High-quality circuit synthesis for modern technologies

Citation for published version (APA):

Jozwiak, L., Chojnacki, A., & Slusarczyk, A. S. (2008). High-quality circuit synthesis for modern technologies. In *9th International Symposium on Quality Electronic Design, 2008 : ISQED 2008 ; 17 - 19 March 2008, San Jose, California* (pp. 168-173). IEEE Computer Society. <https://doi.org/10.1109/ISQED.2008.4479720>

DOI:

[10.1109/ISQED.2008.4479720](https://doi.org/10.1109/ISQED.2008.4479720)

Document status and date:

Published: 01/01/2008

Document Version:

Publisher's PDF, also known as Version of Record (includes final page, issue and volume numbers)

Please check the document version of this publication:

- A submitted manuscript is the version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher's website.
- The final author version and the galley proof are versions of the publication after peer review.
- The final published version features the final layout of the paper including the volume, issue and page numbers.

[Link to publication](#)

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal.

If the publication is distributed under the terms of Article 25fa of the Dutch Copyright Act, indicated by the "Taverne" license above, please follow below link for the End User Agreement:

www.tue.nl/taverne

Take down policy

If you believe that this document breaches copyright please contact us at:

openaccess@tue.nl

providing details and we will investigate your claim.

High-quality Circuit Synthesis for Modern Technologies

Lech Jozwiak¹, Artur Chojnacki², Aleksander Slusarczyk¹

¹Eindhoven University of Technology, ²PDF Solutions Inc.

L.Jozwiak@tue.nl

Abstract

Due to weaknesses in circuit synthesis methods used in today's CAD tools, the opportunities created by modern microelectronic technology cannot effectively be exploited. This paper considers the issues and requirements of circuit synthesis for the nano CMOS technologies, and discusses our new circuit synthesis technology that satisfies these requirements. The new technology considerably differs from all other known synthesis methods and overcomes their main weaknesses. The experimental results demonstrate that it produces very fast, compact and low-power circuits. The new technology has however many more major advantages that are discussed in the paper.

1. Introduction

Introduction of the nano CMOS technologies created unusual opportunities, but also resulted in many new difficult to solve issues and changed importance relationships among various circuit characteristics. In consequence, it largely invalidated the traditional circuit synthesis technologies [8]. This paper is devoted to an adequate synthesis of digital circuits for the modern nano CMOS circuit implementation technologies. It briefly considers the issues and requirements of circuit synthesis for the nano CMOS technologies, and discusses our new quality and information driven circuit synthesis technology that satisfies these requirements. It compares the experimental results from our new circuit synthesis tool to the results from some other tools, demonstrates that the results from our new tool are superior, discusses the major advantages of our new circuit synthesis technology, and shows that this technology is adequate for the modern nano CMOS synthesis targets.

2. Issues and requirements of circuit synthesis for modern technologies

The recent spectacular advances in microelectronics introduced unusual silicon and system complexity, and particularly: an extremely high number of various devices and interconnects on a chip, extremely high device and interconnect densities, extremely small devices' dimensions, and huge length of interconnects. Due to this unusual complexity, as well as, interconnect scalability problems, power supply reduction and very high operating frequencies, many previously ignorable phenomena have now a great impact on the circuit and system correctness and other quality aspects [8]. This results in **many new difficult to solve issues**, including: *power and energy crisis, increased leakage power, interconnect scalability problems and dominating influence of interconnects on major physical circuit and system characteristics* (e.g. area, speed, ...), etc. Additional challenges of the sub-wavelength lithography and process variability are expected below 65 nm [8]. To temper the negative effects of the sub-wavelength lithography imperfections and process variability, enhanced circuit regularity and simplicity may be required. ***These problems cannot be resolved without more adequate methods and EDA-tools for designing of area, power and speed optimized circuits that will effectively address the new issues, and enable effective tradeoff exploitation among different circuit optimization objectives and total quality maximization.***

Regarding the circuit power consumption observe the following. The circuit *leakage power is roughly proportional to the circuit area*, because it is proportional to the number of leaking transistors. It also *depends on the states of logic element input signals*. The *dynamic power* is proportional to the signal activity and parasitic capacitances of transistors and interconnects. Consequently, it is *roughly proportional to the total number of transistors and total length of interconnects*, i.e. *roughly proportional to the circuit area*. Consequently, ***the circuit synthesis***

techniques minimizing the circuit area, number and length of interconnects, and signal activity, and selecting signal polarities for low leakage states of logic elements will be of primary importance for the current and future nano CMOS technologies.

Unfortunately, the available circuit synthesis methods and tools do not well address the needs of circuit synthesis for the modern technologies, due to:

- *not accounting for the recently changed importance relationships among various circuit characteristics* (e.g. the static power being negligible in the past technologies is becoming dominating, interconnects being of the secondary importance in the past have now a dominating influence on major circuit characteristics);
- *not applying the now necessary multi-objective circuit optimization and trade-off exploitation*;
- *being not effective for many classes of circuits* - this was several times demonstrated by us, and once more in a recent paper by J. Cong and K. Minkovich [2].

Moreover, in modern circuit implementation technologies the constraints are not imposed on the function type a logic block can implement, but on the structural dimensions of logic blocks and interconnections between the blocks. For instance, an FPGA logic block is able to implement any function with limited input dimensions and an ASIC library delivers very many functions with limited dimensions. *The modern circuit implementation technologies deliver maximal or reach functionally complete systems, while the traditional logic synthesis methods applied in the most of contemporary industrial and academic EDA-tools construct circuits using the minimal functionally complete systems* (e.g. AND+OR+NOT). Additionally, the traditional logic synthesis methods and tools:

- do not well account for hard structural constraints of logic blocks and for interconnections,
- do not explicitly account for timing and power, and use some proxy attributes for area that often do not well correlate with the actual area,
- consider only some special cases of possible implementation structures: make many prior assumptions excluding many network structures.

In result, *the proxy synthesis targets of the available logic synthesis methods and tools very much differ from the actual synthesis targets of circuits implemented in modern technologies.* In consequence, the circuits synthesized with the available methods and tools require a substantial post synthesis technology mapping effort. Unfortunately, *the technology mapping can not guarantee proper final results*, because the

initial circuit synthesis is performed without close relation to the actual synthesis target.

From the above it should be clear that **a new much more adequate circuit synthesis technology is needed for the modern circuit implementation technologies that will enable the following:**

- *consideration of all possible circuit implementation structures during the synthesis* (not making any prior assumptions during the method or tool development that exclude some possibly very good structures from consideration);
- *direct synthesis into specific technology targets* when accounting for the constraints and characteristics of the target's logic building blocks and interconnects;
- *synthesis of robust more regular circuits with minimized interconnects*;
- *explicitly accounting for the actual area, timing and power related information*;
- *the total multi-objective optimization of the circuit's quality* in relation to the circuit interconnects, area, speed, power-dissipation, and possibly some other circuit's features;
- *effective tradeoff exploitation among the different optimization objectives.*

According to our knowledge such a circuit synthesis technology did not exist till now: none of the commercial circuit synthesis tools has the above features and no information on any technology having these features has been published.

3. Information-driven circuit synthesis

To satisfy the needs of an adequate circuit synthesis for the modern nano CMOS technologies, we proposed a **new information-driven circuit synthesis approach** and developed two theories that support this approach:

- *the theory of general decomposition of discrete relation networks* [3], and
- *the theory of information relationships and measures* [4].

Using them, we developed a family of circuit synthesis methods and tools that implement the information-driven approach. One of these methods and corresponding EDA-tool will be discussed in this paper.

The information-driven circuit synthesis approach relies on the analysis of the information flow structure and relationships in the function to be implemented, as well as, in the circuit under construction, and usage of the results of this analysis to control the circuit construction. Information flows in the circuit are appropriately ordered, combined, compressed and kept as local as possible. In this way both interconnections

and active elements are minimized. **The information-driven approach uses:**

- **general decomposition generator** that is able to generate *all correct circuit structures for a given function* (no structures are excluded a priori);
- **information relationships and measures** to control the generator in order to *efficiently construct only the most promising circuit structures*;
- **timing, power and area related information** (e.g. physical gate/LUT characteristics, signal arrival and required times, signal activity etc.) to control the *satisfaction of the optimization constraints and objectives*, and enable the *multi-objective optimization* and *trade-off exploitation*.

Due to usage of general decomposition, a circuit node can represent any function that satisfies certain specific structural constraints of a given implementation technology. The circuit synthesis is not divided into the technology independent logic synthesis and technology mapping, but is directly performed into the primitives of a given implementation technology (e.g. LUTs and CLBs of a given FPGA or gates of a given technology library). Information relationships and measures make it possible to control the circuit convergence, compactness and interconnections. Our approach minimizes both the number and length of interconnections and explicitly accounts for area, timing and power consumption. Since in parallel to information relationships and measures any sort of additional information can be accounted for (as e.g. related to the signal timing or activity), the timing and power driven synthesis, as well as very flexible and precise delay, power and area tradeoffs are possible. In consequence, the circuits synthesized are small, ultra-fast and low-power at the same time. **This all together fulfils the requirements of an adequate circuit synthesis for the modern nano CMOS technologies** as formulated in Section 2.

The apparatus of information relationships and measures facilitates the analysis and quantitative measurement of the information flows and their relationships. Some main ideas of this apparatus are briefly introduced below. Let us consider a finite set of elements S , called symbols. Information about symbols pertains to the ability to distinguish certain symbols

from other symbols. Table 1 shows the truth table of a multi-output Boolean function. Each row of the truth table (function's product term) is represented by a unique symbol from S . Through its two values 0 and 1, variable x_i induces two compatibility classes on the symbols (terms): $B^0 = \{0, 2, 3, 4\}$ and $B^1 = \{1, 2, 3, 5\}$. x_i has value 0 (1) for each symbol in class B^0 (B^1) (don't care '–' means: 0 and 1). Variable x_j is not able to distinguish between symbols 0, 2, 3, and 4, because they belong to the same compatibility class. x_j is able to distinguish between 4 and 5, because they are not placed together in any compatibility class. In this way information is modeled with set systems [3][4]. An *elementary information* describes the ability to distinguish a certain single symbol s_i from another single symbol s_j ($s_i, s_j \in S$ and $s_i \neq s_j$). Any set of such atomic portions of information can be represented by an *information set* IS defined on $S \times S$ as follows [4]: $IS = \{\{s_i, s_j\} \mid s_i \text{ is distinguished from } s_j \text{ in the information modeled}\}$. For instance, information given by set system $\pi_{x_i} = \{0, 2, 3, 4; 1, 2, 3, 5\}$ induced by x_i , can be represented by information set $IS(\pi_{x_i}) = \{0|1 \ 0|5 \ 1|4 \ 4|5\}$. Information relationships between variables or set systems representing various information streams can be analyzed by considering relationships between their corresponding information sets. In particular, the relationship and relationship measure expressing information similarity are defined in [4] as follows:

- *common information* CI (i.e. information that is present in both π_1 and π_2): $CI(\pi_1, \pi_2) = IS(\pi_1) \cap IS(\pi_2)$
- *information similarity (affinity) measure* ISIM: $ISIM(\pi_1, \pi_2) = |CI(\pi_1, \pi_2)|$.

In real applications, we use some more complex normalized and weighted measures obtained through associating an appropriate importance weight $w(s_i|s_j)$ with each elementary information and combining selected simple measures.

Example 1. (information modeling and analysis)

The corresponding set systems and information sets for all inputs and outputs of the Boolean function shown in Table 1 are as follows:

$$\begin{aligned}\pi_{f_1} &= \{0, 1, 2, 3, 4, 5\}, & \pi_{f_2} &= \{0, 1, 2, 3, 4, 5\}, \\ \pi_{x_1} &= \{0, 2, 3, 4; 1, 2, 3, 5\}, & \pi_{x_2} &= \{0, 2, 5; 1, 3, 4\}, \\ \pi_{x_3} &= \{0, 3, 5; 1, 2, 4\},\end{aligned}$$

$$\begin{aligned}IS(\pi_{f_1}) &= \{0|4 \ 0|5 \ 1|4 \ 1|5 \ 2|4 \ 2|5 \ 3|4 \ 3|5\}, \\ IS(\pi_{f_2}) &= \{0|2 \ 0|3 \ 0|4 \ 0|5 \ 1|2 \ 1|3 \ 1|4 \ 1|5\}, \\ IS(\pi_{x_1}) &= \{0|1 \ 0|5 \ 1|4 \ 4|5\}, \\ IS(\pi_{x_2}) &= \{0|1 \ 0|3 \ 0|4 \ 1|2 \ 1|5 \ 2|3 \ 2|4 \ 3|5 \ 4|5\}, \\ IS(\pi_{x_3}) &= \{0|1 \ 0|2 \ 0|4 \ 1|3 \ 1|5 \ 2|3 \ 2|5 \ 3|4 \ 4|5\}, \\ CI(\pi_{f_1}, \pi_{x_1}) &= \{0|5 \ 1|4\}, \quad ISIM((\pi_{f_1}, \pi_{x_1})) = 2.\end{aligned}$$

Table 1 Example 3-input, 2-output Boolean function f

S	$x_1 \ x_2 \ x_3$	$f_1 \ f_2$
0	0 0 0	0 0
1	1 1 1	0 0
2	– 0 1	0 1
3	– 1 0	0 1
4	0 1 1	1 1
5	1 0 0	1 1

In a single step of general functional decomposition, function f being decomposed is split into two sub-functions (see Fig. 1): **predecessor sub-function** g and **successor sub-function** h . The input support of f is divided into two subsets: **bound-set** U , being the g 's input support, and **free-set** V , being a partial input support of h . Outputs of g constitute the remaining part of the h 's support. This single decomposition step is recursively applied to both predecessor and successor functions until each sub-function in the network constructed this way can be directly mapped onto LUTs of a given FPGA or gates in a given technology library.

Our circuit synthesis constructs the circuit level by level from its primary inputs to primary outputs (bottom-up) through repeating the single decomposition step (Fig. 2). The bottom-up synthesis enables continuous precise control of timing, area and power consumption during the synthesis. The continuous availability of precise information on the timing, area and power consumption of the already synthesized bottom part of the circuit, as well as, on the corresponding characteristics of support signals for its upper (to synthesize) part enable very well informed synthesis decisions and very precise tradeoff exploitation. In the support of a certain level, only the variables from any lower level (primary inputs and/or logic blocks' outputs) can be used. The output variables of the logic blocks already built at the current level – that constitute the **cover-set** C – cannot be used in any bound-set of this level (see Fig. 1 and 2). At each level, the input support (primary inputs and/or intermediate variables) of the not yet synthesized part of a function being decomposed has to provide all information necessary to compute the function's output values. However, information necessary for computing the function's values is distributed across its support variables. These variables also contain some redundant information. To implement the function, the decomposition network has to eliminate the redundant information present in the function's inputs, and

Fig. 1 Single step of the general functional decomposition

Fig. 2 Bottom-up functional decomposition

preserve and restructure the required information, to finally represent the required information at the output as demanded by the function. Consequently, each sub-function g should eliminate some redundant information and combine the required information from its inputs, transfer the required information to its output and represent it in an appropriate manner. The **bound-set** U determines *what information is delivered* to a certain sub-function g . The g 's output set system π_g determines *what information is transferred* by g . U and π_g together define the multi-valued function of g . In order to implement this function in binary hardware, it has to be transformed into a set of binary functions. The g 's binary functions determine *how the transferred information is represented* at the g 's binary outputs [5].

The **sub-function construction procedure** is composed of the following steps:

1. Construct a limited set of the most promising bound-sets U and corresponding output set systems π_g .
2. Order the input supports U from the set constructed in step 1 according to their quality.
3. Consider a limited set of the supports U in the order of their quality, and for each support U construct a set of the corresponding binary (LUT or gate) implementations of the multi-valued sub-function g .
4. From the set of implementations constructed in step 3, select the implementation that maximizes the signal and/or information convergence of the sub-function g and optimizes a given area/delay/power tradeoff.
5. Construct a new function h by expressing f in new variables.

The first four steps are guided by analysis of the information relationships, LUT or gate characteristics, area, timing and power related information, and optimization constraints and objectives. They were explained in our previous publications [4][5]. The last step is straightforward.

4. Experimental results

The circuit construction method discussed in the previous sections has been implemented in our information-driven circuit synthesis tool in two versions:

- FPGA targeted: referred to as **IRMA2FPGAS** (Information Relationship Measures Applied to FPGA Synthesis), and
- technology gate library targeted: referred to as **IRMA2GATES**.

Below we compare some results from our **IRMA2FPGAS** to the results from the well known UC Berkeley's tool **SIS 1.3** [6] and from the newest UC Berkeley's tool **abc**, regarding the LUT count (area) and number of LUT levels (delay) for the 5-input LUTs. For **SIS** we used the script dedicated to the LUT-based architectures proposed in [6] and for **abc** two scripts: **standard** script (**abc**) and **choice** script (**abc-ch**). Fig. 3 presents the comparison of the results obtained for the MCNC benchmark set [7]. Results of this experiment demonstrate that our **IRMA2FPGAS** constructs much better circuits than **SIS** and **abc**, both regarding the circuit area and delay. For the MCNC benchmark set, the circuits produced by **IRMA2FPGAS** are on average over 2 times faster and have 50% less CLBs than the circuits synthesized by **SIS**, and are 12-15% faster and have almost 60% less CLBs than the circuits synthesized by **abc**. Fig. 4 shows that **IRMA2FPGAS** much more outperforms both **SIS** and **abc** on another more representative benchmark set of 200 circuits representing different kinds of logic from various applications (filters, arithmetic logic, condition logic, function look-up-tables, next-state and output logic of differently encoded FSMs etc.). The circuits produced by **IRMA2FPGAS** for this benchmark set are on average over 2 times faster and have over 2 times less CLBs than the circuits synthesized by **SIS**, and are 30% faster and have over 2 times less CLBs than the circuits synthesized by **abc**. In Fig. 5 and 6, example circuits are presented synthesized with our tool **IRMA2FPGAS** and with **abc** for MCNC benchmark 9sym. We also compared **IRMA2FPGAS** to several FPGA-targeted commercial tools on various benchmarks. **IRMA2FPGAS** consistently produced substantially better circuits than the commercial tools (see e.g. [5]). The results from the gate library targeted version of our tool are very similar to the results from the FPGA-targeted version, and therefore we do not present them in this short conference paper. Please observe that the circuits synthesized by our tool have much smaller area, less logic levels, much less interconnects and shorter interconnects than the circuits from the other

tools, and this all results in their substantially lower power dissipation. The circuits from our tool are small, quick and low-power at the same time.

5. Major advantages of the new technology

The main advantage of our new information-driven circuit synthesis technology is of course the **superior result quality**. However, this is not its only major advantage. Our information-driven circuit synthesis technology has several other **important unique features that are absent in the currently available circuit synthesis tools**, including the following:

- **generality and very high flexibility**: a natural ability to account for all possible circuit realization structures and realize any possible tradeoff among the circuit area, power consumption and speed;
- **high synthesis precision**: an ability to very precisely structure the circuit and tune it in relation to the actual objectives and constraints;
- **direct synthesis into the technology primitives of a given circuit implementation technology** (e.g. LUTs or gates);
- **very effective and efficient processing of incompletely specified functions**;
- **minimization of the number and length of interconnections**;
- **simplicity and regularity of the circuit structures synthesized**;
- **enhanced route-ability, low usage of resources, high-speed and low power consumption** resulting from the circuits compactness, regularity, and minimized interconnects.

6. Conclusion

We developed a new effective, efficient and very flexible circuit synthesis technology adequate for the modern synthesis targets. The technology implements our original information-driven approach to circuit synthesis. The experimental results demonstrate its high quality. Our tools construct substantially smaller, lower-power, and faster circuits than other tools, and enable a very flexible timing- and power-driven circuit structuring, re-structuring, and trade-off exploitation.

7. References

- [1] Berkeley Logic Synthesis and Verification Group, ABC: A System for Sequential Synthesis and Verification, Release 70909, <http://www.eecs.berkeley.edu/~alanmi/abc>
- [2] J. Cong and K. Minkovich: Optimality Study of Logic Synthesis for LUT-Based FPGAs, FPGA'06, February 22-24, 2006, Monterey, California, USA, ACM, pp. 33-40.
- [3] L. Jóźwiak, A. Ślusarczyk: General Decomposition of Incompletely Specified Sequential Machines with Multi-State Behaviour Realisation, Journal of Systems Architecture, Vol. 50, December 2003, pp. 445-492.
- [4] L. Jóźwiak: Information Relationships and Measures - An Analysis Apparatus for Efficient Information System Synthesis, *23rd EUROMICRO Conference*, Budapest, Hungary, September 1-4, 1997, IEEE Computer Society Press, pp. 13-23.
- [5] L. Jóźwiak, A. Chojnacki: Effective and efficient combinational circuit synthesis for the FPGA-based reconfigurable systems, Journal of Systems Architecture, Special Issue on Reconfigurable Computing Vol. 49 No 4-6, 2003, pp. 247-265
- [6] E. M. Sentovich, K. Singh, L. J., Lavagno, C. Moon, R. Murgai, A. Saldanha, H. Savoj, P. R. Stephan, R. K. Brayton, A. Sangiovanni-Vincentelli, SIS: A system for sequential circuit synthesis, Memorandum No. UCB/ERL M92/41, Electronic Research Laboratory, University of California, Berkeley. <ftp://ic.eecs.berkeley.edu/pub/Sis/Sis-paper.ps.Z>
- [7] Collaborative Benchmarking Laboratory, Department of Computer Science at North Carolina State University, <http://www.cbl.ncsu.edu/>
- [8] The 2005 International Technology Roadmap for Semiconductors, SIA, San Jose, CA, USA, 2005, <http://www.itrs.net/Links/2005ITRS/Home2005.htm>

Acknowledgements: This paper has been prepared in the private capacity of its authors. The authors are indebted to the Eindhoven University of Technology for partially supporting their early research on the theories and preliminary prototypes that initiated development of the circuit synthesis technology discussed in the paper, and the paper presentation.