

Early Wire Characterization for Predictable Network-on-Chip Global Interconnects

I. Hatirnaz^{*}, S. Badel,
N. Pazos, and
Y. Leblebici
LSM, EPFL, Switzerland
ilhan.hatirnaz@epfl.ch,
stephane.badel@epfl.ch,
nuria.pazos@epfl.ch,
yusuf.leblebici@epfl.ch

S. Murali
CSL, Stanford University, USA
smurali@stanford.edu

D. Atienza, and
G. De-Micheli
LSI, EPFL, Switzerland
david.atienza@epfl.ch,
giovanni.demicheli@epfl.ch

ABSTRACT

This work envisions a common design methodology, applicable for every interconnect level and based on early wire characterization, to provide a faster convergence to a feasible and robust design. We claim that such a novel design methodology is vital for upcoming nanometer technologies, where increased variations in both device characteristics and interconnect parameters introduce tedious design closure problems. The proposed methodology has been successfully applied to the wire synthesis of a Network-on-Chip interconnect to: (i) achieve a given delay and noise goals, and (ii) attain a more power-efficient design with respect to existing techniques.

Categories and Subject Descriptors

B.8.2 [Hardware]: Performance and Reliability—*Performance Analysis and Design Aids*

General Terms

Design, Performance, Reliability

Keywords

Wire characterization, design methodology, global interconnects, NoCs

1. INTRODUCTION

Traditional design methodologies are far from providing predictable results for coming nanometer technologies. Actual nanometer related design problems are mainly due to the increased variations in both device characteristics and interconnect parameters. The serialization of tasks in existing design methodologies, which

^{*}now with Freescale Semiconductor, Munich

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, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

SLIP'07, March 17–18, 2007, Austin, Texas, USA.

Copyright 2007 ACM 978-1-59593-622-6/07/0003 ...\$5.00.

requires to start over again in case a design closure problem occurs at any further step along the design flow, makes it a tedious and time-consuming work handling the unpredictability problem in multi-billion transistor systems. This issue especially affects the system interconnect, which, in multi-core Systems-on-Chip (SoC), is responsible for managing the latency of the on-chip communication.

A predictable design can be tackled at two levels: (i) Architectural level, and (2) design level. In the first case, a completely regular structure is intended to overcome all possible manufacturing problems due to silicon submicron effects in every successive technology generation (e.g., a regular 2D-mesh topology for a global on-chip interconnect). On the other hand, following a predictable design flow involves considering the concept of predictability in every stage of the design flow, starting at system level down to physical design. This implies the usage of more predictable models during the simulations at different abstraction levels.

We propose a solution-set to break the serial flow that might cause many iterations back and forth between different stages of the design. Such solution-set is expected to benefit from the knowledge about the process heavily before going into the details of the physical implementation of the design. We envision a common design methodology applicable for every interconnect level and based on early wire characterization, as a successful solution for a faster convergence towards a feasible solution for every wire encountered in a SoC.

In this work, we have compared the results obtained by a conventional design flow with the ones obtained following the proposed approach based on early wire characterization to highlight the main drawbacks linked to the conventional flow (i.e., cumbersome iterations, increased design time, and unpredictability). Furthermore, we have applied the novel method to enhance a Network-on-Chip (NoC) synthesis algorithm, where the geometry of the physical channels between routing switches is chosen, such that the final design satisfies the design requirements in terms of length and delay, while targeting a power-efficient interconnect.

The rest of the paper is organized as follows: Section 2 introduces the main drawbacks encountered in traditional semi-custom design flows and their implications in the synthesis of NoC interconnects. Section 3 presents the novel approach based on early wire characterization to overcome the unpredictability issue of upcoming nanometer technologies. The application of the predictable global interconnects to NoC synthesis and the experimental results are then shown in Section 4. Finally, Section 6 concludes the paper.

2. BACKGROUND AND RELATED WORK

Similar to the methodology presented in [1], the proposed method approaches the interconnect predictability problem by shifting from the conventional *construct-by-correction* approach to *correct-by-construction* design. *Construct-by-correction* requires multiple passes from architecture to layout, where each iteration provides more insight about the final design and allows designers to correct mistakes in the next iterations.

On the other hand, *correct-by-construction* design refers to downstream enforcement of specifications used in early design through top-down constraints [2]. Its goal is to avoid unexpected issues during the design process, by increasing predictability. This is fundamental to speed-up the design of final chips, especially in today's designs with hundreds of millions of transistors. *Correct-by-construction* can be thought of a sequence of *guaranteed-correct* design transformations, in contrast to the more widely prevalent *construct-by-correction* design process consisting of large iterative loops. The *guaranteed-correctness* is achieved by restricting the set of available design transformations, which can be searched and characterized efficiently. This restriction creates a trade-off between optimality and predictability that is easily acceptable, considering the today's time-to-market issues and the complexity of design solution spaces. Another advantage of the design following the *correct-by-construction* approach is to alleviate the workload on the verification process, because it provides constant design convergence.

A traditional semi-custom design flow starts with the functionally-verified RTL description of the design, that is passed on to a logic synthesis tool with a set of constraints (timing, power, etc.). The synthesis tool outputs a netlist consisting of gates from the target standard-cell library. Thereafter, the physical design is generated through P&R. The final layout is extracted in order to account for the wiring effects. The extracted netlist is analyzed by different tools to check if the design requirements like timing, power consumption, signal integrity and such, are satisfied. In most cases, most of these design issues like signal integrity are addressed for the first time at this point of the design process, mainly because there is not an easy way to represent them earlier in the design flow. Any failure reported after these analysis might cause a number of design iterations become necessary, for which, the size of the loop depends on the severity of the issue being handled. As the unpredictability grows with the trend in changing interconnect geometries over different technologies, these problems can only be solved at a high cost of mainly the design time. All these facts lead to two main drawbacks of the traditional design flow: Insufficient incorporation of the physical knowledge during the different steps of the design flow and a lack of design flow control by the designers.

There is a number of solutions developed to overcome the issues listed above. One long-time-existing solution is the use of wireload models during logic synthesis where the physical layout data is not yet available. The synthesis tools use the wireload models to estimate the wire characteristics for a given net and a fanout. The wireload model specifies the capacitance, the resistance and the area of the wire, mainly based on statistical information [3]. With the emergence of $0.18\mu\text{m}$ technology, where a significant portion of the delay comes from the topology of the wire, floorplans and placement of cells drastically affect the path timing. At $0.18\mu\text{m}$, the traditional wireload model broke down, especially for the cases where the interconnect capacitance becomes dominant over the gate capacitances [4, 5].

Other solutions were offered like custom wire load models, enhanced floorplanning, and physical synthesis. All of these tech-

niques have found their applications in concrete designs, but no solution exists that adequately addresses all types of designs [6]. Therefore, in this paper we propose a new methodology, which relies on the fact that the detailed analysis tools employed at the end of Placement and Routing (P&R) are based on knowledge, that is already known before the design is physically structured, but not taken into account until P&R is finished. This a priori knowledge is particularly important in future technologies and paradigms to interconnect multiple cores in new MPSoC designs, such as NoCs, where multiple iterations are required due to the large number of wires present in a single die [8, 7].

The use of NoCs to achieve a predictable and modular global interconnect has been proposed in [7]-[9]. In this case, standard P&R schemes have been used in recent research works trying to address the synthesis of bus-based [10, 11, 12] and NoC-based interconnects [13]-[17]. Lately, additional knowledge about physical synthesis has been proposed to be added [14, 15] as we suggest, namely, floorplanning is used during the topology design process to get area and wire-length estimates. In [17], the problem of supporting multiple applications within the NoC synthesis process has been addressed. These works do not consider early wire characterization for the interconnects and are limited to initial area and power estimations of the cores included in the design. Thus, they do not target to find the most appropriate values for the different wire parameters as we address in this work.

In addition, methods to build area and power models for various NoC components have been presented in [19]-[22]. These works are complementary to our proposed approach. The area-power models of the NoC components (such as the switches) can be obtained from these models and our proposed early wire design approach can be used to design the NoC links.

3. EARLY WIRE CHARACTERIZATION FOR INTERCONNECTS

Conventional IC design flow (see Figure 1) focuses mainly on reducing the delay of the critical paths by optimizing the logic function, i.e., rearranging gates. Mainly, interconnect-related issues are not addressed until the detailed routing is finished. Only then, they are approached as an *after-thought*. At this point, different techniques are applied on the wires that do not meet the design constraints, such as optimal buffer insertion, transistor resizing, re-routing wires, replacing modules and even redesigning portions of the design. Although these techniques can help to reduce the interconnect delay [23], they do not help reducing the growing gap between the interconnect and device performance [24].

Knowing the constraints and the costs associated to the design decision regarding the interconnects at an early stage of the design process, one can develop more appropriate solutions, which will lead to a faster global convergence. Such design knowledge, extracted from early wire characterization, can be stored in a set of rules, which are benefited from during the decision-making on the wire parameters at each hierarchical level. Therefore, such a common design methodology is applicable for every interconnect level and provides a faster convergence to a feasible and robust design. In this work, we describe the application of such novel methodology applied to the global interconnects of a system design. It mainly comprises two stages: Characterization simulations and wire synthesis.

3.1 Characterization Simulations

The proposed methodology emerges from the idea of having the interconnects characterized as cells from a cell library and provides

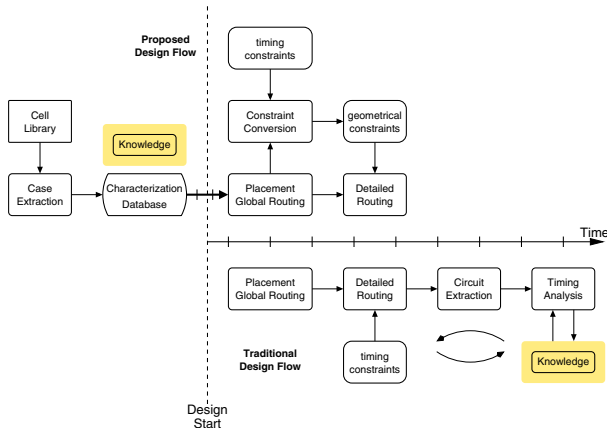


Figure 1: Comparison of the proposed design methodology with the traditional flow indicating the earlier usage of the design knowledge.

their different instances to be employed according to the prevailing requirements. Therefore, on top of the characterization of the cells, this methodology introduces the formalization of the interconnection resources for a given technology.

When a library is targeted for having the design mapped to, it is already known that for any possible path, the driving and the loading cells will be picked from the same set of a finite number of cells. What is not known until the routing is finished is the geometrical shape of the wiring that forms the path between any two gates. Hence, for each simulation, the *driver* is replaced with a cell picked from the target library; the *loading cell* is represented with a capacitance, for which, the possible values are extracted from the input capacitances of the cells; and the *interconnect* effects are accounted for with a RLCK wireload model. This process is illustrated in Figure 2.

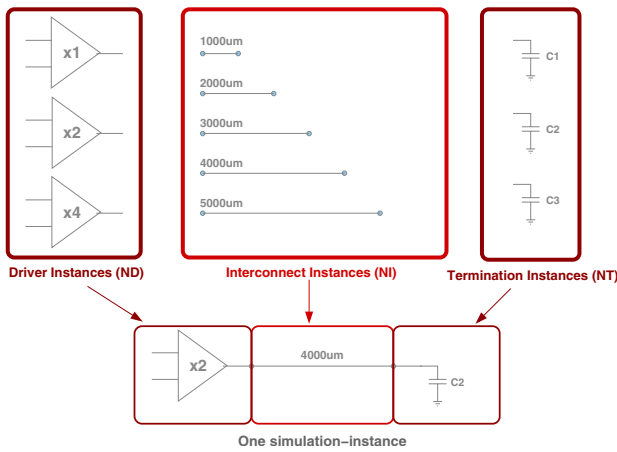


Figure 2: A number of simulations are run for all relevant combinations of driving gate, interconnect dimensions and the proximiety effects, and the load (the loading gate).

These simulation sets exhibit the following properties: They are design-independent; they are required to be run only once and then be utilized extensively as long as the same cell library is in use; they can be setup and run right after a technology and the target library are picked, without delaying the design start or causing longer de-

sign time; and their results are stored in a data structure (look-up table), which can be easily addressed during the design flow.

One important factor contributing to the unpredictability of the final designs is the signal integrity effects, specifically the coupling between any two nets due to the coupling capacitance and the mutual inductance. If these coupling effects are to be accounted for, then the wiring environment, in which the net-under-consideration is located, is also to be modeled. At the time these simulations are being run, this environment can not be defined because the neighboring wires are all unknown. One solution is to assume a worst case, namely, instantiating adjacent wires at each side of the considered net and making them act as aggressive as possible. In addition, design techniques, such as wire shielding, can be characterized as well, in order to observe the improvement.

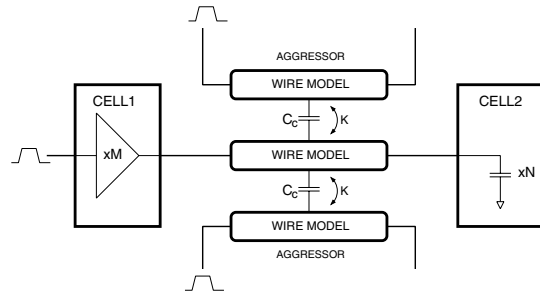


Figure 3: Case simulations setup. ('xM': cell drive strength, 'xN': cell input capacitance). 'Cc' and 'K' represent the capacitive and the inductive coupling between the aggressor nets and the victim net, respectively.

Figure 3 describes how different cases can be introduced to the simulation environment. The driver gate ('CELL1') is a cell with a given drive strength ('xM') from the target cell library. The load is a capacitor, of which, the value is equal to the input capacitance of the corresponding pin of the loading gate ('CELL2').

Figure 4 shows the characterization results for total wire delay (driver plus line delay), total energy, and noise for global interconnects (2, 4, and 6 mm), for a conventional 0.18μm technology, while increasing wire width (for minimum spacing) and wire spacing (for minimum width) and for a given driving buffer strength and a given receiver at the other end of the line.

3.2 Wire Synthesis

The predefined-characterized values extracted from the previous simulations are stored in a database as a set of rules. Such rules are used by the designer to choose the interconnect that meets the predefined requirements at early stages of the design flow.

An implemented cost-driven algorithm will recursively explore different interconnect alternatives by selecting and combining different design parameters, which were previously characterized. One possible solution set consists of the following items: Optimize wire length, use shielding, increase wire width, increase wire spacing, and limit the number of layers for routing. Note that the list is not in any order of preference and the solutions are not limited to these items. At this point, one important issue is to make sure that the routing tool is able to capture every kind of geometrical constraint listed above and attach them to the corresponding nets.

If and when a satisfying solution is found in the solution set, this solution dictates the geometrical dimensions of the wire and its location in reference to its neighbors. This information comes into the play as a geometrical constraint to be processed by the placement and the routing tools.

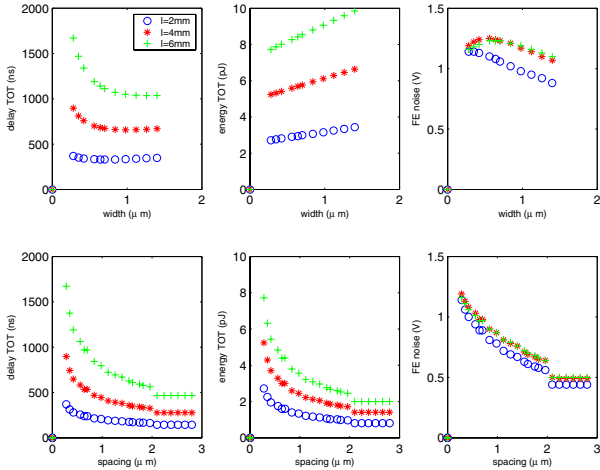


Figure 4: Variations of total wire delay, total energy, and far-end noise with respect to wire width (for minimum spacing) and spacing (for minimum width) for three different wire lengths (2, 4, and 6 mm.) in a conventional $0.18\mu\text{m}$ technology.

		Wire Characterization (ns)		P&R Tool (ns)	
		Driver	Line	Driver	Line
1 mm	WC	102 (27.6%)	31 (5.2%)	80	30
	BC	74 (11.7%)	7 (-16.1%)	95	9
2 mm	WC	109 (14.7%)	104 (13.04%)	95	92
	BC	77 (2.6%)	27 (-12.9%)	75	31
3 mm	WC	108 (0%)	363 (6.4%)	108	341
	BC	76 (-2.5%)	93 (-13.08%)	78	107
4 mm	WC	129 (16.2%)	744 (1.2%)	111	735
	BC	76 (-3.7%)	188 (-11.3%)	79	212

Table 1: Delay accuracy results. “WC” and “BC” stand for worst-case and best-case, respectively.

3.3 Accuracy Results

To measure the consistency of the values obtained via the characterization process, different lengths of 32-bit buses are routed between selected buffer cells using Cadence Encounter. After the detailed extraction of the final layout, the timing analysis is carried out on this netlist that includes the wiring effects. The comparison between the delay numbers from the characterization tables and the extracted design is provided in Table 1, in which, the first column lists the different wire lengths (in nm) and the second column indicates the condition the delay is calculated. “Worst-case” corresponds to a case, where a given wire has its neighboring tracks occupied with other wires for the longest path. This increases the total capacitance and slows the interconnect down. “Best-case” delay is the delay of the wire, which is routed away from the rest of the bus by the automatic routing tool. Taken the values reported by the commercial Placement and Routing tool as reference, we can observe a difference varying between 0% and 27.6% (average 9.8%). We also constate a higher variance of the inaccuracy for the driver delay than for the line delay. Such differences can be explained by two main factors: (1) The P&R tool does not account for inductance, while the proposed early wire characterization does; (2) during the detail extraction, the P&R tool may also consider the less-significant coupling effects of wires that are located further away, whereas, only the effect of the neighboring wires are taken into account during the characterization process.

4. PREDICTABLE SYNTHESIS OF NOCS

In this section, with the help of two case studies, we present the application of the proposed early wire characterization to design NoC-based global interconnects. The first case study is concentrated on achieving given delay and noise goals (Section 4.1), whereas the second one addresses the optimization of power for a given delay constraint (Section 4.2). Note that we have selected 2D-mesh topologies for the NoCs of the two case studies for their regularity. Nevertheless, the method can be applied to any regular interconnect topology.

4.1 Delay and Noise for NoC Global Wires

This case study intends to show the aforementioned issues related to the conventional design flows (cumbersome iterations, increased design time, unpredictability) and then compares its results with the ones obtained following the proposed approach on early wire characterization.

Problem statement: Assuming a 2D-mesh topology for a NoC-based interconnect (see Figure 5) where the distance, number of channels (2 per link), and number of bits per channel (32) between switches are predefined, find the fastest and the most power efficient solution for a given driver and a given loading cell by varying the width and spacing of the global wires.

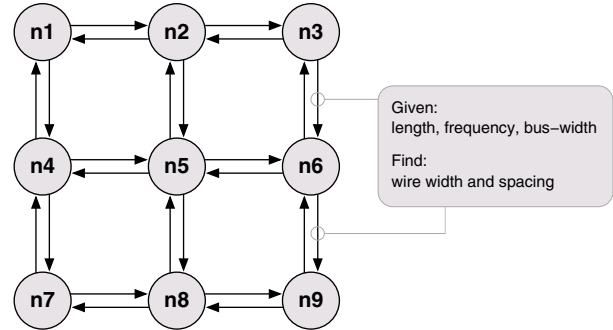


Figure 5: 2D-mesh Network-on-Chip Interconnect.

length [mm]	driver	rec.	width [μm]	spacing [μm]	delay [ns]	energy [pJ]	noise [mV]
2	16X	4X	0.35	2.10	137	0.80	0.14
		16X	0.35	2.10	144	0.85	0.14
		4X	0.28	2.10	281	0.65	0.32
		16X	0.28	2.10	299	0.70	0.28
4	16X	4X	0.56	2.10	248	1.71	0.16
		16X	0.56	2.10	255	1.76	0.16
		4X	0.28	2.10	533	1.25	0.30
		16X	0.28	2.10	555	1.30	0.27
6	16X	4X	0.64	2.10	388	2.63	0.17
		16X	0.64	2.10	397	2.68	0.17
		4X	0.28	2.10	832	1.85	0.26
		16X	0.28	2.10	858	1.90	0.24

Table 2: Fastest solutions for the 2D-mesh NoC interconnects together with the corresponding power and crosstalk numbers.

Table 2 and Table 3 show, for different wire lengths, drivers and loading cells, the fastest and most power-efficient solution, respectively. A search in the characterization database suffices to obtain, at an early design stage (after placing), the wire width and spacing values which will lead to the desired result in the final design layout. These values have been obtained without imposing any area

constraint, thus, the search routine always selects the biggest spacing between wires (as it can be observed in Figure 4 for a fixed wire width, delay and energy decrease while increasing the spacing). On the other hand, the search algorithm varies the wire width to obtain the fastest solution for each alternative, selecting always the minimum wire width ($0.28\mu\text{m}$) while looking for the most power-efficient interconnect.

length [mm]	driver	load	width [μm]	spacing [μm]	delay [ns]	energy [pJ]	noise [mV]
2	4X	4X	0.28	2.10	281	0.65	0.32
2	4X	16X	0.28	2.10	299	0.70	0.28
2	16X	4X	0.28	2.10	137	0.76	0.15
2	16X	16X	0.28	2.10	144	0.81	0.13
4	4X	4X	0.28	2.10	533	1.25	0.30
4	4X	16X	0.28	2.10	555	1.30	0.27
4	16X	4X	0.28	2.10	264	1.36	0.14
4	16X	16X	0.28	2.10	276	1.41	0.14
6	4X	4X	0.28	2.10	832	1.85	0.26
6	4X	16X	0.28	2.10	858	1.90	0.24
6	16X	4X	0.28	2.10	450	1.96	0.15
6	16X	16X	0.28	2.10	467	2.00	0.15

Table 3: Most power-efficient solution for the 2D-mesh NoC interconnects together with the corresponding delay and cross-talk numbers.

The values obtained at an early stage by applying the method based on early wire characterization have been checked against the output values provided by a commercial P&R tool. To this end, a 2D 3x3 mesh NoC is laid out using Cadence SOC Encounter, where the switching routers are internally modeled with only the receiver and transmitter parts, and their placements are arranged so that the physical links between them take different wire lengths (a screenshot of the layout for the complete 2D-mesh and for one of the switching routers can be seen in Figure 6). Each link consists of 32 bits in one direction, which sums up to 768 nets in total.

A timing-driven detailed routing is performed on the design, for which, the timing constraint is taken as the fastest solution provided in Table 2. The horizontal links are routed on third-level metal (METAL3), whereas, the vertical lines are routed on METAL4. These preferred routing directions are dictated inside the technology. A detailed extraction is carried out on the layout, followed by the timing analysis including the interconnect effects. The timing analysis lists a timing violation in all the global nets (see Table 4 for a wire length of 2mm, a driver of 16X and a loading cell (receiver) of 4X). This is due to the fact that the P&R tool is not able to adjust the interconnect geometry of the global nets in order to meet the imposed timing constraints.

path	required time [ns]	arrival time [ns]	slack [ns]	violation
1	0.135	0.230	-0.095	VIOLATED
2	0.135	0.228	-0.093	VIOLATED
3	0.135	0.228	-0.093	VIOLATED
4	0.135	0.228	-0.093	VIOLATED
5	0.135	0.227	-0.092	VIOLATED
6	0.135	0.227	-0.092	VIOLATED
...

Table 4: Timing analysis report for a wire length of 2 mm, a driver of 16X-buffer (BUF16) and a loading cell of 4X-buffer (BUF4).

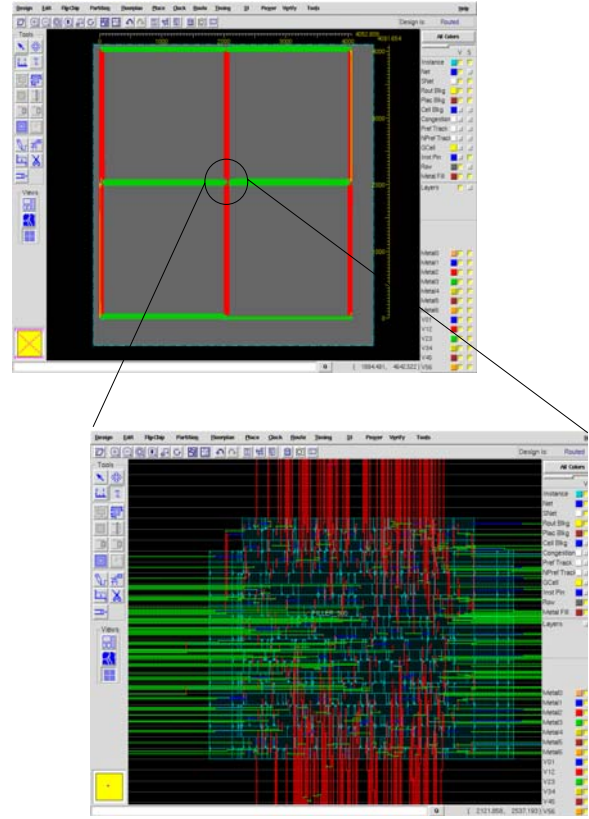


Figure 6: The layout of the switches in an NoC structure implemented in Cadence SOC Encounter.

The same design is then analyzed for crosstalk issues. The tool is asked to report any victim net, on which the total amount of coupling from the neighboring wires (aggressors) exceeds 40% of V_{DD} , that is 720 mV. The distribution of the number of nets over the amount of coupled noise is provided in Figure 7. Out of the 768 nets, 721 nets (94%) are reported to be outside the allowed noise limit (noise > 720 mV). It should be noted that the coupled noise becomes over 1.5 V (83% of V_{DD}) in some cases. This result shows that additional number of iterations are already required to fix the crosstalk issues, which might affect the design performance in other aspects, for example, increasing the power consumption or decreasing the operating speed and most importantly, augmenting significantly the design time.

Comparing the results obtained by applying both methods, we can conclude that the proposed approach based on early wire characterization provides a faster convergence to a feasible solution according to the imposed requirements. On the other hand, the classical design flow by means of a commercial P&R tool requires several iterations to meet the timing constraints and fix the crosstalk issues.

4.2 Optimizing Power in a NoC Design

In this application, we assume that the NoC topology, the NoC data width and the traffic rates across each of the NoC links are given as user inputs. Then, to satisfy the traffic rates, the different links in the NoC can be sized with different number of physical channels. That is, each link is segmented into different physical channels that can be utilized by different traffic flows in parallel. As an example, a 2×3 mesh topology is presented in Figure 8.

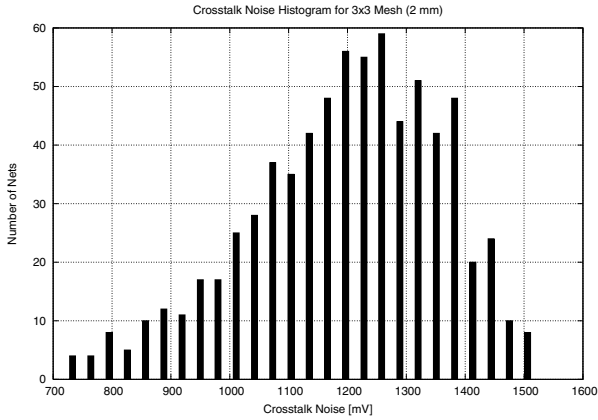


Figure 7: The number of victim nets for different ranges of coupling noise on them.

Each vertex in the figure represents a switch (and the core that is connected to the switch) and a link between two vertices has one or more physical channels. For example, the link from vertex v_1 to vertex v_3 has two physical channels, while the link from vertex v_0 to vertex v_1 has one physical channel.

The objective of the synthesis algorithm is to find the number of physical channels required for each link and to automatically tune the NoC operating frequency of operation, so that the most power efficient NoC configuration is obtained. Note that it is non-trivial to find the most power-optimal NoC operating frequency. When the NoC operating frequency is small, a large number of physical channels would be needed to satisfy the traffic rates. This results in larger switches and more wires, which can lead to large power consumption. On the other hand, a higher frequency of operation, though results in smaller switches and fewer wires, can also lead to higher power consumption. This is because, at higher operating frequencies, the switch and link hardware complexity is higher (as more logic is needed to achieve faster clock speeds during physical design) and the clock-net power consumption is also higher. Thus, exploring this trade-off with a synthesis engine is beneficial. To this end, we have defined an algorithm (presented in Algorithm 1) for NoC link synthesis that utilizes the proposed method based on early wire characterization for estimating the interconnect delay and power consumption.

In the first step of the algorithm, the NoC topology (T), the data

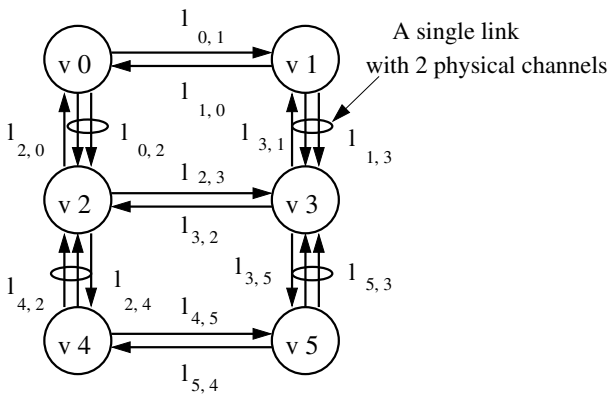


Figure 8: Example 2×3 mesh topology

width (dw), the link lengths (LL) and the traffic rates across the links (TR) are obtained as inputs from the user. In step 2, the operating frequency of the NoC ($freq$) is varied in user defined steps.

In the next step of the algorithm (step 3), the number of physical channels that are required for each link are computed. For the chosen frequency point, $freq$, the number of physical channels required at a link l_i , with traffic rate of tr_i is given by:

$$nc_i = tr_i / (freq \times dw) \quad (1)$$

As an example, if the traffic rate of a link is 800 MB/s, NoC data width is 4 Bytes and NoC frequency point is 100 MHz, then 2 physical channels are needed for that link.

Algorithm 1 Synthesis Algorithm

- 1: Obtain the NoC data width (dw), topology (T), link length (LL) and traffic rates (TR) as user inputs
- 2: **for** Each NoC frequency ($freq$) design point in user defined range **do**
- 3: Compute number of channels (nc_i) needed for each link i as $nc_i = tr_i / (freq \times dw)$
- 4: Compute the switch sizes from the number of physical channels instantiated.
- 5: Evaluate whether the switch size implementations can match the target $freq$
- 6: **for** each link i **do**
- 7: Use the proposed early wire characterization to find the best wire spacing (ws_i) and wire width (ww_i) for each wire of the link i , respecting the $delay_constraint(freq, ll_i)$ and wiring area constraint ($availablearea_i \geq (ww_i + ws_i) \times dw \times nc_i$).
- 8: If a valid wire parameter set (width and spacing) satisfying the delay and area constraints are met, set $all_constraints_met_i$ to true
- 9: **end for**
- 10: If ($all_constraints_met_i, \forall i$), obtain the power consumption for the synthesized NoC.
- 11: **end for**
- 12: From the set of synthesized NoCs, choose the design with least power consumption

In step 4, the sizes of the different switches are obtained, which are based on the number of physical channels instantiated for each link. Next, in step 5, we evaluate whether all the switches can meet the particular NoC operating frequency. This information is obtained from the P&R of switches of different sizes, which is computed off-line and fed as an input to the synthesis engine.

Then, if all the switches meet the design constraints, in steps 6-9, we start iterating through all the interconnects defined in the NoC using our proposed early wire characterization method (see Section 3). Using the values extracted from the early wire characterization, we estimate the best (most power efficient) wire spacing and width for the wires of each link, such that the delay constraint (based on the frequency point chosen in step 2) and the wiring area available for each channel are respected. For the delay constraints, we check whether the length of each link can be traversed in a single clock cycle at the chosen frequency point. We assume that the wiring area of the links are obtained as inputs. For regular topologies such as the mesh, the layout of the NoC is predictable and well structured. Thus, the available wiring area for the links can be easily obtained. Next, if all the design constraints are met, the power consumption of the NoC topology is stored (step 10) for that frequency design point. Finally, the most power efficient design among all the frequency points is chosen.

Experimental Results for NoC Designs

We have applied our early wire characterization based NoC synthesis flow for the design of several real-life mesh-based NoC systems.

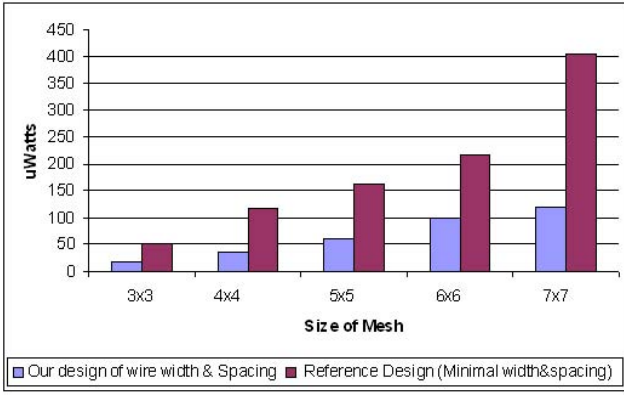


Figure 9: Power consumption for different sizes of mesh-based NoC topologies

All of them use the \times pipes architecture [18] NoC technology. We have performed a full wiring characterization to cover a large range of possible driving buffer strengths (values in the range between 4X and 16X), wire lengths (between 2mm and 6mm), wire widths (from 0.28 μ m to 1.40 μ m), spacing between neighboring wires (from 0.28 μ m to 2.80 μ m) and have accounted for crosstalk on the interconnects. Also, we have included accurate analytical models for the power consumption of the switches and wires in the used \times pipes [18] NoC. Then, to get the final power estimates to validate our initial design exploration, we perform the P&R phase of the components using Cadence SoC Encounter [25] and accurate capacitances and resistances are obtained, as back-annotated information from layout, with 0.13 μ technology library. The switching activities of network components are varied by injecting functional traffic. Capacitance, resistance, and switching activity results are combined to estimate power consumption of the whole NoC design (including the used clock-tree) using Synopsys PrimePower [26].

In the first set of experiments, we have the NoC topology to be a mesh, the core operating frequency to be 100 MHz, link length to be 2mm, and the data width of the NoC channels to 32-bits. Next, we obtain the wiring area available for each link to be the width of a switch. This is because, in a regular mesh topology, the links that

are between adjacent switches have the width of the switch to route wires. We also consider the realistic case, where the wires of a single channel are routed in a single metal layer. We vary the size of the mesh topology and synthesize the power efficient configuration for each mesh size, using the proposed early wire characterization based synthesis method. For comparison purposes, we also synthesize the designs with minimum wire width and spacing, as is done in the current state-of-the-art synthesis methods [14].

The power consumption results obtained for the different implementations of the links of the NoC are shown in Figure 9. The use of the early wire characterization proposed in the paper leads to large link power reductions (between 53% and 72%) when compared to the current state-of-the-art approaches. The link power consumption contributes up to 30% of the total NoC power consumption in all the studied implementations.

In the second set of experiments, we have fixed an intermediate size topology (i.e. 5x5 Mesh) and studied the possible benefits of the early wire characterization. The results obtained are shown in Figure 10. They clearly illustrate the benefits of our approach with respect to the reference design of minimal width and spacing. The power savings of the proposed approach increases significantly when the link length grows, already reaching up to a 60% power consumption reduction in the case of a 6mm interconnect length.

In the final set of experiments, we have evaluated the effect of the core speeds in the final design. In this case, the power consumption figures for different core speeds using our design and the reference design of minimal width and spacing (link length is assumed to be 2mm) are depicted in Figure 11. This figure shows that our proposed flow has clear benefits. In fact, similar to the previous experiments, the more demanding the requirements of the final design are (i.e. cores running faster), the more benefits our methodology achieves with respect to the reference case study. Thus, the proposed approach for early wire characterization achieves more power-efficient NoC designs with respect to more standard techniques for wiring modeling.

Finally and most importantly, as the accurate wiring delay values are considered during the synthesis phase itself, the design gap between the architectural level model and the physical layout implementation is bridged. This leads to a predictable interconnect architecture, which is highly desirable to achieve design closure and faster time-to-market of NoCs.

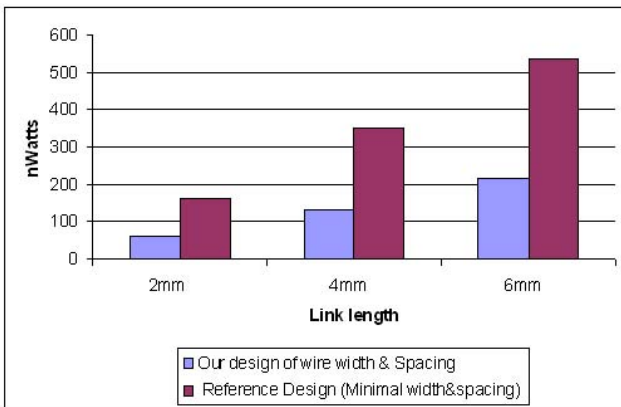


Figure 10: Effects on power in a 5x5 mesh NoC design due to variations of average link length

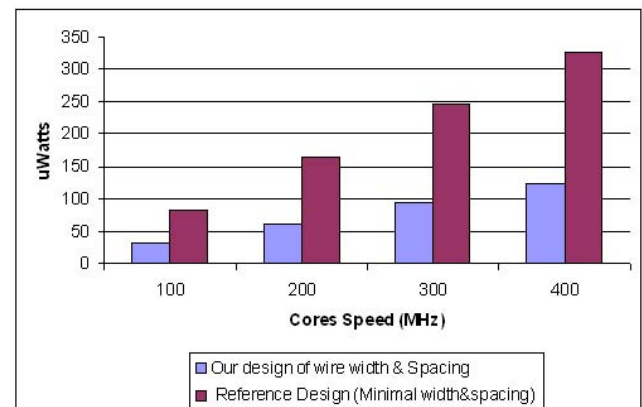


Figure 11: Effect on power due to variations of core speeds in NoC links of 2mm

5. ACKNOWLEDGMENTS

This work is partially supported by the US National Science Foundation (NSF, contract CCR-0305718) for Stanford University. It is also partially supported by the Swiss National Science Foundation (FNS, Grant 20021-109450/1) and the Spanish Government Research Grant TIN2005-5619.

6. CONCLUSIONS

In this paper we have presented a novel interconnect design methodology, which introduces the physical features of the wires at an early stage of the design process. This approach enables a faster convergence for more optimal solutions compared to the ones provided by standard techniques. Furthermore, it includes in a single model timing, power, crosstalk and inductive features of the wires, which, otherwise, are handled sequentially by the existing approaches in a more time-consuming iterative design process. This predictable method has been successfully applied to the synthesis of global NoC interconnects, focusing in the geometrical characteristics of the wires (e.g., wire width and spacing). In addition to achieving more power-efficient NoC designs, it leads to a predictable global interconnect, eliminating many iterations back and forth between different stages of the design. It has been demonstrated that leaving a commercial P&R tool perform the place-and-route unguided might produce worse results in terms of delay, crosstalk and power consumption compared to the proposed approach, where, the geometrical constraints, that are extracted from the characterization process, are preferred over other types of constraints (delay, noise, power).

7. REFERENCES

- [1] L. Carloni and A. Sangiovanni-Vincentelli, "Coping With Latency in SOC Design," in *IEEE Micro*, 2002.
- [2] P. Saxena, N. Menezes, P. Cocchini, and D. Kirkpatrick, "The Scaling Challenge: Can Correct-by-Construction Design Help?" in *Proc. of ISPD*, 2003.
- [3] S. Golson, "Resistance is futile! building better wireload models," in *Proc. of SNUG*, 1999.
- [4] S. Hojat and P. Villarrubia, "An integrated placement and synthesis approach for timing closure of powerpctm microprocessors," in *Proc. of ICCD*, 1997.
- [5] P. Gopalakrishnan, et al., "An analysis of the wire-load model uncertainty problem," in *IEE Trans on CAD*, 2002.
- [6] S. Bali, "Does Single-Pass Physical Synthesis Work for FPGAs?" in *Journal of FPGA and Structured ASIC*, 2004.
- [7] L. Benini and G. De Micheli, "Networks on Chips: A New SoC Paradigm", *IEEE Computer*, 2002.
- [8] W. Dally, B. Towles, "Route Packets, not Wires: On-Chip Interconnection Networks", in *Proc. of DAC*, pp. 684-689, June 2001.
- [9] M. Sgroi, M. Sheets, A. Mihal, K. Keutzer, S. Malik, J. Rabaey, A. Sangiovanni-Vincentelli, "Addressing the System-on-a-Chip Interconnect Woes Through Communication-Based Design", in *Proc. of DAC*, pp. 667-672, June 2001.
- [10] M. Gasteier, M. Glesner, "Bus-based communication synthesis on system level", in *ACM Trans on Design Automation of Embedded Systems (TODAES)*, 1999.
- [11] K. Lahiri et al., "Design Space Exploration for Optimizing On-Chip Communication Architectures", *IEEE Trans on CAD*, 2004.
- [12] S. Pasricha et al., "Floorplan-aware automated synthesis of bus-based communication architectures", in *Proc. of DAC*, 2005.
- [13] J. Hu, R. Marculescu, "Exploiting the Routing Flexibility for Energy/Performance Aware Mapping of Regular NoC Architectures", in *Proc. of DATE*, 2003.
- [14] S. Murali et al., "Mapping and Physical Planning of Networks on Chip Architectures with Quality-of-Service Guarantees", in *Proc. of ASPDAC* 2005.
- [15] K. Srinivasan et al., "An Automated Technique for Topology and Route Generation of Application Specific On-Chip Interconnection Networks", in *Proc. of ICCAD*, 2005.
- [16] A. Hansson et al., "A unified approach to constrained mapping and routing on network-on-chip architectures", in *Proc. of ISSS*, 2005.
- [17] S. Murali et al., "A Methodology for Mapping Multiple Use-Cases onto Networks on Chips", *Proc. of ASP-DAC*, 2006.
- [18] F. Angiolini et al., "Contrasting a NoC and a Traditional Interconnect Fabric with Layout Awareness", pp. 124-129, *Proc. of DATE*, 2006.
- [19] T. T. Ye et al., "Analysis of power consumption on switch fabrics in network routers", in *Proc. of DAC*, 2003.
- [20] H-S Wang et al., "Orion: A Power-Performance Simulator for Interconnection Network", in *Proc. of MICRO*, 2002.
- [21] N. Banerjee et al., "A power and performance model for network-on-chip architectures", in *Proc. of DATE*, 2004.
- [22] G. Palemore and C. Silvano, "PIRATE: A Framework for Power/Performance Exploration of Network-On-Chip Architectures", in *Proc. of PATMOS*, 2004
- [23] J. Cong, et al., "Interconnect Design for Deep Submicron ICs," in *Proc. of ICCAD*, 1997.
- [24] J. Cong, "Challenges and Opportunities for Design Innovations in Nanometer Technologies," SRC Design Sciences Concept Paper, Tech. Rep., December 1997.
- [25] Cadence, Cadence SoC Encounter, www.cadence.com.
- [26] Synopsys, Synopsys PrimePower, www.synopsys.com.
- [27] B. Hendrickson, R. Leland, "The Chaco User's Guide: Version 2.0", Sandia Tech Report SAND94-2692, 1994. URL: [//www.cs.sandia.gov/~bahendr/chaco.html](http://www.cs.sandia.gov/~bahendr/chaco.html)