Design of Standard-Cell Libraries for Asynchronous Circuits with the ASCEnD Flow

Matheus Trevisan Moreira, Ney Laert Vilar Calazans
GAPH – Faculty of Informatics – Pontifical Catholic University of Rio Grande do Sul - Porto Alegre – Brazil
matheus.moreira@acad.pucrs.br, ney.calazans@pucrs.br

***Abstract*—This work presents the ASCEnD flow, a design flow devised for the design of components required for the design of asynchronous systems using standard-cells. The flow is fully automated except for the layout generation step, and can be parameterized for any CMOS technology. It was employed in the design of a dedicated standard-cell library, which contains over five hundred components, for the STMicroelectronics 65 nm technology. This library supported implementation of different circuits, like network-on-chip routers and cryptographic cores.**

Keywords—standard-cell; design flow, asynchronous

I. INTRODUCTION

The semi-custom design flow of integrated circuits is often referred as a key factor for the rapid growth of integrated circuits and systems [1]. This flow lowers design complexity by using pre-designed and pre-characterized functional components called standard cells, instead of assuming that designers have to draw, place and connect each transistor by hand [2]. In this way, complex systems can be much more easily modeled. As the CMOS technology evolves into deep submicron nodes, asynchronous techniques gain relevance in the research community, due to their ability to cope with problems that are hard to solve with the synchronous paradigm [3-4]. However, several specific components required by asynchronous design are unavailable in commercial standard cell libraries, constraining asynchronous design to use approaches close to full-custom. This limits modularity and increases design complexity. Thus, one of the factors that can enable further acceptance of the asynchronous paradigm is the availability of asynchronous standard cell libraries.

This work presents the ASCEnD flow, devised for generating asynchronous circuits components at the standard-cell level. The flow uses automated tools, which are parameterizable for any CMOS technology, the only remaining manual step being layout generation. With the ASCEnD flow, standard cell libraries can be produced more easily. This flow has been employed in the design of a library with over five hundred components for the STMicroelectronics 65nm technology. It is used in conjunction with the technology native basic standard cell library, herein called *corelib*. ASCEnD is also currently being used for the generation of a library targeting the IBM 130nm technology, compatible with the MOSIS service [5], to ease the library and design flow validation on silicon.

II. THE ASCEND DESIGN FLOW

The first step in the ASCEnD flow is the process of setting the dimensions of the component transistors. These dimensions vary depending on the required driving strength, the capability of charging/discharging loads in a given period of time. The bigger the driving strength, the bigger the load the standard-cell will be able to charge/discharge in a same period of time. Usually, standard-cell components are available at different driving strengths for a same logic function, to provide more optimization opportunities in operating speed and power.

Fig. 1 shows a basic structure for the components required for an asynchronous design. These usually employ a pull-up (PUN) and pull-down (PDN) transistor networks, for implementing the component functionality, and a hysteresis mechanism based on a back to back inverters loop, to guarantee the stability of the output for unbounded periods of time. The transistors of the PUN and the PDN are those that are dimensioned by the presented flow. The transistors of the loop inverters have dimensions of the corelib cells. For inverted gates, given the inverting nature of CMOS logic, the output of the gate is the node right after the pull-up (PUN) and pull-down (PDN) logic stacks. Therefore the inverters of the loop are assumed to have the dimensions of those with smallest driving strength available in the corelib, to present lower interference in the standard-cell performance. As for non-inverting gates, the output is given by one of the inverters of the loop (in bold). In this way, the transistors of this inverter are dimensioned according to those of a corelib inverter with a same driving strength. The other inverter, used only for the feedback loop, is kept with minimum dimensions. Based on these assumptions, a schematic of the component is designed in SPICE at transistor level and is used as input for the ASCEnD flow. Fig. 2 shows the basic steps followed.

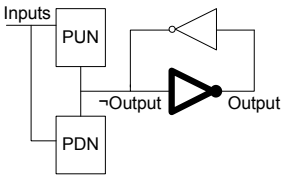

Fig. 1. Basic structure of a component required for asynchronous circuits.

From the provided schematic, an in-house tool, called ROGen, generates a simulation circuit and environment. This environment is set to measure the standard cell power and delay figures. The Cadence Spectre tool simulates the generated circuit and the results of the measurements are input to another in-house tool called CeS. At this step, the designer provides the tool with a cost function, which is given as an operating speed and power tradeoff. Based on this function, the tool selects the most suited transistor dimensions. The tool also plots charts with the cost function for varying transistors sizes and allows incremental cost functions computations.

978-1-4799-1331-2/13/$31.00 ©2013 IEEE

Once the transistors are dimensioned, the layout of the standard cell can be generated, using e.g. Cadence Virtuoso, and validated, using design rule check (DRC) and layout versus schematic (LVS) tools, like Mentor Calibre DRC and LVS. Also, it is recommended that design for manufacturability (DFM) rules are employed during this step. Once the correct layout is verified, parasitics can be extracted, using for instance Mentor Calibre PEX. The extracted circuit serves as input to another in-house tool called LiChEn [6-7], which is used for the electrical characterization of the standard-cells. The tool automatically detects all static states and transition arcs and characterizes internal and switching power, transition and propagation delay, input capacitance and also leakage power. Results are exported to the Liberty Format, which is widely accepted by industrial electronic design automation (EDA) tools for synthesis and power/timing analysis. The generated results are then evaluated to check if the cell follows the specified functional and electrical behaviors. Once these are verified, an abstract view is generated with an automated tool, like Cadence Abstract Editor, to produce models for place and route. A symbol view and a behavioral Verilog view are also generated for hierarchical design and digital simulation, respectively. After all these views are generated, the cell can integrate the library.

Using this design flow, a library of over five hundred components was designed for the STMicroelectronics 65nm CMOS technology. The library was already used in the design of three different network-on-chip routers and an RSA cryptographic core, all until the layout level. More details about this library and the tools that compose the ascend flow can be obtained in references [8-12].

III. Conclusion and Future Work

The ASCEnD flow is a response to the growing need for developments in techniques, tools and libraries for asynchronous design. It was successfully employed for the design of over five hundred components dedicated for asynchronous circuit design. Also, all in-house tools were designed by the authors during the development of the flow. Ongoing work includes the development of a new library compatible with the MOSIS service and silicon validation. Improvements on the ASCEnD flow, like supporting multi-output gates, are also expected results of ongoing work.

Acknowledgements

Authors acknowledge CNPq support under grants 55679/2009-6 and 310864/2011-9. Ney Calazans acknowledges FAPERGS support under grant 11/1445-0.

References

[1] H. Eriksson, P. Larsson-Edefors, T. Henriksson and C.Svensson, "Full-Custom vs. Standard-Cell Design Flow – an Adder Case Study," In: ASPDAC'03, pp. 507-510.

[2] J. M. Rabaey, A. Chandrakasan and B. Nikolic, "Digital Integrated Circuits a Design Perspective," Upper Saddle River: Pearson Education, 2003, 761p.

[3] A. J. Martin and M. Nyström, "Asynchronous Techniques for System-on-Chip Design," in: Proceedings of the IEEE, June 2006, 94(6), pp. 1089-1020.

[4] P. A. Beerel, R. O. Ozdag and M. Ferretti, "A Designer's Guide to Asynchronous VLSI," Cambridge University Press, 2010, 337 p.

[5] MOSIS Integrated Circuit Fabrication Service, "http://www.mosis.com".

[6] M. T. Moreira and N. L. V. Calazans, "Electrical Characterization of a C-Element with LiChEn," In: ICECS'12, pp. 583-585.

[7] M. T. Moreira and N. L. V. Calazans, "LiChEn: Automated Electrical Characterization of Asynchronous Standard Cell Libraries" In: EUROMICRO DSD 2013, 4p.

[8] M. T. Moreira, B. S. Oliveira, J. J. H. Pontes, N. L. V. Calazans, "A 65nm Standard Cell Set and Flow Dedicated to Automated Asynchronous Circuits Design," In: SOCC'11, pp. 99-104.

[9] M. T. Moreira, B. S. Oliveira, J. J. H. Pontes, F. G. Moraes, N. L. V. Calazans, "Adapting a C-Element Design Flow for Low Power," In: ICECS'11, pp. 45-48

[10] M. T. Moreira, B. S. Oliveira, F. G. Moraes and N. L. V. Calazans, "Impact of C-Elements in Asynchronous Circuits," In: ISQED'12, 4p.

[11] M. T. Moreira, C. H. M. Oliveira, R. C. Porto and N. L. V. Calazans, "Design of NCL Gates with the ASCEnD Flow," In: LASCAS'13, 4p.

[12] M. T. Moreira, B. S. Oliveira, F. Moraes and N. L. V. Calazans, "Charge Sharing Aware NCL Gates Design," In: DFT'13, 6p.

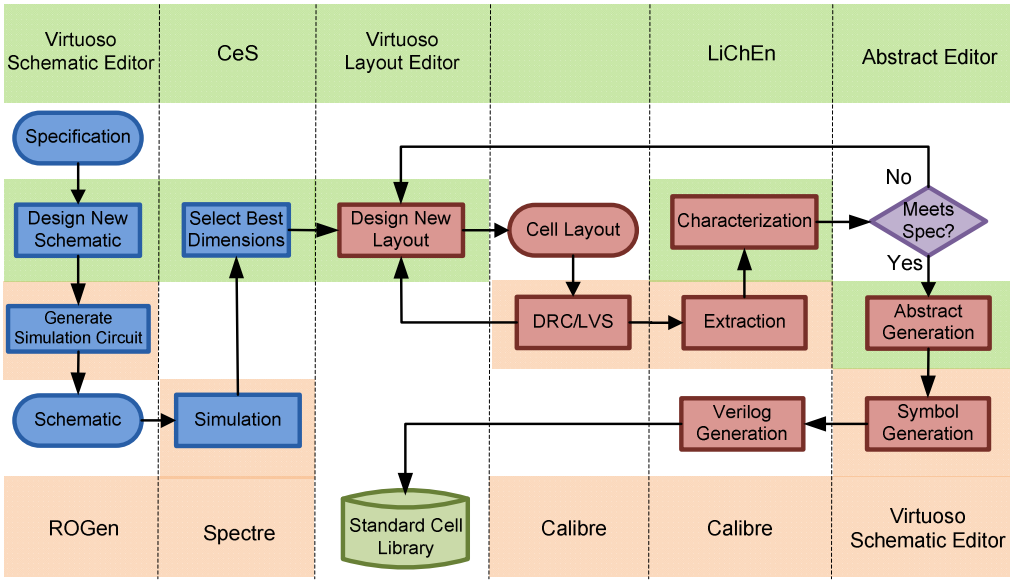

Fig. 2. The ASCEnD design flow. Boxes are actions, round boxes are descriptions, diamonds are decisions and the cylinder is the repository.