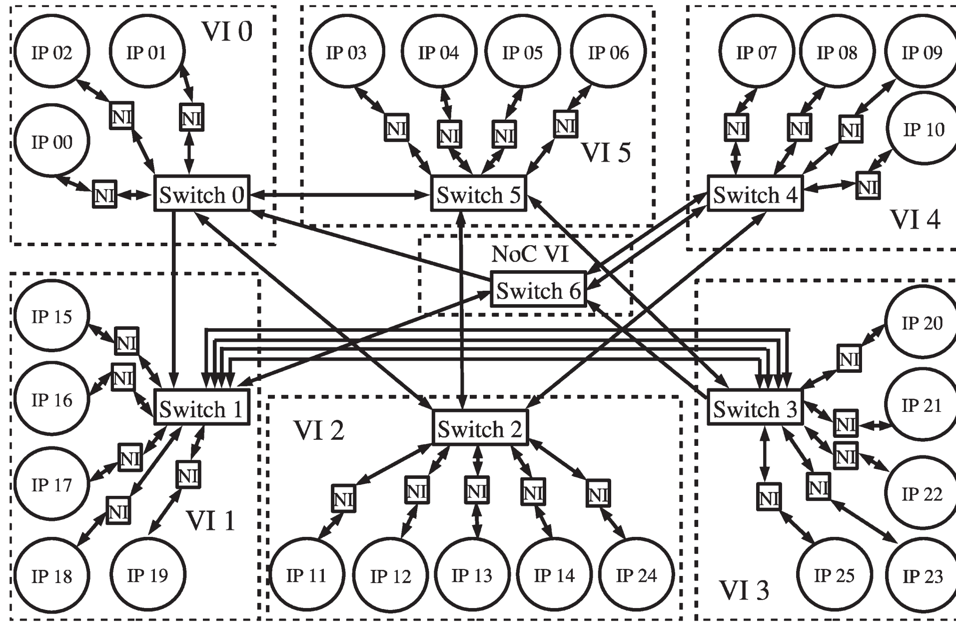

Fig. 1. Topology example.

TABLE I
NoC Component Figures

	Energy ($\mu W/MHz$)	Area (μm^2)	Freq (MHz)
switch 4x4	7.2	10000	803
switch 5x5	8.4	14000	795
1 mm 32bit wire	2.72		
Converter	0.34	1944	1000

V. Experiments and Case Studies

For the experiments, we use NoC components based on the architecture from [27]. To estimate the power and the area of the NoC components, we performed synthesis of them using a 65-nm technology library. For reference, the power consumption (with 100% switching activity), area, and maximum operating frequency for some of the components are presented in Table I. In [26], the authors show that the power consumption of tightly packed TSVs is smaller than that of horizontal interconnect by two orders of magnitude. Therefore, the impact of power consumption and delay of the vertical links is negligible, as they are very short as well (15–25 μm). Under zero-load conditions, the switch delay is 1 cycle, an unpipelined link delay is 1 cycle, and the worst case converter delay (which we use in the analysis) is 4 cycles (of the slowest clock) [30].

A. Comparison for a Different Number of VFIs

To compare the power difference between a 2-D and 3-D design, we use a realistic benchmark with 26 cores *(D26_Media)* that describes a multimedia and wireless communication SoC [12]. We assigned the cores to a different number of islands (from 1 to 7) based on logical connectivity and application constraints. For example, in the case of the two islands, the processor, the digital signal processor, and the hardware accelerators were assigned to the same VFI, and the memories and peripherals were assigned to the second VFI. An example of a topology for the six-island case produced by our methods is presented in Fig. 1.

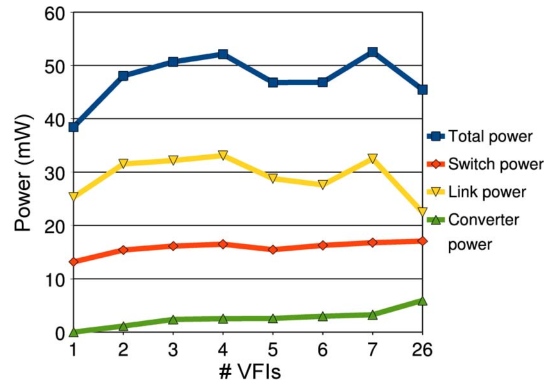

Fig. 2. Power 2-D designs.

As a reference, we consider the case where both the 2-D and 3-D designs are fully synchronous. The topology with the lowest power consumption in 2-D uses 38.5 mW, and in 3-D, it uses 30.9 mW. The minimum frequency at which the NoC has to be operated to support the bandwidth requirements of the benchmark is 270 MHz (power values are given for this frequency). The power consumption of the 3-D design is 20% lower when compared with the 2-D design. A complete analysis of 2-D and 3-D designs for fully synchronous designs is presented in [12]. We will show that with the extra constraints imposed by the VFIs, even larger power savings can be obtained for the 3-D designs.

The power consumption of the best power points for a different number of VFIs for the 2-D and 3-D cases are shown in Figs. 2 and 3. We show the total power consumption of the NoC as well as for the different components in the NoC (switches, links, and converters). It is to be noted that the operation frequency in each VFI is calculated based on the bandwidth requirements in that VFI. For that reason, when more VFIs are

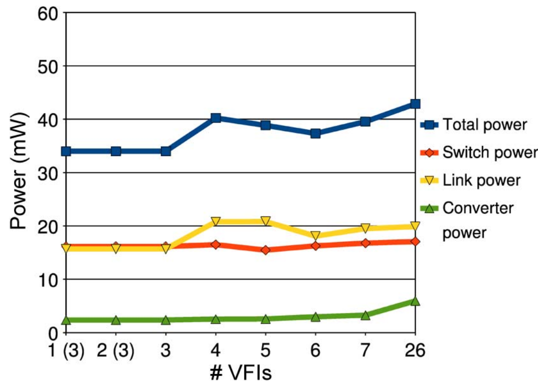

Fig. 3. Power 3-D designs.

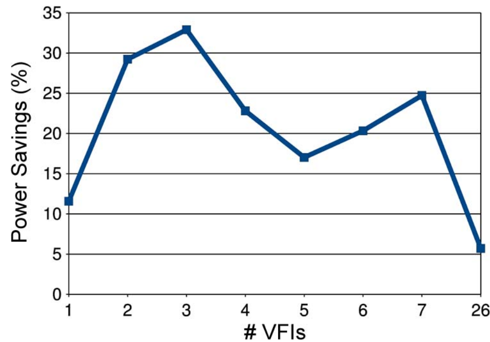

Fig. 4. Power savings of 3-D over 2-D designs.

added, some of them might operate at a lower frequency than required in a fully synchronous design. Although increasing the number of VFIs implies adding more resources like switches, links, and frequency converters, we can see that the power consumption does not go up significantly. This is due to the fact that with more VFIs in the design, a larger part of the NoC can be operated at a lower frequency. The relative power savings of 3-D compared with 2-D are shown in Fig. 4. In this experiment, we assume a clock skew across the different 3-D layers even for a fully synchronous design, thereby leading a minimum of three VFIs for 3-D, with one for each layer. Thus, the total power consumption plotted is the same for one to three VFIs in 3-D. We obtain a maximum power saving for three VFIs. This is because, in 3-D, we have a minimum of three VFIs, and as we increase the number of VFIs, the converter power consumption increases, and also wires in 2-D are more segmented and shorter. The zero-load latency of the designs with different numbers of VFIs is presented in Fig. 5. We can see that the latency goes up with the number of VFIs because more links use frequency converters (which incur a four-clock-cycle penalty to traverse) and because more islands are operated at a lower frequency.

In Fig. 6, we show the power savings obtained on two other benchmarks, with different communication patterns than the multimedia system considered above. The *D35_bott* benchmark has 16 processors with 16 private memories and 3 shared memories. Most of the high bandwidth traffic is between the processors and their private memories. On the other hand, *D36_8* is a benchmark with a spread traffic pattern, with each core communicating with eight others with equal bandwidth values.

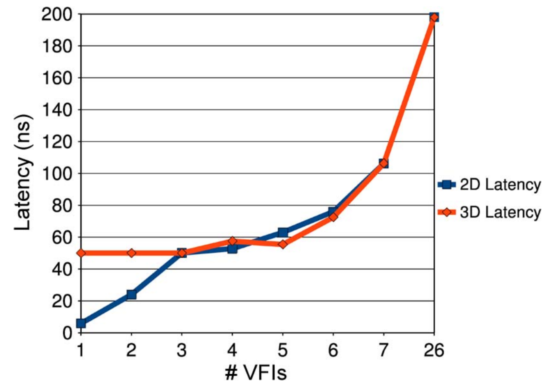

Fig. 5. Average zero-load latency of 2-D and 3-D designs.

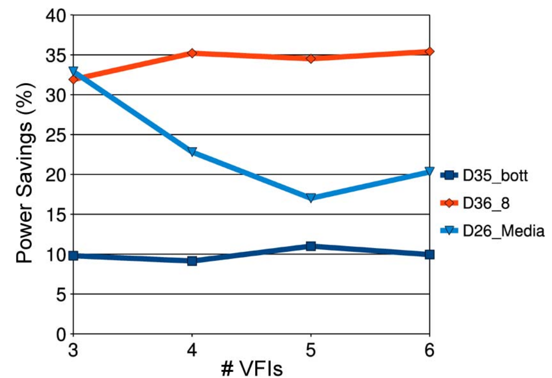

Fig. 6. Power savings of 3-D versus 2-D for different benchmarks.

As expected, these two benchmarks represent two extremes: the former providing low power savings, while the latter providing large power savings for 3-D. As a reference, the topologies for *D35_bott* with three VFIs consume 80 mW in 3-D and 88 mW in 2-D. The previously analyzed benchmark (the *D26_Media*) is a realistic benchmark, and the power savings are in between these two benchmarks. Also, the power savings depend on the number of VFIs.

The number of VFIs can be decided according to the design in order to achieve low power consumption by the shutdown of unused VFIs. However, it can also be forced by the technology when the area of the design becomes large enough that a single clock tree cannot be designed to synchronize all the components. We performed experiments to explore the effect of increasing the number of VFIs due to an increase in the size of the design. We scale the size of the design proportional to the number of VFIs. For example, a design with five VFIs has an area increase of 60% over the one with three VFIs. The difference in power consumption between the 3-D and 2-D designs in percentage is shown in Fig. 7. As the wires are longer when the benchmark is larger, we obtain more power savings in 3-D when compared with the experiments in the previous subsection.

B. Analysis of Results

When the whole design is synchronous, 3-D designs give very low power savings (11%), as frequency converters or mesochronous synchronizers are needed to tolerate clock skew

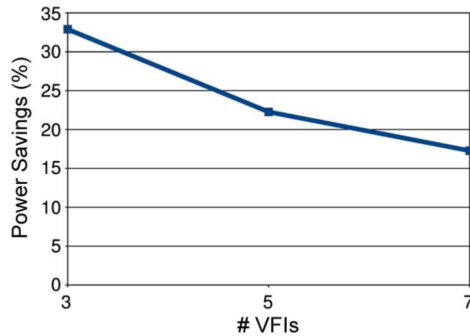

Fig. 7. Power savings of 3-D versus 2-D for different core areas.

across layers. As the number of VFIs increases, 3-D SoCs have large NoC power reductions (up to 32%) due to reduction in wire lengths. However, after a sweet spot, the gains fall again, as the wires get shorter in 2-D due to the use of more switches, and the contribution of the converter power to the overall power also becomes significant. Our results show the need for an early architectural exploration of the whole design space, as the number of VFIs used plays a major role in determining the power savings achieved when migrating to 3-D. Our experiments also show that the reduction in delay is not very significant when moving from 2-D to 3-D systems (up to 11%), if both are designed efficiently. This is because the number of links in 2-D that are long enough to require pipelining is less, and the synchronizer delay is more dominant when compared with the wire delay. The area overhead due to the insertion of TSVs in 3-D is negligible, as the TSV macros occupy less than 2% area when compared with the area of the cores.

VI. Conclusion

In many of today's chips, the design is partitioned into multiple *VFIs*. In such a system, it is not clear if the interconnect can benefit by using 3-D technology. In this brief, we have presented a detailed comparison of NoCs for 2-D and 3-D using a realistic mobile platform. We have shown that as the number of VFIs increases, 3-D SoCs have large NoC power reductions (up to 32%) due to reduction in wire lengths. We have also shown that after a certain number of VFIs, the gains fall again due to shorter wires in 2-D and more significant converter power consumption in both cases. Our experiments have also shown that the reduction in delay is minimal when moving from 2-D to 3-D systems (up to 11%) because the converter delay is more dominant than the wire delay.

References

[1] K. Banerjee, S. J. Souri, P. Kapur, and K. C. Saraswat, "3-D ICs: A novel chip design for deep-submicrometer interconnect performance and systems-on-chip integration," *Proc. IEEE*, vol. 89, no. 5, pp. 602–633, May 2001.

[2] J. Cong, J. Wei, and Y. Zhang, "A thermal-driven floorplanning algorithm for 3D ICs," in *Proc. ICCAD*, Nov. 2004, pp. 306–313.

[3] W.-L. Hung, G. M. Link, Y. Xie, N. Vijaykrishnan, and M. J. Irwin, "Interconnect and thermal-aware floorplanning for 3D microprocessors," in *Proc. ISQED*, Mar. 2006, pp. 98–104.

[4] C. Guedj, N. Claret, V. Arnal, M. Aimadeddine, and J. P. Barnes, "Evidence for 3D/2D transition in advanced interconnects," in *Proc. IRPS*, 2006, pp. 64–68.

[5] R. Weerasekera, L.-R. Zheng, D. Pamunuwa, and H. Tenhunen, "Extending systems-on-chip to the third dimension: Performance, cost and technological tradeoffs," in *Proc. ICCAD*, 2007, pp. 212–219.

[6] L. Benini and G. De Micheli, "Networks on chips: A new SoC paradigm," *Computers*, vol. 35, no. 1, pp. 70–78, Jan. 2002.

[7] G. De Micheli and L. Benini, *Networks on Chips: Technology and Tools,* 1st ed. San Mateo, CA: Morgan Kaufmann, Jul. 2006.

[8] A. Pinto, L. P. Carloni, and A. L. Sangiovanni-Vincentelli, "Efficient synthesis of networks on chip," in *Proc. ICCD*, Oct. 2003, pp. 146–150.

[9] K. Srinivasan, K. S. Chatha, and G. Konjevod, "An automated technique for topology and route generation of application specific on-chip interconnection networks," in *Proc. ICCAD*, 2005, pp. 231–237.

[10] S. Murali, P. Meloni, F. Angiolini, D. Atienza, S. Carta, L. Benini, G. De Micheli, and L. Raffo, "Designing application-specific networks on chips with floorplan information," in *Proc. ICCAD*, 2006, pp. 355–362.

[11] S. Murali, C. Seiculescu, L. Benini, and G. De Micheli, "Synthesis of networks on chips for 3D systems on chips," in *Proc. ASPDAC*, 2009, pp. 242–247.

[12] C. Seiculescu, S. Murali, L. Benini, and G. De Micheli, "SunFloor 3D: A tool for networks on chip topology synthesis for 3D systems on chip," in *Proc. DATE*, 2009, pp. 9–14.

[13] D. Lackey, P. S. Zuchowski, T. R. Bednar, D. W. Stout, S. W. Gould, and J. M. Cohn, "Managing power and performance for System-on-Chip designs using Voltage Islands," in *Proc. ICCAD*, 2002, pp. 195–202.

[14] F. Fallah and M. Pedram, "Standby and active leakage current control and minimization in CMOS VLSI circuits," *IEICE Trans. Electron.*, vol. E88-C, no. 4, pp. 509–519, Apr. 2005.

[15] Q. Ma and E. F. Y. Young, "Voltage island driven floorplanning," in *Proc. ICCAD*, 2007, pp. 644–649.

[16] T. Bjerregaard, S. Mahadevan, R. G. Olsen, and J. Sparsoe, "An OCP compliant network adapter for GALS-based SoC design using the MANGO network-on-chip," in *Proc. Int. Symp. Syst.-on-Chip*, Nov. 17, 2005, pp. 171–174.

[17] I. Miro-Panades, F. Clermidy, P. Vivet, and A. Greiner, "Physical implementation of the DSPIN network-on-chip in the FAUST architecture," in *Proc. 2nd ACM/IEEE Int. Symp. Netw.-on-Chip*, Apr. 7–10, 2008, pp. 139–148.

[18] L. Leung and C. Tsui, "Energy-aware synthesis of networks-on-chip implemented with voltage islands," in *Proc. DAC*, 2007, pp. 128–131.

[19] U. Y. Ogras, R. Marculescu, P. Choudhary, and D. Marculescu, "Voltage-frequency island partitioning for GALS-based networks-on-chip," in *Proc. DAC*, Jun. 2007, pp. 110–115.

[20] Y. Thonnart, E. Beigné, and P. Vivet, "Design and implementation of a GALS adapter for ANoC based architectures," in *Proc. ASYNC*, 2009, pp. 13–22.

[21] C. Addo-Quaye, "Thermal-aware mapping and placement for 3-D NoC designs," in *Proc. SOCC*, 2005, pp. 25–28.

[22] J. Kim, C. Nicopoulos, D. Park, R. Das, Y. Xie, N. Vijaykrishnan, M. S. Yousif, and C. R. Das, "A novel dimensionally-decomposed router for on-chip communication in 3D architectures," in *Proc. ISCA*, 2007, pp. 138–149.

[23] D. Park, S. Eachempati, R. Das, A. K. Mishra, Y. Xie, N. Vijaykrishnan, and C. R. Das, "MIRA: A multi-layered on-chip interconnect router architecture," in *Proc. ISCA*, 2008, pp. 251–261.

[24] V. F. Pavlidis and E. G. Friedman, "3-D topologies for networks-on-chip," in *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, Oct. 2007, vol. 15, no. 10, pp. 1081–1090.

[25] B. S. Feero and P. P. Pande, "Networks-on-chip in a three-dimensional environment: A performance evaluation," *IEEE Trans. Comput.*, vol. 58, no. 1, pp. 32–45, Jan. 2009.

[26] I. Loi, F. Angiolini, and L. Benini, "Supporting vertical links for 3D networks on chip: Toward an automated design and analysis flow," in *Proc. Nanonets*, 2007, pp. 23–27.

[27] S. Stergiou, F. Angiolini, S. Carta, L. Raffo, D. Bertozzi, and G. De Micheli, "×pipesLite: A synthesis oriented design library for networks on chips," in *Proc. DATE*, 2005, pp. 1188–1193.

[28] C. Seiculescu, S. Murali, L. Benini, and G. De Micheli, "NoC topology synthesis for supporting shutdown of voltage islands in SoCs," in *Proc. DAC*, 2009, pp. 822–825.

[29] I. Loi, F. Angiolini, and L. Benini, "Developing mesochronous synchronizers to enable 3D NoCs," in *Proc. DATE*, 2008, pp. 1414–1419.

[30] T. Cinotti, "Progettazione di Una Unita per la Comunicazione Asincrona per Link di Network on Chip," Tesi di Laurea, DEIS, Univ. Bologna, Bologna, Italy, 2007.