

Voltage-Frequency Island Partitioning for GALS-based Networks-on-Chip

Umit Y. Ogras, Radu Marculescu, Puru Choudhary, Diana Marculescu

Department of Electrical and Computer Engineering

Carnegie Mellon University, Pittsburgh, PA, USA

e-mail: {uogras,radum,puruc,dianam}@ece.cmu.edu

ABSTRACT

Due to high levels of integration and complexity, the design of multi-core SoCs has become increasingly challenging. In particular, energy consumption and distributing a single global clock signal throughout a chip have become major design bottlenecks. To deal with these issues, a globally asynchronous, locally synchronous (GALS) design is considered for achieving low power consumption and modular design. Such a design style fits nicely with the concept of voltage-frequency islands (VFIs) which has been recently introduced for achieving fine-grain system-level power management. This paper proposes a design methodology for partitioning an NoC architecture into multiple VFIs and assigning supply and threshold voltage levels to each VFI. Simulation results show about 40% savings for a real video application and demonstrate the effectiveness of our approach in reducing the overall system energy consumption. The results and functional correctness are validated using an FPGA prototype for an NoC with multiple VFIs.

Categories and Subject Descriptors

B.7 [Hardware]: Integrated circuits.

General Terms

Algorithms, Design.

Keywords

Voltage-frequency island, GALS, Multi-processor systems, networks-on-chip.

1. INTRODUCTION

Recognized by the International Roadmap for Semiconductors as the main bottlenecks in providing increased performance and platform capabilities, the on-chip communication and power management require a drastic departure from the classic design methodologies [1]. Networks-on-Chip (NoC) communication architectures have recently emerged as a promising solution for on-chip scalable communication beyond the capabilities of classical bus-based and Point-to-Point (P2P) architectures [7][13]. Besides its advantages in terms of modularity, design re-use, and performance, the NoC approach offers a matchless platform for implementing the GALS paradigm [4] and makes clock distribution and timing closure problems more manageable. Given that

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

DAC 2007, June 4–8, 2007, San Diego, California, USA.

Copyright 2007 ACM 978-1-59593-627-1/07/0006 ...\$5.00.

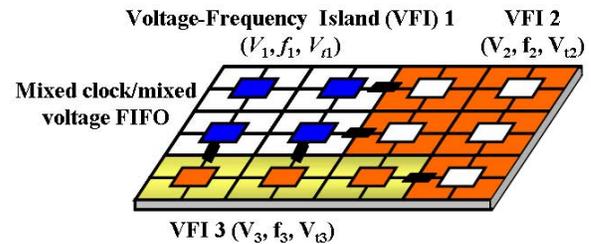


Figure 1 A sample 2D Mesh network with 3 VFIs. Communication across different islands is achieved through mixed clock/mixed voltage FIFOs.

for complex systems built at 65nm and below it is almost impossible to move signals across the die in a single clock cycle or in a power efficient manner, it becomes obvious that a shift towards global on-chip asynchronous communication is needed. In addition, a GALS-based design style fits nicely with the concept of VFIs, which has been recently introduced for achieving fine-grain system-level power management. The use of VFIs in the NoC context is likely to provide better power-performance trade-offs than its single voltage, single clock frequency counterpart, while taking advantage of the natural partitioning and mapping of applications onto the NoC platform. However, despite the huge potential for energy savings when using VFIs, the NoC design methodologies considered so far are limited to a single voltage-clock domain [2,10,15]. On the other hand, studies that do consider multiple VFIs assume that each module/core in the design belongs to a different island and different islands are connected by P2P links [8,17].

To address these challenges (and unlike existing work), this paper explores the design and optimization of novel NoC architectures partitioned into multiple VFIs which rely on a GALS communication paradigm. In such a system, each voltage island can work at its own speed, while the communication across different voltage islands is achieved through mixed clock/mixed voltage FIFOs (see Figure 1). This provides the flexibility to scale the frequency and voltage of various VFIs in order to minimize energy consumption. As a result, the advantages of both NoC and VFI design styles can be exploited simultaneously.

The design of NoCs with multiple VFIs involves a number of critical steps. First, the granularity (*i.e.*, the number of different VFIs) and chip partitioning into VFIs needs to be determined. While an NoC architecture where each processing/storage element (PE) constitutes a separate VFI exhibits the largest potential savings for energy consumption, this solution is very costly. Indeed, the associated design complexity increases due to the overhead in implementing the mixed-clock/mixed-voltage FIFOs

and voltage converters required for communication across different VFIs, as well the power distribution network needed to cover multiple VFIs. Additionally, the VFI partitioning needs to be performed together with assigning the supply and threshold voltages and the corresponding clock speeds to each VFI.

The novel contribution of this paper is a design methodology for *partitioning* a given NoC architecture into multiple voltage-frequency domains, and *assigning* the supply and threshold voltages (hence the corresponding clock frequencies) to each domain such that the total energy consumption is minimized under given performance constraints. We present a *unified* approach for solving the VFI partitioning and voltage/speed assignment problems. The basic idea is to start with an NoC configuration where each PE belongs to a separate VFI characterized by given supply and threshold voltages and local clock speed (*i.e.*, having initially N VFIs, for N PEs). This solution may achieve the minimum *application energy consumption*, but not necessarily the minimum *total energy consumption* due to the additional overhead incurred by implementing a large number of VFIs. Therefore, the proposed approach finds two candidate VFIs to merge such that the decrease in the energy consumption is the largest among all possible merges, while performance constraints are still met. The process is repeated until a single VFI implementation is obtained. Consequently, for all possible VFIs (*i.e.*, $1, 2, \dots, N$), we obtain the partitioning and corresponding voltage level assignments such that the total energy is minimized, subject to given performance constraints. Finally, among all VFI partitionings determined by the iterative process, the one providing the minimum energy consumption is selected as being the solution of the VFI partitioning problem.

The remainder of this paper is organized as follows. Section 2 reviews the related work and highlights our major contribution. Section 3 presents the problem formulation and solution to VFI partitioning and static voltage-frequency assignment. Experimental results are included in Section 4. Finally, Section 5 concludes the paper.

2. RELATED WORK AND CONTRIBUTIONS

GALS-based systems consist of several synchronous IPs that communicate with each other asynchronously [4]. There have been many efforts to design low latency asynchronous communication mechanisms between synchronous blocks. Some of them include design of mixed-clock FIFOs design [5], while others include design of asynchronous wrappers [16]. The design style based on multiple VFIs has been proposed in [12]. It fits very well with the GALS design style, where the synchronous IPs in the design have both different voltages and frequencies.

Despite the natural fit between VFI design style and NoCs, the existing design methodologies for NoCs are confined to single voltage/single frequency domain [10,11,15]. There have been several design efforts to combine the benefits of NoC interconnect mechanism with GALS-based design style. An FPGA prototype of a GALS-based NoC with two synchronous IPs is presented in [18]. In [6], a method to reduce the wire propagation delays in a GALS-based NoC is proposed. However, these studies assume that all the nodes in the network belong to different clock domains, which may be a costly proposition.

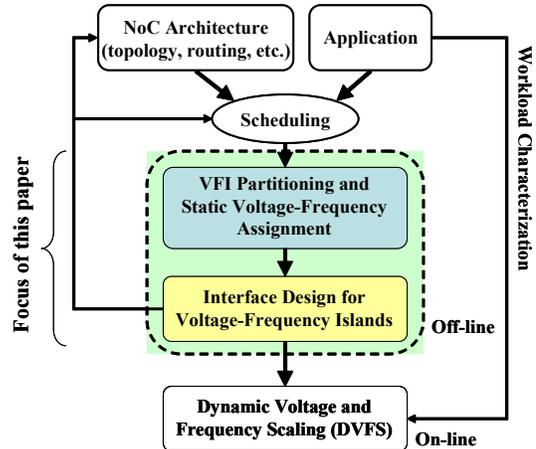


Figure 2 Overview of the proposed methodology.

In contrast to previous approaches, we address herein the VFI partitioning and voltage/speed assignment problem for NoCs with the objective of minimizing the overall energy consumption and present a novel algorithm for solving it. To this end, our contributions in this paper are as follows:

- A methodology for multi-clock/voltage domain NoC design
- An algorithm for VFI partitioning and supply/threshold voltage assignment
- Extensive evaluation of the proposed approach using realistic benchmarks and an FPGA prototype.

3. VFI PARTITIONING AND STATIC VOLTAGE ASSIGNMENT PROBLEMS

3.1. Basic Assumptions and Methodology Overview

We consider a tile-based implementation laid out as a grid, as shown in Figure 1. Each tile contains a processing or storage element (referred to as PE) and a router. We employ wormhole flow control and XY routing algorithm. Communication across different voltage-frequency islands is achieved through mixed-clock/mixed-voltage FIFOs.

The target application is first scheduled to the target NoC architecture which consist of several heterogeneous PEs. In this paper, we used Earliest Deadline First (EDF) and a heuristic called Energy Aware Scheduling (EAS) to generate both computation and communication task schedules [11]. However, in general, the proposed approach can be used with arbitrary schedules.

As shown in Figure 2, given the target architecture, the driver application and its schedule, the proposed methodology determines the VFI partitioning and the supply and threshold voltage assignment to the VFIs. The voltages are assigned such that the total energy spent in *both* computation and communication is minimized, subject to performance constraints. The performance constraints are imposed by the driver application as *deadlines* for certain tasks and/or minimum *throughput* requirements.

The voltage-frequency assignments computed using the proposed approach are static. For the applications with large workload variability, the operating voltage and frequency can be further controlled dynamically around these static values. In this paper, however, we restrict ourselves to the case of *static* voltage/speed assignment, as summarized in Figure 2.

3.2. Formulation of the Problems

3.2.1. Energy and delay models

The set of tiles in the network is denoted by $T = \{1, \dots, N\}$. The supply and threshold voltages for tile $i \in T$ are given by V_i and V_{ti} , respectively. Using this notation, the sum of dynamic and static energy consumptions associated with tile $i \in T$ can be written as:

$$E_i(V_i, V_{ti}) = R_i C_i V_i^2 + T_i k_i V_i e^{\left(-\frac{V_i}{S_i}\right)} \quad (1)$$

where R_i is the number of active cycles, C_i stands for the total switched capacitance per cycle, T_i is the number of idle cycles, k_i is a design parameter, and S_i is a technology parameter [3].

We assume that the static component of the communication energy component is included in the second term of Equation 1, since each link and router belongs to a tile in the network. The dynamic part of the communication energy consumption is found using the *bit energy metric* [20] defined as:

$$E_{bit} = E_{L_{bit}} + E_{B_{bit}} + E_{S_{bit}} \quad (2)$$

where $E_{L_{bit}}$, $E_{B_{bit}}$, and $E_{S_{bit}}$ represent the energy consumed by the link, buffer and switch fabric, respectively. Assuming the bit energy values are measured at V_{DD} , the energy needed to transmit one bit from tile $src \in T$ to tile $dst \in T$ is found as:

$$E_{bit}(src, dst) = \sum_{i \in P} (E_{L_{bit}}(i) + E_{B_{bit}}(i) + E_{S_{bit}}(i)) \frac{V_i^2}{V_{DD}^2} \quad (3)$$

where P is the set of tiles on the path from tile src to tile dst .

The clock period for each tile i is a function of its supply and threshold voltage, and it can be expressed as:

$$\tau_i(V_i, V_{ti}) = \frac{K_i V_i}{(V_i - V_{ti})^\alpha} \quad (4)$$

where $1.2 \leq \alpha \leq 1.5$ is a technology parameter and K_i is a design-specific constant [19]. Thus, the operating frequency of a tile and the VFI j it belongs to, is determined by the largest cycle time of the comprising tiles, *i.e.*

$$f_j \leq \min_{i \in S_j} \left\{ \frac{1}{\tau_i(V_i, V_{ti})} \right\} \quad (5)$$

where S_j is the set of tiles that belong to VFI j .

Finally, we assume each router is locally synchronous and communicates with its neighbors via mixed-clock/mixed-voltage FIFOs. As such, the communication latency between tile src and tile dst , while sending $vol(src, dst)$ bits of data, is given by:

$$t_{comm}(src, dst) = \sum_{i \in P} \frac{\mu_s}{f_i} + t_{fifo} \left[\frac{vol(src, dst)}{W} \right] \quad (6)$$

where W is the channel width and μ_s is the number of cycles it takes to traverse a single router and the outgoing link. t_{fifo} is the latency of the FIFO buffers and it is found experimentally using the prototype described in Section 4.3. The first term of Equation 6 gives the latency for the header flits while passing through the routers on path P , while the second term is the serialization latency.

3.2.2. The static voltage/speed assignment problem

Assume that the target application consists of a set of tasks G . For each task $t \in G$, the initial schedule specifies the deadline (d_t), start time (st_t), the number of cycles required to execute the task (x_t), as well as the tile where the task will run on.

When the network is partitioned into N VFIs, $i = 1, \dots, N$, (*i.e.*, each tile belongs to a different island), the static voltage assignment problem can be formulated as follows:

Given an NoC architecture with N tiles, where each tile is a separate VFI, a communication task graph describing the driver application, and its schedule,

Assign the supply (V_i) and threshold (V_{ti}) voltages, such that the total application energy consumption is minimized; that is:

$$\min E_{App} = \sum_{i \in T} E_i(V_i, V_{ti}) + \sum_{i \in T} \sum_{j \in T} vol(i, j) E_{bit}(i, j) \quad (7)$$

subject to deadline constraints expressed as:

$$\frac{x_t}{f_t} + t_{Comm}^t \leq d_t - st_t \quad \forall t \in G \quad (8)$$

where x_t/f_t is the computation time and t_{Comm}^t is the communication delay encountered when task t needs to communicate with a task mapped to a different tile. The number of cycles required to execute the task x_t is given by the schedule and t_{Comm}^t is computed using Equation 6. The left hand side of Equation 8 gives the sum of computation and communication times, while the right-hand side gives the amount of time the task should be completed without violating the schedule.

3.2.3. The voltage-frequency island partitioning problem

Even though decoupling the supply and threshold voltages of each tile results in a system with finest granularity for voltage/frequency control, the overhead of designing a large number of islands may undermine the potential for energy savings. In fact, it may be possible to merge several islands with a negligible increase in application energy consumption. In the extreme case, all tiles can be conceivably merged into a single VFI. Between the two extreme points, there exists a myriad of choices with varying energy/cost trade-offs. In order to compare these choices, we need to quantify the energy overhead of extra VFIs.

The energy overhead of adding one additional voltage-frequency island to an already existing design can be written as:

$$E_{VFI} = E_{ClkGen} + E_{Vconv} + E_{MixClkFifo} \quad (9)$$

where E_{ClkGen} is the energy overhead of generating additional clock signals and E_{Vconv} denotes the energy consumption of the voltage level converters. Finally, $E_{MixClkFifo}$ is the overhead due to the mixed-clock/mixed-voltage FIFOs used in interfaces.

Besides energy, additional VFIs exhibit area and implementation overheads, such as routing multiple power distribution networks. We assume that the maximum number of VFIs is given as a constraint. Therefore, the VFI partitioning problem is formulated as follows:

Given

- An NoC architecture;
- The scheduling of the driver application across the network;
- Maximum number of allowed VFIs;

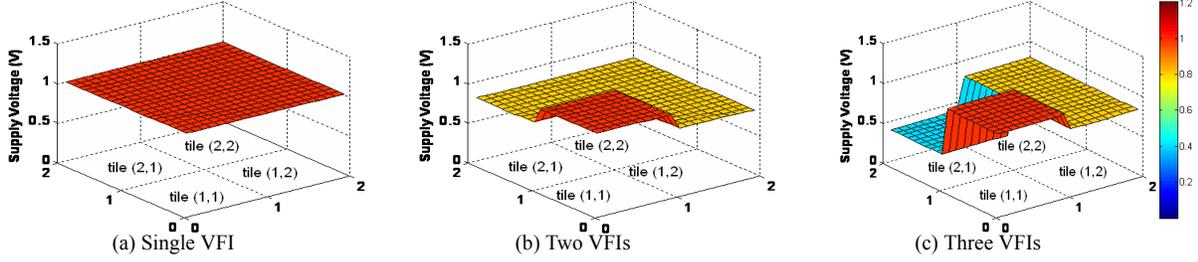


Figure 3 Different voltage-frequency island partitionings and corresponding static voltage assignments for a 2×2 network.

- Physical implementation constraints (e.g., certain tiles have to operate at a given voltage level, or a subset of tiles have to belong to the same VFI)

Find the optimum number of VFIs ($n \leq N$), VFI partitioning; and assign the supply and threshold voltages to each island, such that the *total energy consumption*, i.e.,

$$E_{Total} = E_{App} + \sum_{i=1}^n E_{VFI}(i) \quad (10)$$

is minimized, subject to performance constraints in Equation 8.

3.3. Motivational Example

As an example, we analyze the *office-automation* benchmark [9]. The application is scheduled on a 2×2 network using the EDF discipline, and the proposed approach is used for the VFI partitioning and static voltage assignment.

When the entire network consists of a single VFI, the supply and threshold voltages are found to be 1V and 0.15V, respectively (see Figure 3(a)). The corresponding schedule and these voltage assignments result in a $10.5mJ$ energy consumption. When analyzing this design, we observe that one of the tasks has a zero slack available, while others have a positive slack. When the network consists of two islands, the task with a zero slack is decoupled from the others. As shown in Figure 3(b), only one of the tiles needs to operate at 1V, while the supply voltage of the others is reduced to 0.8V. The energy consumption of this solution drops to $7.5mJ$, which represents about 29% savings.

The network can be further partitioned into three islands, as shown in Figure 3(c). For this example, a finer partitioning granularity results in diminishing rate of returns. In addition, the overhead of the extra island undermines the potential for energy savings. In this example, the energy consumption of the three- and four-island networks is $7.6mJ$ and $7.8mJ$, respectively. Hence, the network partitioning shown in Figure 3(b) results in a minimum energy consumption.

3.4. Solution Methodology

We solve the VFI partitioning and static voltage assignment problems in a unified manner; that is, the partitioning and voltage assignment are performed simultaneously.

We start off with a VFI partitioning where all the PEs belong to separate islands, as shown in the left most box in Figure 4. Then, we solve the static voltage assignment problem in Section 3.2.2. To solve Equation 7, we utilize a nonlinear inequality constrained problem solver based on gradient search available in Matlab and find the supply and threshold voltages that minimize the total energy consumption subject to performance constraints. Naturally, decoupling the supply and threshold voltage levels of

the tiles brings ample opportunities for energy optimization. Hence, the *application* energy consumption becomes minimum in this case. At the same time, a larger number of VFIs results in a more complex system and a larger energy overhead.

The number of VFIs, hence system complexity, can be decreased by merging some of the neighboring islands. Merging islands brings a number of additional constraints to the problem formulation. For instance, when tiles i and j are merged, constraints $V_i = V_j$ and $V_{ti} = V_{tj}$ need to be added and thus, the voltage assignment problem in Section 3.2.2 is solved with these additional constraints. Due to these additional constraints, the *application* energy consumption goes up after the two islands are merged. However, the *total* energy consumption given by Equation 10 may decrease, if the increase in the application energy consumption is smaller than the reduction in the energy overhead due to merging two VFIs.

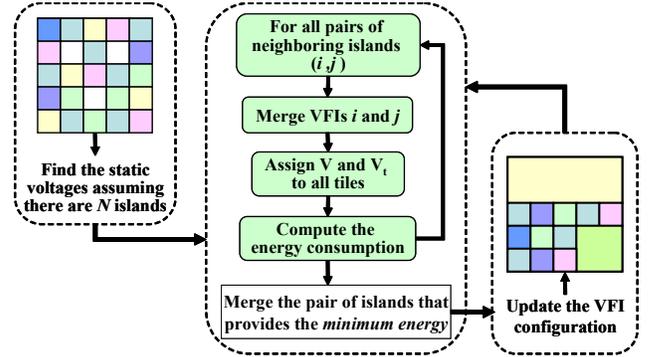


Figure 4 Outline of the proposed VFI partitioning and static voltage assignment methodology.

In the second step of the algorithm (i.e., the middle box in Figure 4), the decrease in the total energy consumption as a result of merging each pair of neighboring tiles is computed. Since there are $2n(n-1)$ pairs of neighbors in a $n \times n$ grid, $2n(n-1)$ evaluations are performed during this stage. Then, the pair of islands that results in the largest reduction of the total energy consumption is picked and merged permanently, as depicted in the right most box in Figure 4. After the VFI configuration is updated, the second step is executed again to find the next pair of candidate tiles to merge. This process continues until all tiles are merged and the network consists of a single island.

The proposed algorithm starts with N VFIs and reaches a single VFI at the end; as such, it evaluates *all* possible levels of VFI granularity. Hence, the proposed algorithm can determine the partitioning with the minimum overall energy consumption and the corresponding supply and threshold voltage assignments.

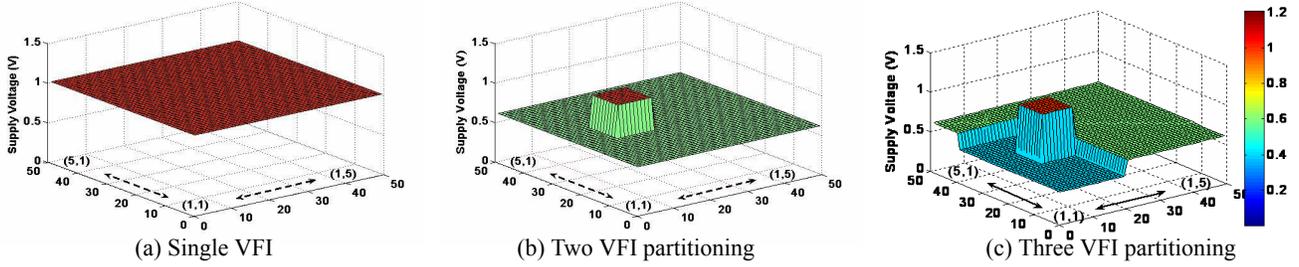


Figure 5 Different voltage-frequency island partitionings and corresponding static voltage assignments for benchmark 3 in Table 1.

4. EXPERIMENTAL RESULTS

This section illustrates the proposed VFI partitioning methodology and demonstrates its effectiveness in minimizing the energy consumption using real benchmarks. The first set of benchmarks is chosen from a public benchmark suite [9], while the second consists of a real video application. The energy related parameters (e.g. energy consumption of a task running on a certain PE) are derived from the benchmarks, while the technology parameters are taken from [14].

After the proposed algorithm is used to find the supply and threshold voltage for each VFI, we map them conservatively to the following discrete levels: $V_{supply} = \{0.4V, 0.6V, 0.8V, 1.0V, 1.2V\}$, $V_{Thold} = \{0.15V, 0.20V, 0.25V, 0.30, 0.35V\}$. Then, we compute the *total* energy consumption using Equation 10. The results reported hereafter are obtained for these discrete levels.

4.1. Experiments with Realistic Benchmarks

Consumer, *networking*, *auto-industry* and *telecom* benchmark applications are collected from E3S [9]. These benchmarks are scheduled onto 3×3 , 3×3 , 4×4 and 5×5 mesh networks, respectively using the EAS scheme presented in [11]. Then, the proposed approach is used for VFI partitioning and static voltage assignment. The second column of Table 1 (“1-VFI”) shows the minimum energy consumption when the entire network consists of a single island. The remaining columns show the energy consumption values obtained for the best partitioning with two and three islands, respectively. The energy consumption of the partitioning picked by the algorithm is marked with an asterisk.

Benchmark	Network Size	Total Energy Consumption (mJ)		
		1-VFI	2-VFI	3-VFI
<i>Consumer</i>	3×3	18.9	12.1*	12.2
<i>Network</i>	3×3	12.9	6.6*	6.7
<i>Auto-industry</i>	4×4	1.67	0.34*	0.40
<i>Telecom</i>	5×5	6.9	2.6	1.5*

Table 1: The reduction in the overall energy consumption obtained as a result of the proposed algorithm.

For *Consumer* benchmark, the minimum energy consumption is obtained when the network is partitioned into two voltage-frequency islands. With this configuration, the energy consumption drops from $18.9mJ$ to $12.1mJ$, which represents about 36% improvement. As shown in Table 1, partitioning the network to a finer granularity does not reduce the energy consumption further due to the overhead of having the extra islands, which is about $1.7mJ$. Similarly, a 2-VFI configuration achieves the minimum energy for *networking* benchmark. *Auto-industry* benchmark has

$1.67mJ$ energy consumption when there is no VFI partitioning. For this case, partitioning the network into two island drops to $0.34mJ$. Finally, generating more VFIs does not decrease the energy consumption further. For *telecom* benchmark, the energy consumption of the partitioning with the proposed algorithm is $1.5mJ$; this is more than $4 \times$ reduction compared to the 1-VFI case.

For better visualization, we show the supply voltage levels obtained for *telecom* benchmark in Figure 5. When there is a single voltage domain, all the tiles operate at $1V V_{DD}$ and $0.15V$ threshold voltage. However, when there are two VFIs, the supply voltage of all the tiles except tile (2,3) can be lowered to $0.6V$, while the threshold voltages remain at $0.15V$. When we increase the number of VFIs to three, the tiles voltage on the lower left corner is further reduced to $0.4V$.

The run-time of the algorithm ranges from a few tens of seconds to 30 minutes for the benchmarks reported in Table 1. In general, the proposed approach does not guarantee the optimal solution. However, we performed an exhaustive search for the example with 2×2 network in Section 3.3, and found that the energy consumption achieved with the proposed approach is within 1% of the optimum solution. More thorough analysis of the optimality is left as future work.

4.2. Experiments with a Real Video Application

We analyzed a video application consisting of MPEG2/MP3 encoder/decoder pairs [11]. The application is partitioned into a set of tasks and scheduled onto a 4×4 mesh network using the EDF scheme. Then, the proposed algorithm is used to find the VFI partitioning with minimum energy consumption.

The proposed approach starts with 16 separate VFIs. Then, it proceeds by merging the islands until a single island is obtained. As shown in Figure 6, the total energy consumption is improved until we reach two islands. For instance, when we move from 16 to 15 islands, the increase in the application energy consumption is negligible. So, the total energy consumption reduces due to the smaller overhead. Finally, a 2-VFI partitioning where the tasks of the same application reside in different islands achieves the minimum energy consumption. This resulting partitioning results in 40% improvement compared to the 1-VFI case.

4.3. Validation of VFI-based NoC via Prototyping

To further validate the simulation results for the VFI-based NoC, we have prototyped a generic system on Virtex2Pro Xilinx FPGA platform. A typical router in a NoC consists of a FIFO and an output controller (OC) for each port, and an arbiter to channel the traffic between the ports, as depicted in Figure 7. To connect a node in a VFI with another node residing in a different VFI, all data and control signals need to be converted from one

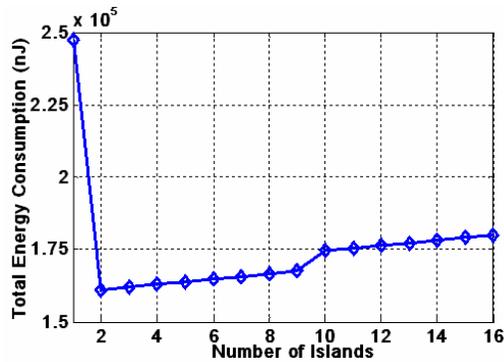


Figure 6 (a) The variation of total energy consumption as a function of the number of voltage-frequency islands.

frequency/voltage domain to another. For this purpose, we implemented mixed-clock/mixed-voltage interfaces using FIFOs, which are natural candidates for converting the signals from one VFI to the another, as shown in Figure 7.

To support the simulation results, we implement a GALS-based NoC with a 4×4 mesh topology using Verilog HDL. Block RAM-based mixed-clock FIFOs from the Xilinx library are used in routers to transfer data between different clock domains. Our design can be partitioned into as many as 16 VFIs. In our implementation, the signal conversion, both in terms of clock and voltage domains, occurs at FIFO interfaces. In this particular design, the Delay Locked Loops (DLLs) present in the Xilinx FPGA device are used to generate the individual clock signals. However, the voltage level conversion is not supported so multiple voltage levels are not readily available for the Xilinx platforms.

For experimental purposes, this implementation is configured with 16 islands and simulated using the *auto-industry* benchmark from E3S [9]. We first verify that no packets are lost in the VFI interfaces. After that, we compute the total energy consumption corresponding to single VFI and 2-VFI implementations, as in Section 4.1. To compute the energy consumption values, we utilize the energy characterization of the on-chip routers reported in [13]. The total energy consumption for single VFI operating at 1V is found as 109nJ. On the other hand, the total energy consumption of the 2-VFI partitioning found using the proposed approach is 21.2nJ. Hence, we observe about 81% reduction in the energy consumption. The energy consumption results obtained using the FPGA prototype are different than that measured by simulation in Table 1 due to the differences in the target platform and implementation details. Nevertheless, we note that according to the simulation results, the relative improvement in the energy consumption for the same benchmark is 80%, which is very close to the result obtained using the actual prototype.

5. CONCLUSIONS

In this paper, we addressed the design of GALS-based NoC communication architectures with multiple voltage-frequency domains. More precisely, we introduced a methodology for multi-clock, multi-frequency domain NoC design, and presented an algorithm for voltage-frequency island partitioning and supply/threshold voltage assignment. We showed that using VFIs in the NoC context provides better power-performance trade-offs than its single voltage, single clock frequency counterpart, while

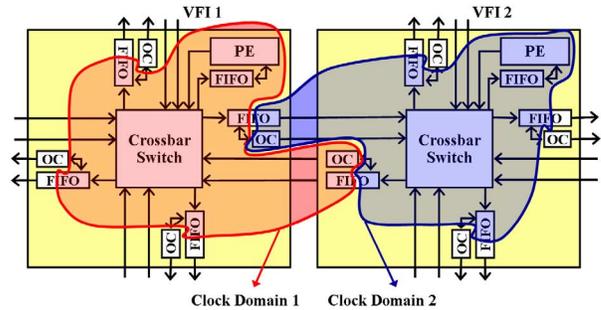


Figure 7 Illustration of the interface between two different voltage-frequency domains VFI1 and VFI2.

taking advantage of the natural partitioning and mapping of applications onto the NoC platform.

As future work, we plan to complete our current FPGA prototype and demonstrate the energy savings through real measurements on several applications.

Acknowledgements: This research is supported by Marco Gigascale Systems Research Center (GSRC) and in part by SRC Contract 2004-HJ-1189.

6. REFERENCES

- [1] International Technology Roadmap for Semiconductors Report, 2006.
- [2] D. Bertozzi, *et. al.*, “NoC synthesis flow for customized domain specific multiprocessor systems-on-chip,” *IEEE Trans. on Parallel and Distributed Systems*, 16(2), pp.113-129, Feb. 2005.
- [3] J. A. Butts and G. S. Sohi, “A static power model for architects,” *In Proc. of Intl. Symp. of Microarchitecture*, December 2000.
- [4] D. M. Chapiro, “Globally asynchronous locally synchronous systems,” PhD thesis, Stanford University, 1984.
- [5] T. Chelcea and S.M. Nowick, “A low latency fifo for mixed-clock systems,” *In Proc. of IEEE Computer Society Workshop on VLSI*, April 2000.
- [6] G. Campobello, *et. al.*, “GALS networks on chip: a new solution for asynchronous delay-insensitive links,” *In Proc. of DATE*, March 2006.
- [7] W. Dally and B. Towles, “Route packets, not wires: On-chip interconnection networks,” *In Proc. of DAC*, June 2001.
- [8] Y. S. Dhillon, A. U. Diril, A. Chatterjee and H. S. Lee, “Algorithm for achieving minimum energy consumption in CMOS circuits using multiple supply and threshold voltages at the module level,” *In Proc. of ICCAD*, Nov. 2003.
- [9] R. Dick, “Embedded system synthesis benchmarks suites (E3S),” <http://www.ece.northwestern.edu/~dickrp/e3s/>.
- [10] J. Dielissen, A. Radulescu, K. Goossens and E. Rijpkema, “Concepts and implementation of the Philips network-on-chip,” *IP-based SoC Design*, 2003.
- [11] J. Hu and R. Marculescu, “Communication and task scheduling of application-specific networks-on-chip,” *In IEE Proceedings Computers & Digital Techniques*, 152(5), pp. 643- 651, Sept. 2005.
- [12] J. D. E. Lackey, *et. al.*, “Managing power and performance for system-on-chip designs using voltage islands,” *In Proc. of ICCAD*, Nov. 2002.
- [13] H. G. Lee, N. Chang, U. Y. Ogras and R. Marculescu, “On-chip communication architecture exploration: A quantitative evaluation of point-to-point, bus and network-on-chip approaches”, to appear *ACM TODAES*, June 2007.
- [14] S. Martin, K. Flautner, T. Mudge and D. Blaauw, “Combined dynamic voltage scaling and adaptive body biasing for lower power microprocessors under dynamic workloads,” *In Proc. of ICCAD*, Nov. 2002.
- [15] M. Millberg, E. Nilsson, R. Thid and A. Jantsch, “Guaranteed bandwidth using looped containers in temporally disjoint networks within the Nostrum network on chip,” *In Proc. of DATE*, Feb. 2004.
- [16] J. Mutersbach, T. Villager and W. Fichtner, “Practical design of globally asynchronous locally synchronous systems,” *In Proc. of Intl. Symposium on Advanced Research in Asynchronous Circuits and Systems*, April 2000.
- [17] K. Niyogi and D. Marculescu, “Speed and voltage selection for GALS systems based on voltage/frequency islands,” *In Proc. of ASP-DAC*, Jan. 2005.
- [18] J. Quartana, *et. al.*, “GALS systems prototyping using multiclock FPGAs and asynchronous network-on-chips”, *In Proc. of Intl. Conf. on Field Programmable Logic and Applications*, Aug. 2005.
- [19] T. Sakurai and A.R. Newton, “Alpha-power law MOSFET model and its applications to CMOS inverter delay and other formulas,” *IEEE Journal of Solid-State Circuits*, 25(2), pp.584-594, April 1990.
- [20] T. T. Ye, L. Benini and G. De Micheli, “Analysis of power consumption on switch fabrics in network routers,” *In Proc. of DAC*, June 2003.