Inter-Module Interconnect Strategy for System on Chip Applications

Abdallah Tabbara, Bassam Tabbara

Department of Electrical Engineering and Computer Sciences,

University of California, Berkeley, CA 94720

{atabbara,tbassam}@ic.eecs.berkeley.edu

Abstract

Interconnect effects that negatively impact performance, and compromise signal integrity are becoming more pronounced as technology moves deeper into sub-micron feature sizes, and designs are operated at higher frequencies. Many interconnect configurations and techniques have been proposed for a variety of domains (FPGA's, ASIC's, and networks…). In this work, we begin by investigating these techniques as they apply to the System on Chip (SoC) application domain. In particular, we restrict our analysis to an envisioned future embedded system architecture that consists of an interconnection of hundreds to thousands of 50K-100K gate IP (Intellectual Property) blocks. We assume that today's design, placement, routing, and interconnect optimization methods can handle intra-block problems, and the main issue to address is inter-module interconnection. Our work culminates in the proposal and evaluation, with respect to area/performance/power, of a new interconnect strategy engineered specifically for the IP "plug-and-play" SoC domain.

1 Introduction

Interconnect on a chip is used for signal communication, as well as power, ground and clock distribution. Interconnect effects that impact performance, compromise signal integrity, and increase power dissipation are becoming more pronounced as technology moves deeper into sub-micron feature sizes, and designs are operated at higher frequencies [17][16]. In this work, we investigate inter-module signal communication techniques for high performance in the context of the System on Chip (SoC) application domain. In particular, we restrict our analysis to an envisioned future embedded system architecture [14] that consists of an interconnection of hundreds to thousands of 50K-100K gate blocks [12]. We assume that today's design, placement, routing, and interconnect optimization methods can handle intra-block problems, and the main issue to address is inter-module interconnection [5].

In the subsections below we define the problem, provide the context, and outline the requirements of the solution of our research into the suitable IP interconnect strategy for the SoC domain.

1.1 Signal Delay and Degradation

According to the NTRS, by 2006 designs will contain over 100M transistors in less than 0.1μm technology with an increase in the number of I/O pins by 1.5X over the present. More devices on the chip and closer packed interconnects with thinner aspect ratios will mean more potential for simultaneous current surges, more coupling events that have the potential of causing false switching or delayed signals, and increased difficulty in determining circuit timing in the design flow [17].

The main consequences of the scaling down of interconnect wires are [15]:

1) wire resistance is maintained constant since both width and length are scaled down
2) wire capacitance per unit length is increasing; total capacitance is roughly constant, but with a different distribution from other contributors; intra-metal contributions becoming bigger than inter-metal ones.
3) contact and via resistances are scaling up, and have increased 6 fold over 4 generations, because their aspect ratio is becoming worse (fixed height, and smaller diameter).

Signal delay and degradation in the deep sub-micron world can have a crippling effect on the performance of designs. Signal preservation cannot be solved as an after thought in the design process. It must be dealt with at an early stage in the flow in order to minimize delay, and accommodate parasitic effects. Increased clock frequencies, larger chip dimensions, and smaller feature sizes (i.e. more device capacity) are making for variable inter-module interconnect lengths, as well as making interconnect delay dominant over that of devices. While system timing constraints are being met by increased clock speeds, functional timing constraints (i.e. relative timing requirements between module inputs) are becoming harder to satisfy because of this interconnect delay, and the variable wire lengths.

1.2 Motivation: The IP-Based SoC Application Domain

Our target application is the chip level assembly of pre-designed IP blocks, each under 100k gates in size, either as hard, firm, or soft macros [12]. Designing such a system involves the placement and wireplanning for performance [13] of 200-2000 modules whose average size is 50k gates with a dynamic range of module sizes of 1-500k gates [5]. Modules can be of different types: hard (layout), firm (gate level), or soft (RTL). Such a network has a large number of nets: 40k-100k with 10-100 pins per module.

This application domain puts limitations on which techniques we can use to address signal performance and integrity issues. Components in the SoC domain are black boxes where the designer may not be able to massage the function in order to remedy interconnect delay or parasitic

problems (see for example the work on "Digital Sensitivity" by Kirkpatrick and Sangiovanni-Vincentelli [10] to reduce cross talk). It is the authors' opinion that in a heterogeneous component-based SoC design feasible solutions to the delay and parasitic problems lie in refining the signaling [4] between components to assure that performance is maximized and parasitic effects are accommodated.

Another constraint that should be kept in mind is the low power and area requirements. Thus solutions geared towards predictability, and which entail a grid-like structure (or "fabric") consisting of laying power or ground wires in between any two adjacent wires [8], may not be suitable. In this domain the success for such a structure is very unlikely given the different shapes and sizes of interconnected components.

1.3 Interconnect Strategy for Enabling "IP-Based Design"

Industry experts agree that IP integration is the ideal technology for rapid SoC design development in a cost-efficient and fast time-to-market manner. This technology has not been widely adopted yet (full-custom designs are still favored) mostly because many issues still need to be resolved in terms of interfacing these components together. One idea [6], now finding wide acceptance in the community, is to register-bound IP's, thus temporally decoupling the inside of the block from the outside. This will allow IP's to be treated as "black-boxes" that are immune to glitches at the input, and do not generate any at their outputs, as well as to support "plug-and-play" where system developers can substitute one black box IP by another, given that they have the same functionality.

The importance of interconnect technology cannot be exaggerated for system LSI's, since IP mix-and-match brings totally new and important aspects beyond the normally discussed views and visions. Interconnect technology should therefore be re-visited in IP-based system LSI development of an electronic system [11].

In an IP-based SoC environment the interconnect strategy should be able to support:

a) connection of heterogeneous components with a relatively large number of pins in a "point-point" fashion,
b) "plug-and-play" and the preservation of the synchronous design assumptions

2 Related Work

Many interconnect configurations and techniques have been proposed for a variety of domains (FPGA's, ASIC's) that address signal delay and integrity preservation. These approaches can be classified as follows:

1) interconnect configurations and schemes:
   a) hierarchical interconnect structures and busses: such as those described in [21], provide for abstraction and hierarchy for large nets by decomposing them into smaller pieces, however such an approach is suitable for homogeneous components and quite inadequate for an interconnection of heterogeneous ones.
   b) synchronization schemes: such as the one described in [9], while quite promising as new approaches, involve changing the "synchronous assumptions", and such assumptions must be preserved when integrating several components designed in today's methodologies.
2) driver and receiver design, and buffer (repeater) insertion:
   a) adequate driver and receiver design (such as [2]), as well as the insertion of repeaters alleviate the quadratic increase in propagation delay with interconnect length while decreasing power dissipation [1]. This technique satisfies the requirements we have set forth so far, but as we will see in the next section, this approach does not scale well with shrinking dimensions and increased clock speeds.

3 Global Wire Problem

As stated in the previous section the most common method for reducing delay in point-to-point interconnection with unidirectional signal flow is to split the wire into segments buffered by inverters (known as repeaters).

Critical wires and critical delays have been computed based on this optimally buffered interconnection line for the 0.25µm and the 0.1µm feature sizes by Otten and Brayton in [13] and are repeated in the Table below.

critical parameter	feature size 0.25 μ	feature size 0.10 μ
$l_{crit}(m1)$	10440	6757
$l_{crit}(m2)$	10600	7162
$l_{crit}(m3)$	36000	43446
$l_{crit}(m4)$	38400	45135
$l_{crit}(m5)$	63200	64932
$l_{crit}(m6)$	62000	56892
$l_{crit}(m7)$		97581
$l_{crit}(m8)$		93378
$T(l_{crit})$	205ps	80ps

Table 1: Critical Wire Lengths (in feature size units)

The l_{crit} shown for the different metal layers (m1-m8) is the wire length between repeater insertions. The delay it takes to cover that length is also shown in the table. If we perform some "back of the envelope" calculations for m8 for a 3GHz global clock design, we see that for a 3cm on the side 0.1µm chip only about 4 segments can be traversed in one clock cycle. These 4 segments amount to 37.4mm, which is about 1.2% of the chip width (and this is the fastest metal layer). This may not be suitable for long global wires since they could potentially be much longer than that (usually estimated to be, in the best case, side x $\sqrt{2}$, and in the worst case, half perimeter (i.e. 2X side)).

The argument above shows that, in the presence of long wires (more than one clock cycle to traverse), repeater insertion as well as driver/receiver adaptation techniques cannot hope to satisfy the functional timing constraints. This fact leads us to the next section where we propose a new strategy for high performance interconnect between IP blocks.

4 IP Integration Design Flow

The effects of wiring and interconnect, although quite problematic in deep sub-micron, have so far been neglected by the traditional System On Chip (SoC) design flow. The back-end of this flow has become a slow iterative scheme with no guarantee of convergence, and even if it does converge, the quality of the final solution is still uncertain. To address this, the authors recently proposed integrating floorplanning, coupled with re-synthesis and retiming techniques [19]. The figure below shows our proposed new design flow for IP integration in the SoC domain.

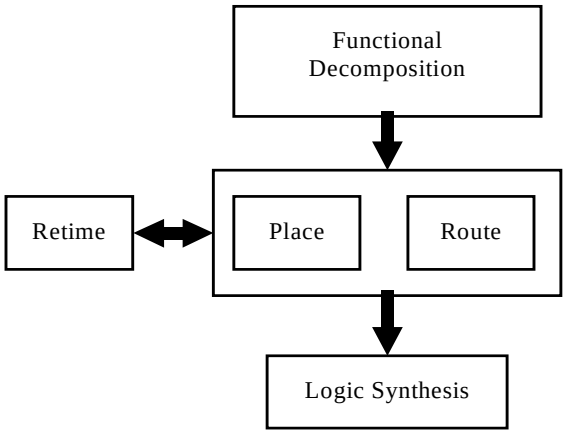

Figure 1: IP Design Flow

4.1 IP Interconnect Strategy

Retiming [18] and placement techniques [13] of modules will be able to handle satisfying delay constraints as long as interconnect delay is within bounds and can be properly characterized and predicted. To this end and to remedy the global wire delay problem presented in the previous sections, this work focuses on suggesting and then evaluating interconnect solutions. We have devised a registered interconnect strategy for the IP integration component-based design SoC domain. This strategy is presented in the subsection below.

4.1.1 Pipelined IP Interconnect (PIPE)

In deep submicron, global wires contribute significantly to the delay of a circuit and therefore need to be "retimed" in order to satisfy the functional timing constraint requirements. The idea is to insert registers (i.e. pipelining) within the (register bounded) global interconnect wires in order to reduce "perceived" delays thus permitting modules to meet constraints on the relative timing of inputs. We propose to use registers that:

1) are high performance,
2) have minimum area impact because of the large number of module input pins, and
3) low clock loading (to minimize clock distribution problems),
4) with small delay, and
5) low power consumption.

5 Circuit Implementation

We now focus on the implementation of the pipelined interconnect strategy. We describe in this section our implementation choices of the registers, and the test environment.

5.1 Driver/Receiver

As stated earlier, we require registers to be present at the boundaries of the IP's. This is a desirable requirement as explained before (i.e. straightforward synchronous integration), and can be easily satisfied by design. Therefore, in order to minimize cross talk effects and glitching, static (or pseudo-static) high-speed (edge-triggered) registers should be used at the IP boundaries [6]. In addition, the driver should be able to support the required fanout. These are the guiding metrics for the driver and receiver design choices. We leave the choice of register at the IP boundaries to the designer and assume standard CMOS line drivers. We focus next on the circuit implementation of the registers needed to support our proposed pipelined interconnect strategy.

5.2 IP Interconnect Strategy Implementation

5.2.1 True Single Phase Clock (TSPC) Latches

TSPC latches are commonly used in high performance digital systems due to their simplicity and fast operation [20]. The advantages that we see in this choice are: the single clock phase which avoids clock overlap problems, and the low clock loading (1 NMOS gate in the case of split-output TSPC). Figure 2 below shows the TSPC latch, and the split-output TSPC latch.

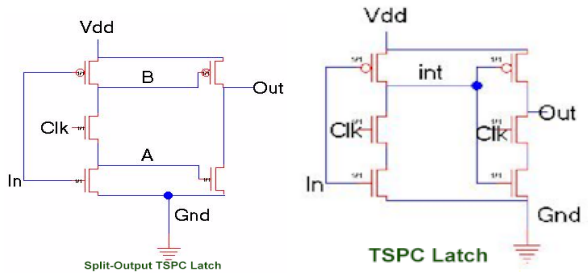

Figure 2: TSPC Latch (with and without split-output)

The split-output latch however not only has lower performance because of the threshold drop on the clocked NMOS: the presence of two interconnect wires internally increases the susceptibility to cross talk effects between the lines marked "A" and "B" in Figure 2 above. Because of this, we will not consider any split output solutions in the sequel even though they have half the clock loading of the regular TSPC latch.

We can generalize the above TSPC latch and recognize that there are 4 basic stages in TSPC latch or register design as shown below [22].

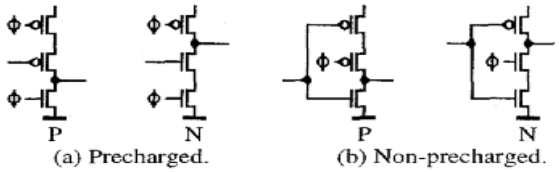

Figure 3: The basic four TSPC stages

A p-latch consists of two p-stages while an n-latch consists of two n-stages. A precharged latch is formed by a precharged stage followed by a non-precharged stage. A non-precharged latch is formed by two non-precharged stages. Registers are formed by the combination of these latches.

5.2.2 C²MOS-like Full Latch

The PN-SN-Full Latch (P)-INV register in Figure 4 has a C²MOS-like stage at the end to utilize the available precharged-node signal. This register is data-dependent during its evaluation phase but works perfectly for both one and zero inputs [23].

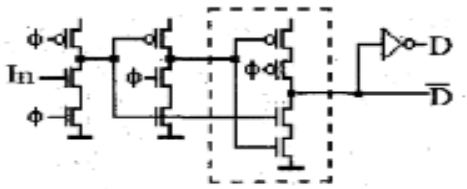

Figure 4: D Flip-Flop (PN-SN-FL(P))

5.2.3 Pipelines

Using the TSPC basic stages shown earlier we have devised several interconnect pipeline schemes. We present these configurations here, and we will in the next 2 sections present the test bench and the evaluation results.

For each single stage block we can have the combinations listed below. The notation below stands for the following:

- SN = Static N
- PN= Precharged N
- SP = Static P
- PP = Precharged P
- "-" = Delimiter between half stages
- Full Latch = C²MOS NORA stage

We have identified 4 basic schemes for implementing a positive edge register. The 4 basic schemes are:

1) SP-PN-SN (This is a D Flip Flop (DFF) as in Figure 5)
2) PP-SP-Full Latch (N)
3) SP-SP-SN-SN
4) PP-SP-PN-SN

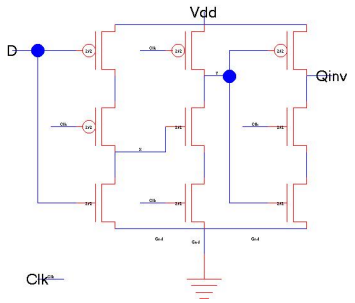

Figure 5: D Flip-Flop (SP-PN-SN)

These 4 schemes can each be implemented as:

- lumped: one block (i.e. internal wiring only), or
- distributed: multiple interconnected blocks.

These schemes can be implemented with or without coupling to account for crosstalk, for a total of 16 possible configurations for evaluation.

6 Evaluation Test Bench

We used a test bench to evaluate the proposed interconnect strategy by comparing the interconnect pipelines introduced in the previous section. The evaluation test bench is shown below.

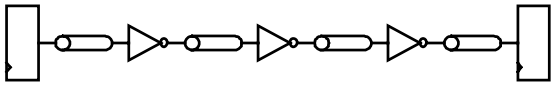

Figure 6: Evaluation Test Bench (no crosstalk)

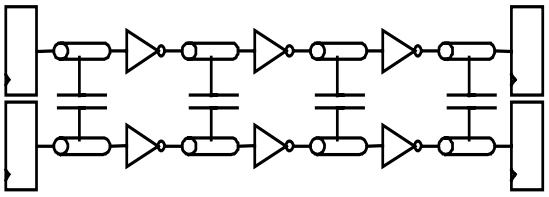

Figure 7: Evaluation Test Bench (crosstalk)

This test bench consisted in each case of four wire segments each of length *lcrit* (at m4 in 0.35µm technology *lcrit* is ~600λ = 600*0.35µm = 210µm = 0.21mm, where delay on a wire of *lcrit* length is ~250ps). Given the prediction for DSM designs, we picked a global clock of 500 MHz (intra-module clocks are projected to be one order of magnitude faster or ~5GHz) which amounts to ~4 *lcrit*'s (about ~0.84mm). Given a ~16mm chip dimension (at 0.35µm technology), we see that ~20 clock cycles to cross the chip is conceivable, as pointed out by Dally [4]. Note that the 4 *lcrit* and 20 global clock cycle estimate also holds for smaller scale technologies as pointed out earlier (see Section 3)

Interconnects were modeled as optimally buffered HSPICE lossy transmission lines. Crosstalk effects were measured between 2 sets of such wire segments, where the coupling capacitance was multiplied by 2 (worst case of 2 aggressors). We used 0.35µm technology: BSIM3 models for the devices, and interconnect parameters from [4]. We used Magic to layout the different stages, to get an estimate of device area, and capacitance.

7 Results

The results for the different pipeline schemes for one segment (i.e. wire + register) for a fanout of 4 inverters without cross talk is shown in the table below. The "Dist" rows refers to a distributed version of the register, i.e. buffers in the optimally buffered interconnect are replaced with register segments (SP, SN, PP, or PN). Power is reported per transition, while the delay is the "perceived" delay at the output. Area of active devices in a segment is reported in λ^2 where λ = 0.2µm (minimum drawn length is 2λ=0.4µm). "Buffered" is the optimally buffered 0.84mm m4

interconnect wire (*L*= 4**lcrit*). The interconnect is clocked at 0.5GHz.

Pipeline Scheme	Area (λ^2)	Power (µW)	Delay (ns)	Clock Load	Area x Power	Area x Power x Delay
Dist.PP	2024	1.34	0.80	4	2712.16	2169.73
Dist.SP	2024	1.34	0.91	4	2712.16	2468.07
Buffered	1848	1.51	1.40	0	2790.48	3906.67
Dist. DFF	1980	1.42	0.56	4	2811.60	1574.50
DFF	2904	1.81	0.39	4	5256.24	2049.93
PP-SP-Full Latch (N)	3036	1.85	0.45	4	5616.60	2527.47
PP-SP-PN-SN	3410	1.93	0.39	4	6581.30	2566.71
SP-SP-SN-SN	3410	1.94	0.40	4	6615.40	2646.16

Table 2: Without Coupling (2ns global clock cycle)

The three different metrics shown above are: delay, area, and power. In order to compare the different implementation, we augmented the table with two additional comparison criteria (Area x Power, Area x Power x Delay). The first criterion would be used in the case where the "perceived" delay is not of primary concern. In this case, the results in Table 2 (sorted by Area x Power metric) show that the distributed solutions are superior, especially the 4 stage register solution that completely replaces the intermediate buffers (thus reducing device count, and switching activity due to clocking, as well as intermediate latching). To better understand the results and to show the full range of trade-offs that could possibly be taken advantage of in an IP based integration environment, we plotted in Figure 8 the delay, area, and power trade-off surface, by performing cubic interpolation on the collected data. It can be seen that the distributed solutions (in particular distributed PP-SP-PN-SN and SP-SP-SN-SN), are better in general (low power with average "perceived" delay overhead).

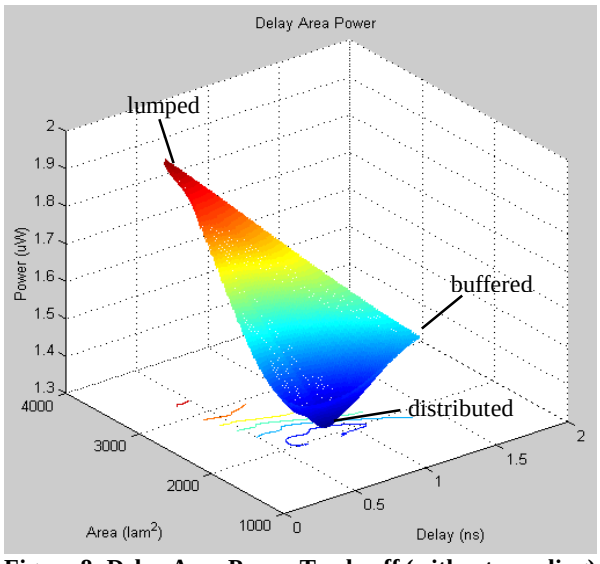

Figure 8: Delay Area Power Trade-off (without coupling)

The worst case cross talk is taken into account in the next table where the coupling capacitance is 2*C_C (2 aggressors as stated earlier). The interconnect is clocked at 0.25GHz. The reduction in global clock frequency is the result of delay introduced by crosstalk coupling between the two aggressor lines and victim driven line. In this case a doubling of the delay was encountered and subsequently a halving of the clock frequency was need to remedy this.

Pipeline Scheme	Area (λ^2)	Power (µW)	Delay (ns)	Clock Load	Area x Power	Area x Power x Delay
Dist. DFF	1980	2.30	1.20	4	4554.00	5464.80
Dist.SP	2024	2.40	1.91	4	4857.60	9278.02
Buffered	1848	3.16	2.87	0	5839.68	16759.88
Dist.PP	2024	3.40	0.96	4	6881.60	6606.34
DFF	2904	3.41	0.39	4	9902.64	3862.03
PP-SP-Full Latch (N)	3036	3.57	0.46	4	10838.52	4985.72
PP-SP-PN-SN	3410	3.46	0.35	4	11798.60	4129.51
SP-SP-SN-SN	3410	3.54	0.39	4	12071.40	4707.85

Table 3 : With Coupling (4ns global clock cycle)

From the results of Table 3, a new ranking can be observed based on the Area x Power metric. To visualize these results we plotted the gathered data, and interpolated as shown in the delay, area, and power surface of Figure 9. In this case, there is a difference between the various distributed solutions, where the non-precharged schemes are clearly more power-efficient than their precharged counterparts, with acceptable "perceived" delay. The precharged solution requires more energy since the precharging affects neighboring lines, making this solution more susceptible to crosstalk effects.

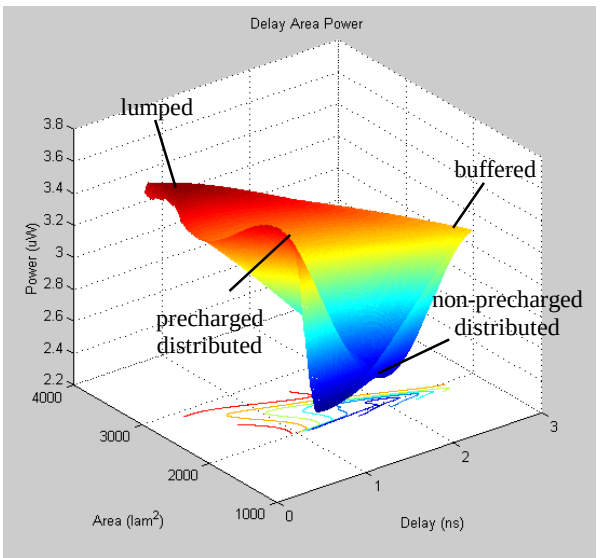

Figure 9: Delay Area Power Trade-off (with coupling)

After the crosstalk results of Table 3, we can conclude that in the distributed solutions (favored by the results of Table 2 where coupling was not taken into consideration), a precharged stage at the segment input (e.g. distributed PP-SP-PN-SN and PP-SP-Full Latch (N)) is not a robust pipelined interconnect solution. This is because the input precharge stage is data dependent during evaluation, and is therefore very susceptible to glitches at the input.

8 Summary and Conclusions

We proposed a new IP interconnect strategy for SoC applications composed of heterogeneous components (hard, firm, and soft). We also presented the pipelined interconnect (PIPE) scheme as a solution to signal delay and degradation in long global interconnect in deep sub-micron. The goal was to improve performance while keeping ease of IP reuse. The main idea is to solve the large latency problems of global wires through pipelining, with a constant "perceived" delay independent of the length of the wire.

We then proceeded to evaluate several circuit implementations of the pipeline positive-edge-triggered registers. The pipeline registers are all rooted in the 4 basic dynamic TSPC stages, and therefore are high-speed registers. In addition they are small in area, have small power consumption, and low clock loading. We evaluated these different schemes with respect to the three main design metrics (area, power, and "perceived" delay) and found a range of possible solutions that can potentially be used in a trade-off optimization setting. In the case where power is the most important measure with acceptable "perceived" delay, we declared clear winner schemes (distributed non-precharged solutions if crosstalk is considered). These are distributed SP-NP-SN (with an additional buffer), and in second place distributed SP-SP-SN-SN.

References

[1] Adler, V.; Friedman, E.G. "Repeater Design to Reduce Delay and Power in Resistive Interconnect", IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing, vol.45, (no.5), IEEE, May 1998. p.607-16.

[2] Bellaouar, A.; Abu-Khater, I.S.; Elmasry, M.I. "Low-power CMOS/BiCMOS Drivers and Receivers for On-chip Interconnects", IEEE Journal of Solid-State Circuits, vol.30, (no.6), June 1995. p.696-700.

[3] Cappuccino, G.; Cocorullo, G.; Corsonello, P. "CMOS Buffer Sizing for Long On-chip Interconnects", Electronics Letters, vol.34, (no.20), IEE, 1 Oct. 1998. p.1937-8.

[4] Dally, W., Poulton, J. "Digital Systems Engineering", Cambridge University Press, 1998.

[5] Dennis Sylvester, and Kurt Keutzer, "Getting to the Bottom of Deep Submicron", ICCAD, 1998.

[6] Katsuichi Tomobe, Toshiro Takahashi, Masatoshi Kawashima, Yasuo Sonobe, Tadashi Kiyuna, Sigeru Yamamoto, "A 1860kG CMOS Gate Array with GTL Input Flip-Flop Circuits", IEEE Custom Integrated Circuits Conference, 1996.

[7] Keating, M. "Design Reuse: Can It Halt the SoC Train Wreck ?", EE290A, "Component-Based Design", UC Berkeley, Spring 1999.

[8] Khatri, S. "A Deep Sub-Micron VLSI Design Flow Using Noiseless Layout Fabrics", Ph.D. Qualifying Examinations, UC Berkeley, Spring 1999.

[9] Kim, S.; Sridhar, R., "Hierarchical Synchronization Scheme Using Self-timed Mesochronous Interconnections", ISCAS, 1997.

[10] Kirkpatrick, D.A.; Sangiovanni-Vincentelli, A.L. **"**Digital Sensitivity: Predicting Signal Interaction Using Functional Analysis", ICCAD, 1996.

[11] Kohyama, Susumu, "IP Revolution and Interconnect Technology", International Interconnect Technology Conference (IITC), 1998.

[12] NexSIS CAD Group, University of California at Berkeley, Principal Investigators: Robert K. Brayton, and Kurt Keutzer, 1999.

[13] Ralph H.J.M. Otten, and Robert K. Brayton, "Planning for Performance", DAC, 1998.

[14] Rolf Ernst "Embedded System Architectures" in "Hardware/Software Co-Design: Principles and Practice", Kluwer Academic Publishers, 1997.

[15] Schoellkpf, Jean-Pierre, "Impact of Interconnect Performances on Circuit Design", International Interconnect Technology Conference, 1998.

[16] Semiconductor Industry Association, "The National Technology Roadmap for Semiconductors", 1997.

[17] SRC Physical Design Task Force: Ray Abrishami, LSI Logic; Eli Chiprout, IBM Corporation, Mike Jassowski, Intel Corporation; Bernd Koenemann, LogicVision Corporation; ChiCheng Liaw, Lucent Technologies; Sury Maturi, National Semiconductor; Ching-Hao Shaw, Texas Instruments Incorporated, "Draft Report on Physical Design", May 1997.

[18] Tabbara, A. "Retiming for DSM with Area-Delay Tradeoffs and Delay Constraints", DAC, 1999.

[19] Tabbara, A., Tabbara B. "Integration of Retiming with Architectural Floorplanning: A New Design Methodology for DSM", 1999.

[20] Y. Ji-Ren, I. Karlsson, C. Svensson, "A True Single-Phase-Clock Dynamic CMOS Circuit Technique", IEEE JSSC, 1987.

[21] Yen-Tai Lai; Chi-Chou Kao; Tsun-Cheng Chang; Kun-Nern Chen, "A Field Programmable Gate Array Chip with Hierarchical Interconnection Structure", ISCAS, 1998.

[22] Jiren Yuan, Christer Svensson, "Fast and Robust CMOS Double Pipeline Using New TSPC Multiplexer and Demultiplexer", International Conference on ASIC, 1996.

[23] Jiren Yuan, Christer Svensson, "New Single-Clock CMOS Latches and Flipflops with Improved Speed and Power Savings", IEEE JSSC, 1997.