Generator Based Approach for Analog Circuit and Layout Design and Optimization

Achim Graupner
Zentrum Mikroelektronik Dresden AG
Dresden, Germany
achim.graupner@zmdi.com

Roland Jancke
Design Automation Division
Fraunhofer Institute Integrated Circuits
Dresden, Germany
roland.jancke@eas.iis.fraunhofer.de

Reimund Wittmann
IP GEN Rechte GmbH
Bochum, Germany
reimund.wittmann@ipgen.de

***Abstract*—Layout generation remains a critical bottleneck in analog circuit design. It is especially distracting when re-using an existing design for a similar specification or when transferring a working design to a new technology. This paper presents a new methodology for layout generation of analog circuits that is based on a modular circuit design and a so-called "executable design flow description". This is created once manually and allows to describe the layout in a technology independent and parameterizable manner assuring a consistent view of circuit and layout design. Complex layouts can be created in negligible time, achieving an early involvement of layout effects in the circuit design. Furthermore, the parameterization of the design description allows simplified technology transfer and seamless access to sizing tools.**

Keywords - circuit design, layout design, analog circuits, parametrizable design cells

I. INTRODUCTION

Automation of engineering steps in the area of integrated analog circuit design has been a challenging task over decades. Continuously increasing complexity of fabrication rules and design guidelines related to low-power, signal integrity, yield, reliability, and first-time-right demands for advances in this area. Systematic automation approaches have to face the wide spectrum of analog and mixed-signal design solutions. Even if the amount of devices in such a single solution covers only tens of devices, their detailed configuration may be sensitive to hundreds of specification parameters. The enormous complexity of the related solution space barricades semi-custom synthesis approaches known from digital circuit design.

The 2009 edition of the ITRS roadmap states: "An automated analog design process is still an open challenge in today's design environments. … The analog and mixed-signal flow requires tools for analog circuit synthesis and automated analog verification on all design levels. … Synthesis tools must concentrate on analog-specific layout synthesis and do not currently take into account all precision matching needs for such designs. ... Practical reuse of building blocks in a scaled version of a device is complicated due to missing verification automation." [1]

The reduction of effort is the primary motivation to apply automation principles within existing design flows. Traditionally, analog design automation is applied to reuse scenarios that allow compensating the initial design overhead by savings in expected re-use case. With the increasing complexity of design rules and physical effects in modern fabrication technologies, this tradition has to change. Today it makes sense to practice design automation also for individual engineering steps by assisting the designer in prioritizing mastering complex design tasks, navigating through decision trees and to consider complex cost functions all sensitive to specification nuances only.

Analog circuit design is dominated by top-down iteration loops. Without access to automation principles especially layout optimization loops can noticeable slow down the whole engineering process. First-time-right requires consideration of non-idealities caused by parasitic layout effects (e.g. parasitic devices, signal coupling, influence of lithography, electro-migration, antenna-rules) and layout optimization steps. In addressing the above requirements, and in the presence of ambitious project schedules there is a necessity to reduce the effort for analog circuit design automation, especially for analog layout tasks.

In the area of analog layout generation there is limited tool assistance or automation. There are no competitive tools for generation of analog layout available. Two problems have to be solved: placement of devices and routing. For the routing of analog circuits there is tool support available. Different technologies are used by various vendors (Cadence [2], Magma [3]). However, the target application of these routers is chip assembly routing, not block device routing. Significant effort has to be spent for defining constraints for the router in order to achieve feasible routing results. The problem of placement of analog devices is not truly solved yet. There is research work undertaken for placement of analog devices [4] which however is not yet suited for industrial application. An overview of the state of the art is given in [5]. For layout migration there is more commercial tool support available. The authors are aware of ALX from Magma [6], Osiris from IN2FAB [7] and Mentor's Chameleon Art [8]. The latter two tools are proprietary and available as design migration services only. In general these layout migration tools can only be used for accomplishing plain layout migration tasks. As such the results are only feasible if all device parameters are scaled similarly. If some device parameters change significantly often a completely new device placement is required. In such cases the migration tools yield sub-optimum results with regard to area.

978-3-9810801-7-9/DATE11/©2011 EDAA

To address the above issues in layout design efficiency, a generator based approach for layout generation in conjunction with a modular circuit design is proposed. The idea of the generator-based layout description is outlined in section two. The modular approach for the design of amplifiers is proposed in the third section. The fourth chapter presents a methodology to combine the generator-based approach with verification and optimization tools.

II. Executable Design-Flow Descriptions

Recently the Generic Engineering Model (GEM) approach addressing full-custom analog, RF and mixed-signal design has been recognized as a valuable contribution to design automation in the area of analog circuit design [9] [10]. Excellent results have been reported with respect to the increased engineering efficiency and re-use capability. An algorithmic design entry for analog circuit design, in addition to the established graphical one, offers the required flexibility to realize complex full custom designs. The main idea is to pay special attention to the IP engineering process itself instead of looking mainly to the result of it. Design parameters, PDK selections and even design frameworks can be exchanged easily during the IP engineering process. The professional IP development platform 1*Stone*®-Developer (IPGEN) acts as a complementary extension to available Mentor or Cadence based design frameworks (Fig. 1). It completes a future oriented design flow addressing design flexibility down to final layout by adding *symbolic design abstraction* and *synthesis-like design refinement* as mandatory design automation features.

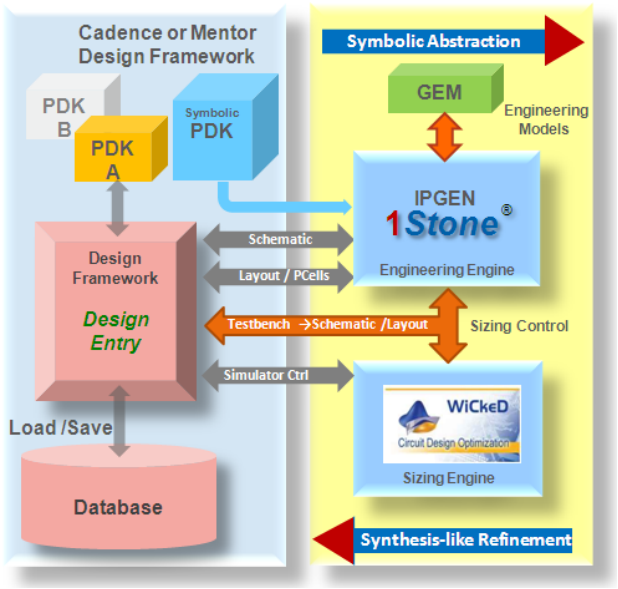

Figure 1. Design Flow Extension that enables analog IP Retargeting and Design Porting

Symbolic Abstraction - The capability of Generic Engineering Models covers the transforming of complex analog and mixed-signal designs into compact symbolic design descriptions, taking the complete engineering process into account. The complexity of analog layout, which has to consider a "galaxy" of design rules (process, DFY, quality, design constraints) is today's main re-use limiter for analog circuit design. The GEM approach especially transforms and abstracts the complex geometries on layout level into soft properties for a unified symbolic layout view. After a transformation operation the analog design description is optimized for a language based processing, similar to design flows available for digital design for many years. GEM enables reuse in a unique approach by providing an executable design flow and taking care of abstracted design guidelines.

Synthesis-like Design Refinement - Once the symbolic abstraction step has taken place, the related engineering and design know-how is stored in a compact circuit description which includes tool interaction and work flow history. For re-use purpose the abstraction process has to be reversed. The Synthesis-like Design Refinement step enables an individual constraint and fabrications process mapping. Since optimal IP sizing depends on the available process models as well as updated specifications, optimization and sizing usually does not follow given algorithms. The circuit optimizer WiCkeD [11] (MunEDA) enriches the refinement flow in a way that takes advantage of given test bench and described layout characteristics. WiCkeD takes care of sizing optimization steps, yield estimation/optimization and IP reliability in all specified operating conditions. Individual architecture redefinition and sizing optimization carried out by WiCkeD are back-annotated into the GEM description (schematic & layout). 1*Stone*® takes care of WiCkeD setup parameters since all IP details related to schematic, layout and test bench(es) are documented within the existing GEM description.

The GEM design approach can be followed without leaving or shortening the already qualified design flows and by interfacing the original PDKs from the process vendors. The graphical design entry stays available. 1*Stone*®-Developer enables the step-wise execution of structured design descriptions referring to a defined design flow. Similar to the manual design approach, the design result is stored in the database of the selected design framework. This automated design approach covers schematic/test bench creation, simulation, layout creation, optimization and design verification. The test bench view plays a dominant role within the executable design flow description: starting from the role of an executable specification, it monitors design progress, takes care about constraint processing and consistency across the defined abstraction levels and different design views. In addition it monitors sizing, secures reliability and minimizes power consumption and area. Finally it assists the designer in preparing the required design verification steps. Design abstraction in the GEM based design flow supports decoupling of design and process related constraints and therefore increases the flexibility to tailor the design towards alternative process nodes or updated application scenarios.

A specific, database independent engineering language comes along with 1*Stone*®-Developer which is able to control the whole design process from specification down to layout. Alternative language based approaches directly accessing low level design databases (e.g. Open Access) mostly have to struggle with the merge of process and design related constraints with an unclear history. The proposed symbolic design description clearly separates process and design related constraints, which are just soft properties completing the

configurable unified engineering description. The engineering process itself forms an editable 3rd dimension for the analog design description and documents the workflow history. The first two dimensions are represented by a snap-shot of the final design result only.

Tradeoff Transparency vs. Efficiency - The introduction of hierarchical design levels improves design automation comfort since it divides complex design tasks into smaller ones. It enables re-use of simple building blocks even if a re-use at macro level has not been considered so far. In the presented design methodology intelligent parameterizable 1*Stone*®-Device-Generators are configured directly from the related schematic or test bench (e.g. geometries, shielding, array setup & distribution, guard ring, path widths). Analog designers prefer operating on flat schematic structures, having access to all relevant information directly without a need for clicking through the hierarchy. A new type of symbol generator has been developed in order to overcome the tradeoff between optimized hierarchical design processing (relevant for efficiency) and circuit transparency (relevant for design supervision). It is able to generate schematic-like-symbols out of existing schematic databases. It allows reporting "flat" schematics composed of hierarchical subcells (see also the example in chapter III).

Individual design scenarios which consider re-use as an option but not as a must, are supported by specific features of 1*Stone*®-Developer**,** e.g. assisting reference for consideration of best practices for reliability and yield, consideration of shielding and matching techniques (statistical averaging, common centroid placement, dummy structure generation) layout hierarchy and device grouping mechanisms, intelligent parameterizable devices, abstraction of essential constraints, constraint propagation across symbol, schematic, layout and test bench views.

III. CIRCUIT DESIGN USING MODULAR AND PARAMETRIZABLE BUILDING BLOCKS

Amplifiers like operational amplifiers, transconductance amplifiers etc. form a circuit class often used in analog circuit design. They are used in a large number of applications resulting in a wide specification range e. g. for the required gain or bandwidth. In conventional analog design first an appropriate circuit topology is selected where the parameters of the single devices – mainly widths and lengths of the transistors – have to be tuned in an optimization step to meet the given specification. Afterwards, the layout is created in a separate design step. This work flow creates very application-specific implementations for each single amplifier and necessarily causes a lot of rework on the layout side – often up to a complete new layout – even if only single parameters like widths/lengths are changed during optimization. The same effort usually occurs when an existing layout shall be re-used with a different aspect ratio or in a different target technology.

Most of the commonly used amplifier topologies can be assembled using a limited number of structures like differential pairs, bias generation or output stages, in the following referred as circuit blocks. In Fig. 2 a fully differential operational amplifier with folded cascode and class AB output stage and its separation into circuit blocks are depicted: differential pair, folded cascode with output stage, and bias generation. These circuit blocks can be used for the assembly of other amplifier structures. They are implemented using the methodology described previously as GEM in a scalable and technology independent manner. As such the layout implementation phase of an amplifier can be shortened significantly because it becomes simply an assembly of the layout blocks.

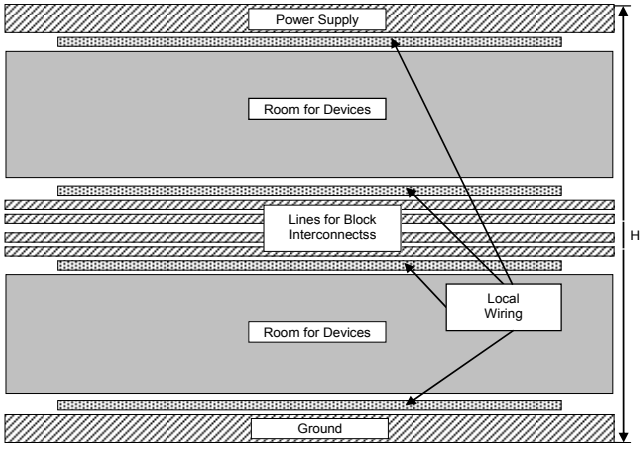

Figure 3. Street principle for layout floorplan

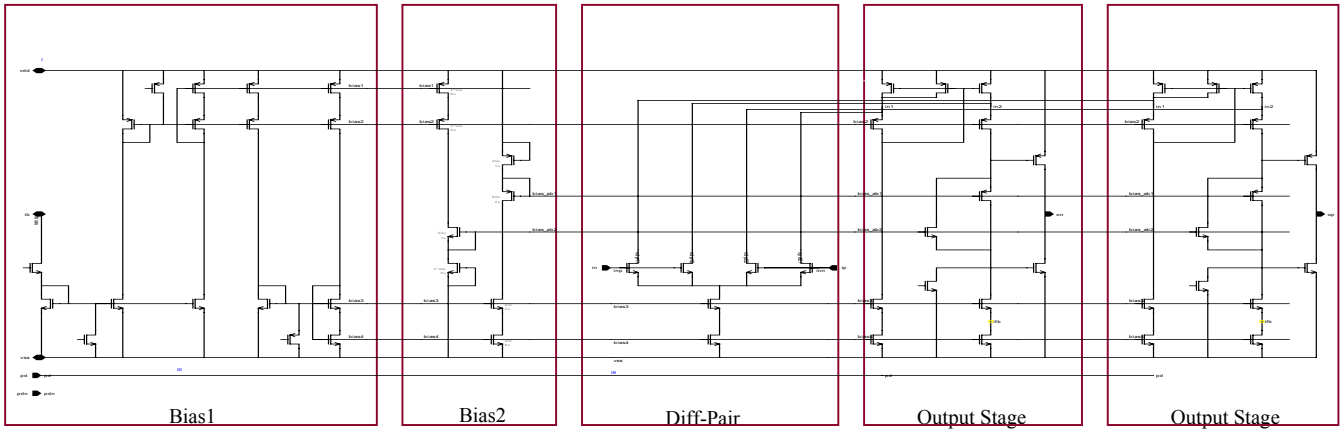

Figure 2. Folded cascode with AB output stage – subdivided in reusable circuit "brick" blocks (common-mode feedback and AC compensation not shown)

In order to achieve flexible scalability of the layout blocks, a street-like floor plan principle as shown in Fig. 3 is used. It is adapted from the structures of digital standard cells. The height is given as parameter and fix for all layout blocks. On top and bottom there is room for power supply and ground. In the middle there are lines to connect aligned blocks. In the remaining space there is room for the devices. The width of the respective blocks is determined by the device parameters.

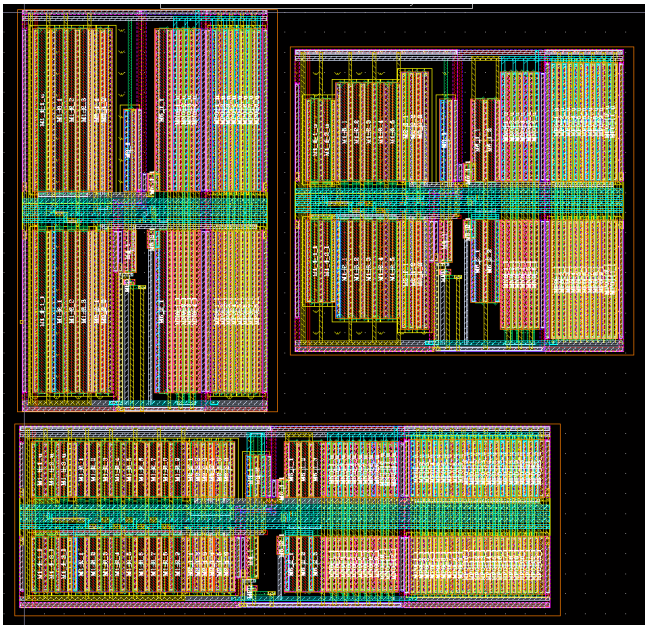

Figure 4. Layout examples with different cell heights of output stage

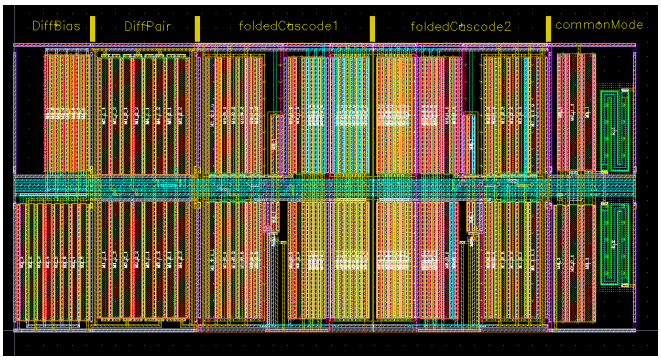

Figure 5. Layout example of amplifier

Fig. 4 shows some results of the generated layout for the output stage of Fig. 2. It can be seen that depending on given height and device parameters the width scales accordingly. It is noted again that the layout can be generated in any technology without any modification of the generator code. In order to achieve good matching, most devices are split in two or more devices and placed in the upper or lower device placement area. As such common centroid structures can be achieved.

In Fig. 5 a layout example of the amplifier of Fig. 2 is shown in a 0.35u technology. It has been assembled using the blocks denoted in Fig. 2: differential pair, output stage, and common mode feedback. The amplifier above is fully scalable and can be transferred into any technology with little effort.

IV. Design Optimization and Verification

The sketched design methodology using executable design descriptions is especially useful where design tasks are repeated with the same topology but different design parameters. This is the case in analog circuit design within a design optimization loop or if a proven design has to be transferred to a new technology.

The aim of design optimization is to adapt the design parameters of an analog circuit topology to ensure that a given specification is met. This has to be checked for the nominal case as well as under parametric variations from operation and manufacturing conditions. During this process of yield optimization, the design is centered within the parameter space by iterations of sizing, simulation, and yield estimation. Tools like WiCkeD (MunEDA [12]) and Neocircuit (now integrated into Cadence ADE) are available to support this design task.

The design optimization tools usually operate only on the schematic level of a circuit. Once the sizing is optimal with respect to this analysis, the layout of the circuit is prepared, usually done by a different designer. Unfortunately, the prepared layout again has influence on the circuit performance and the reachable yield, therefore it is desirable to include the layout preparation in the optimization loop. Some ideas have already been published to tackle this problem [13].

The proposed generator approach is best suited to address the task of including the layout into the optimization loop. This is possible since the current design parameters (i.e. w/l measures of each device) are fed into the parametrizable generator which automatically and instantly gives a correct and DCR clean layout for these parameters. The design rules were already taken into account when writing the generator code using relative distances between devices which work for a large parameter range. Nevertheless, extraction of layout parasitics is still a time consuming task which means that layout generation is not possible in each turn of the iteration loop. More efficient methods are required in this field.

A second very promising application area for the new generation methodology is technology migration. This usually means that a working design or even a whole library of basic circuit cells has to be adapted to a new technology. This could be either due to shrinking to a smaller technology node or when changing the technology provider using the same technology node. In general, during the migration the circuit topology remains the same but a different sizing of the devices is required to meet the specification in terms of the changed technological conditions. Moreover, each new technology features different design rules with dedicated minimum distances and others.

The generator approach solves the migration problem in a very smart manner. If the generator is written in a suitable way it does not contain absolute values for the distances between devices. Instead, variables are utilized that receive their value from a technology setup. Once the setup for the new technology is available, all generators produce automatically

correct layouts with respect to this setup, moreover, the generator itself is technology independent and instantiates devices from the actual technology environment. This means in the new environment the respective devices are used automatically, thus, a first version of the design that can be directly simulated in the new technology is immediately available.

After the circuit topology is migrated, the design parameters have to be adjusted, i.e. an adjusted sizing has to be found by optimization. Here again the generator approach can help as mentioned before. During the optimization process sizing tools like WiCkeD need design constraints that ensure successful search for an optimum, in a proper subspace of the parameter space. Many of the parameter restrictions are so-called structural constraints which guarantee correct working area for the devices (e.g. linear or saturation region of a transistor). Since these structural constraints depend only on the circuit topology, they again can be generated, thus representing an additional constraints view for a design.

In order to start the optimization process a number of test benches have to be prepared that are used to check the actual circuit performances against the given specification. All the different tasks that are related to design verification, like generation, configuration, and execution of test benches are in principle, technology independent and are suited for automation. Generation of test benches enables fast setup of design verification in a different technology environment. This once more fits well into the described generator approach and is supported by the 1*Stone*® tool suite.

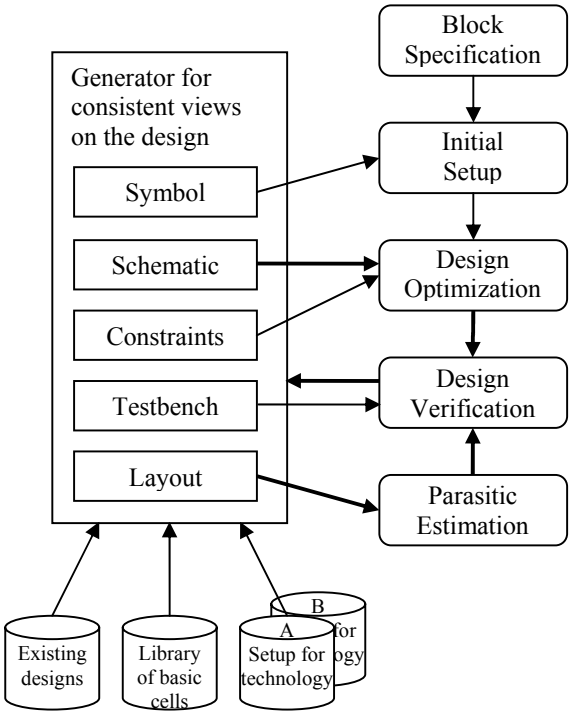

Figure 6. Design flow using the generator approach

A major issue of the proposed methodology is the necessity to manually write generator code for the different design views (circuit, layout, test bench) this task is drastically simplified by using the capability to import existing circuit designs. In addition, improved methods for hierarchical design using "standardized" macro cells as basic re-usable building blocks are being investigated as described in chapter III.

A flow is sketched in Fig. 6 where these elements are integrated into a powerful methodology for efficient analog circuit design. Central element is the generator that produces consistent views for the different design aspects. It uses technological information as well as basic cells and designs from existing libraries. The generator is part of an optimization loop that adjusts design parameters, verifies the circuit performances, and includes information from the parasitic estimation.

The described methodology was tested using the design of an OTA which was available in a 150nm technology (see Fig. 7). The task was to re-use this design for a different application in a 350nm technology from a different foundry.

Since both technologies were already prepared for the 1*Stone*® tool suite, mapping of the schematic and the initial setup of the design in the new technology was a matter of minutes. This included all test benches that were necessary to verify the circuit characteristics in the previous technology. After the draft design was ready for simulation in the design environment, the setup for the optimization tools was prepared. It took some effort to set all constraints, design targets, and operating conditions since circuit optimization was done by hand previously. As a result of the optimization process it turned out that the initial specification of the OTA could not be reached with a sufficient yield in the larger technology. Therefore a slight design modification was necessary. Running the optimization again we achieved all the specified performance values including DC gain, transfer frequency, slew rate, gain and phase margin with a yield of 99.9%. For this verification a number of corner case and Monte-Carlo analyses were performed by the optimization tool. The resulting layout is shown in Fig. 8.

Looking at the entire technology migration process for this example circuit, the overall effort for the designer from the first raw circuit mapping until final yield estimation and layout generation took approximately 2 days (except batch task that are running automatically). The time to do all the migration work by hand without generator support would take in the order of two weeks, according an experienced designer. That means that the generator saves nearly 80% of the effort that is necessary for design migration from one technology to another. Thinking about the migration of complete libraries of analog and mixed-signal IP the enormous savings become obvious.

Part of the migration for this example has still been done by hand because interfaces were not yet available. Here we expect even more efficiency in the future. Specification and implementation of these interfaces is part of the ongoing work in cooperation with the respective tool vendors.

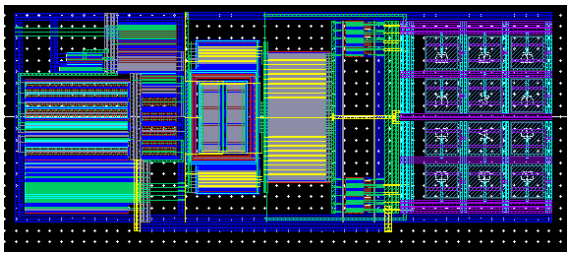

Figure 7. Layout of the OTA example in the original 150nm process

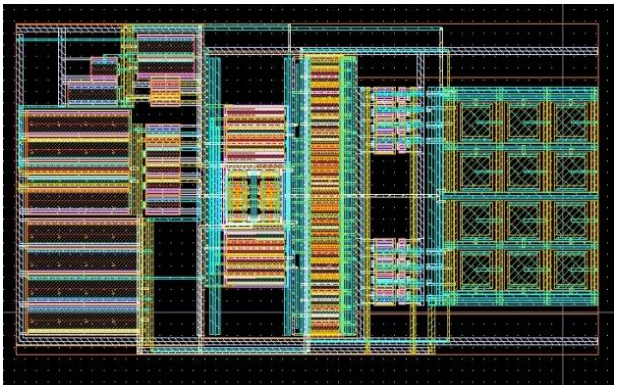

Figure 8. Layout of the OTA example in the target 350nm process

V. CONCLUSIONS

The presented generator approach allows generating many different but inherent consistent views on the design of a circuit. It allows exploiting a modular and hierarchically structured design for the generation of a regular and dense layout. The approach allows a highly efficient design methodology where many tedious and error prone tasks of design verification, optimization, and technology migration are automated.

ACKNOWLEDGMENT

This work is supported by the German Ministry of Education and Research BMBF under grant number 01M3086. The authors would like to thank their colleagues at Fraunhofer IIS for providing the example circuit as well as Karl-Heinz Rooch (Fraunhofer IIS/EAS), Harald Bothe (IP GEN), and Stefan Jäkel (ZMDI) for their contributions.

REFERENCES

[1] International Technology Roadmap for Semiconductors, 2009 Edition

[2] www.cadence.com/rl/Resources/datasheets/space_based_router.pdf

[3] www.magma-da.com/products-solutions/chipfinishing/titanrouter.aspx

[4] M. Eick, M. Strasser, K. Lu, U. Schlichtmann, and H. Graeb, Comprehensive Generation of Hierarchical Placement Rules for Analog Integrated Circuits, IEEE Transactions on Computer-Aided Design on Integrated Circuits and Systems, Dec 2010.

[5] H. Gräb (ed.), Analog Layout Synthesis - A survey of Topological Approaches, Springer-Verlag, New York, 2010.

[6] www.magma-da.com/uploadedFiles/products-solutions/Analog_Mixed_Signal/Titan-ALX%20Datasheet.pdf

[7] www.in2fab.com/tech_services.html

[8] S. Hammouda, H. Said, M. Dessouky, M. Tawfik, Q. Nguyen, W. Badawy, H. Abbas, and H. Shahein, "Chameleon ART: a non-optimization based analog design migration framework", 43rd ACM/IEEE Design Automation Conference, pages: 885 – 888, 2006

[9] R. Wittmann, N. Nandra, J. Kunkel, M. Vanzi, J. Franca, H.-J. Wassener, Ch. Münker, "Life begins at 65 – Unless you are mixed-signal?", Proceedings DATE 07, Nice, 16-20 April 2007, pp. 936-941, IEEE Computer Society, ISBN 978-3-9810801-2-4

[10] M. Kosakowski, R. Wittmann, W. Schardein, H.-J. Jentschel, "Yield prediction and optimization to gain accurate devices for analog design in nonideal nanoscale processes", 10th International Workshop on Symbolic and Numerical Methods, Modeling and Applications to Circuit Design (SM²ACD '08), Erfurt, Oct. 2008

[11] M. Pronath. "Design for Yield Concepts for Analogue and Mixed-Signal Circuits", MEDEA+ Design Automation Conference, May 30th - June 1st, 2006, Prien, Germany

[12] www.muneda.com

[13] R. Castro-Lopez, O. Guerra, E. Roca, and F. Fernandez. An integrated layout-synthesis approach for analog ICs. In IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, volume 27, pages 1179–1189, 2008.