Mixed-Signal Design in Chip Development

Ingo Kunkel
Institute of Computer Engineering
University of Heidelberg, Ruperto Carola
Computer Architecture Group
Heidelberg, Germany 691117
Email: i.kunkel@stud.uni-heidelberg.de

***Abstract*—This paper summarizes the special requirements and important aspects for mixed-signal design. After introducing some background informations, three different design methodologies are described. A brief overview of actual development environments, commercial and open source are given. The important aspect of modeling analog modules in mixed-signal design is covered with a practical demonstration.**

I. Introduction

The requirements in chip development of integrated circuits changed since the first computational support tools in early 80's. First, self-developed tools supported the engineers with simulations in the command line to reduce the consumption of Programmable Array Logic (PAL) [1] chips in development. In the mid 80's the first Hardware Description Language (HDL) [2], Verilog [3], and a little later VDHL were introduced and changed the way of chip development significantly. In the early 90s, first simulators for HDL were developed. The ability to simulate and verify the chip design in a reasonable time reduced the development time. At the same time, advanced manufacturing processes were invinted [4]. Chips became more complex while the power consumption decreased. In the last decade, System on a Chip (SoCs) became more and more important. A SoC is the integration of an amount of functions that were previously processed by two or more chips. These complex chips requires a good cooperation between analog and digital chip engineers to realize the complex chips in a shorter time. The mixed-signal approach combines the analog and digital work flow to reduce costs and verification time in development.
This paper is structured as follows. The second section introduces basic about analog and digital circuits. In the next section, the different design approaches in chip development are described. The fourth section gives a brief overview of commercial and open source Electronic Design Automation (EDA) tools. Section five contains a practical example with a comparison of a real and modeled output-stage with the Cadence EDA tools. The paper ends with a summary and conclusion.

II. Basics

A. What is mixed-signal design?

Mixed-signal design or analog-mixed-signal design (AMS-Design) describes a combined design work flow for analog and digital circuits in one chip to develop integrated circuits (IC) like microprocessors, op-amps, or SoCs.

1) Digital circuits: In digital circuits, signals are discrete: low, high, or high impedance. The circuits consist of integrated MOSFETs to build logic gates and data storage. Examples for digital circuits are microprocessors, FPGAs, and memories. The simulation is event-driven and therefore faster than real time simulations. Typically, a library contains gate modules such as "and", "or" etc. The development is on a higher abstraction level - the Register Transfer Level (RTL). The basic gate modules are combined into functional modules such as counters, FIFOs or adders . These functional modules are verified and also stored in the library for further use in higher functional modules e.g. an aritmetic logic unit (ALU).

2) Analog circuits: The analog signal contains information using non-quantized variations in frequency and amplitude. Analog circuits consist of linear components such as capacitors, inductors, and resistors [5]. Examples for analog circuits are op-amps, active filters, or phase lock loops (PLL). The simulations depends on the interaction between components and mathematic equations. Consequently, the simulation time increases and is approximately 1000 times slower than digital simulations. Usually, the analog libraries contain the spectre modules like resistors or transistor and the development is on a low-level.

3) Problems with analog libraries and analog intellectual property (IP): The usage of libraries in digital circuits allows the easy reuse of already verified modules. It is possible to change some aspects like data width with parameters. The reuse of analog modules is not as simple. Each transistor or resistor in the circuit influences the behavior. When changing one value, it is not guaranteed that the system behaves as specified. As a result, it is impossible to parameterize an analog circuit. Furthermore, the behavior depends on the production technology.
In addition, the use of analog IP is complicated und causes a the high verification time. Nothing is guaranteed to work as described, so there has to be an additional verification as well.

B. Why is mixed-signal design important today

The number of interfaces between analog and digital modules on one chip has increased. A fault in one can lead to a re-spin of the chip which causes a higher development time and additional costs. In a re-spin the found errors have to be fixed and the chip has to be produced again. Because of the expensive mask costs in manufacturing, this is expensiv. Furthermore, it is important to have well specified simulation environments on different levels of the design. This is necessary because of the high time consumption of

analog simulations. With mixed-signal design, the modeling of analog circuits for higher-level simulations reduces the simulation time with an adequate accuracy.

The advantages of AMS-design can be summarized as:
- reduced verification time
- faster time to market
- avoided re-spin

However, some challenges are still there. By today's standard power-saving function the verification effort increases because different power-states must be simulated.

C. Computation support in chip development

Nowadays, chips cannot be development without the support of computers. The EDA tool chain supports the engineer in every necessary step. Not even thirty years ago, powerful HDLs where invented and developed further over the years. Design environments simulate the behavior and support the engineer in the verification process. Placement and routing, clock and reset tree synthesis, parasitic influences calculation, time extraction and much more is done automatically. The results are then reinserted into the first steps to get an accurate simulation of the behavior of the whole chip. By creating real number models of analog circuits, the high-level simulation time is reduced while an adequate accuracy is maintained.

D. Work-flow before the Mixed Signal approach

For historical reasons, digital and analog circuits were made by specialized teams with different workflows, sometime even on different chips do make it easier. In SoCs with analog modules, this is not possible. From the system specification, the interfaces between analog and digital parts are defined and both parts get their own sub-specification. The teams work on their specific parts and integrate them together when finished. This approach works, but it is error-prone and can throw back the project because of interrupts in the interfaces or issues in intra-team communication.

III. Design Methodologies

A. Top-down approach

The top-down approach starts at a high level of abstraction and specifications. Each module is specified and then split in sub-modules which are specified again. This process is repeated until the smallest part is specified and can be implemented and verified. Afterwards, the next higher level will be verified and so on. When a module does not met the specification, it can directly be modified and verified again. Large projects are split into sub-projects.
The top-down approach is typically used for the development of digital circuits.

B. Bottom-up approach

The bottom-up approach is the other way around, therefore it starts with the implementation ant verification of the small parts. Afterwards, the parts will be combined to larger parts and so on. Analog circuits are usually developed this way, because the electrical-elements are defined and must be combined in the right way. The base circuit already exists and is modified until it fits the specifications. When the highest level is reached, the module possibly does not match the specification. In this case, a redesign at lower levels is necessary, which leads to high development costs.

C. Analog-centric mixed-signal flow

The analog-centric design methodology is for analog projects with a small part of digital modules. The schematic editor is the tool of choice. The analog designer creates the layout, builds test benches, runs simulations and optimizes the circuit, all done in the schematic editor. Because the entire design process is in the schematic editor, the few digital parts are implemented at the transistor level, along with the analog part. This "custom digital" approach is used when no standard cells are available [6]. For larger digital blocks, the use of standard cells has the advantages of faster simulation, automatic placement and routing and the definition of constraints.

D. Digital-centric mixed-signal flow

For projects with digital focus and few analog modules, the digital-centric methodology is used. The digital design starts from a high level of abstraction with descriptions written in RTL, for example. The synthesis automatically creates the physical implementation using standard cells [4], [6]. The so called netlist contains informations about placement and routing, timing, and much more. Most informations are stored in the standard cell libraries. Therefore, the user does not have to know background details about the library in use. The netlist is typically written in Verilog or VHDL.
One important element in digital development is the static timing analysis (STA) [6]. With STA, the timing behavior and constrains between the elements were verified and also the netlist uses these timings. When analog modules or IP should be used, there are some difficulties. The timing behavior and communication with the analog part need high verification work. It is important to take a look at the parasitic influences and provide the synthesis with all these informations. One approach to reduce the verification work and time is the use of analog real number models (RNM). Two examples and benefits of RNM are described in [7]. The model works with analog electric quantities such as current, voltage, capacities, and resistors. Nonetheless, there are some critical timing paths which have to be simulated with a high accuracy analog simulation to guarantee the functionality. Another important topic is the demand for radio frequency (RF) analog modules. The wires inside and outside the chip transferring the RF - signals often require the same impedance because of differential transmission. Sometimes, a so called "coaxial shielding" at the layer above and below the signal is necessary. All these conditions must be met in the design process. In the last step of physical implementation, it is common to optimize small things for the manufacturing process or to improve the yield. Changes of the Analog IP can cause malfunctions, so they must be verified as well.

E. Unified, concurrent approach

The concurrent mixed-signal design methodology unifies the analog and digital-centric methodologies. In order to make this approach work, an effective abstraction of the analog

functions is inevitable. Therefore, it is possible to integrate and verify the analog modules in top-down manner. To accelerate the verification, the use of RNM is recommended [6]. The central role in this approach is a database realized in Open Access. All analog and digital objects, floorplan, netlists, as well as constraints for each object are stored in this database. The designer knows in which context his block should work and can look into the neighbor modules, because everything is stored in the database. The design intent is clearly defined by constraints throughout all design stages. The verification ensures that the constraints are applied properly and the specification is met.
The initial floorplaning is an important stage in design. IP and self-made blocks, which were defined in the system specification, are combined together. After that, some trade-offs have to be made in order to improve chip area, power and performance. To resolve these trade-offs a close cooperation between the system, analog and digital designer during the block implementation and integration phase is required.
In order to make this approach work, new challenges in teamwork and cross training among design team members come up. This is a great opportunity to learn more about the other domain, "read the other code" and become a mixed-signal designer.

IV. Brief Overview of Development Environments

The development environments for integrated circuits are called Electronic Design Automation (EDA). On the market, there are commercial EDA tool suites and some open source tools. The overview in [8] lists all known EDA vendors and describes some of them in more detail. This paper take a looks at some of them. in the ADMIRE Project,Reynaert et al. explained what is important for an mixed-signal EDA [9].

A. Well known commercial development environments

1) Synopsys, Cadence: Synopsys and Cadence are the two largest vendors for commercial EDA tool suites. Both offer a highly integrated toolchain for mixed-signal design. They support the complete workflow for the three methodologies and gives access to their mixed-signal IP portfolio. It is also possible to use third-party tools in the chain for special tasks. [8], [10]–[14]

2) Others Vendors: Vendors like Zuken, Mentor Graphics, or Dolphin Integration also sell integrated EDA tool suites. Often these are compatible to Open Access and the different file formats. Typically, one tool out of the chain is better and will be used for special tasks like analog simulation or has a better usability like ModelSim. [8], [10], [15]

B. open source

The open source tools are significantly less integrated than the commercial EDA - suites. Yet, they provide free tool for the development of integrated circuits and gives everyone who want the ability to develop integrated circuits. Furthermore, the usage of IP and the newest manufacturing process could be difficult because of the integration in the tools.

1) Alliances: Alliances is a tool suite developed by the department for SoC at the Pierre et Marie Curie University in Paris. It focuses on the development of multiprocessor SoCs. The tools assist the engineer in design, modeling, simulation, and verification of mixed and heterogeneous circuits. Furthermore, some tips for architecture and methods are provided. [16]

2) gEDA: The gEDA project was started in 1998, because of the lack of free EDA tools. At first, there was a schematic capture program and a netlister (gEDA/gaf). Over time, other programs like PCB, ngspice, gwave, GnuCap, Icarus Verilog or GTKwave were developed or joined the gEDA project. [17], [18]

3) Open Circuit Design: Open Circuit Design is a collection of open source tools like gEDA. The tools are e.g. Magic, Netgen, Qrouter or QFlow. Some of them have a long history. The development of Magic for example starts in the 1980's. [19]

V. Practical Example – AMS behavioral modeling

A. Modeling

As already mentioned, the simulation time with analog simulators on transistor level is time-intensive. To apply the top-down approach and simulate at a higher level of abstraction, a simulation using implemented analog modules has to run faster. Therefore, there are two possible solutions: AMS behavioral modeling and real number modeling RNM [20]. Which technique is applied for verification depends on the module complexity, the required speed, and the necessary accuracy. For demonstration purpose an AMS behavioral model was used.

1) AMS behavioral modeling: In a behavioral modeling, a simulator solves analog differential algebraic equations which are written in VHDL-AMS, Verilog-A, or Verilog-AMS [20]. The calculation expenses are reduced compared to the transistor-level description because fewer calculations are necessary. The degree of simplification determines the speed of the simulation of the model. It is a compromise between accuracy and simulation speed. The possible speed-up is one or two orders of magnitude.

2) Real Number Modeling: RNM is even closer to the digital simulation because it is event-driven. It uses floating point numbers and is at a higher level of abstraction than AMS behavioral modeling. The speed-up could be several orders of magnitude [20]. On the other hand, RNM has some limitations. First, it can only manage voltage or current. Interdependence between voltage and current cannot be modeled, therefore one of them must be fixed. Sabiro describes in [20] further information about the possibilities and limitations of RNM and compares both approaches.

B. Differential Output Stage

The differential output stage is a converter between the digital and the analog environment. On the input side, a digital signal is read while voltage and current are output. Some key characteristics of an output stage are the output voltage levels, slew rates, and power consumption. Slew rates defines the time

to achieve 90% of the output voltage level (high) respectively 10% (low). For power limitations and heat development, the power consumption is relevant. Furthermore, two important indicators for an output stage are the power supply rejection ratio (PSRR) and common-mode rejection ratio (CMRR). The PSRR is the ratio of change in the supply voltage to the produced output voltage [21]. It is an indicator how much the power supply noise affect the output.
The CMRR is the ratio of the differential and common-mode gain [22]. If both input voltage change in the same amount, the output voltage is supposed to stay the same. PSRR and CMRR are usually represented in logarithmic scale.
To measure all these parameters, three types of simulations are necessary. The transient simulation shows the behavior over time. The voltage levels, slew rates, and power consumption can be simulated with it. For the calculation of the PSRR and CMRR, the values are taken from the DC and AC simulation. The DC simulation measures the output voltage with a fixed input voltage, while the AC simulation capture the output resulting from changes in the input frequency over a given spectrum.

1) real analog output stage : In the real output stage the output voltage level depends on the supply voltage and the count of elements in the output path (transistors, resistors). The power consumption depends on the amount on parallel output transistors. Each parallel path leads to a higher power consumption. More than one parallel path is used to get a higher current at the output. The power consumption is also affected by the manufacturing technology. In reality, the current is limited, slew rates depend on the maximum current and the capacitive load on the outputs. For the practical example a full developed and verified output stage was used.

2) output stage AMS model: The workflow to create an AMS behavioral model is closely related to analog circuit design [6]. An analog circuit is designed out of common physical elements. The model is constructed out of dependent sources with functional relationships between their input and output voltage or current. Furthermore, internal variables are available and the outputs can be described depending on them. For the demonstration a properly working behavior model was given. It had to be adjusted in some places in order to use it in the same test bench. At first the digital filter had to be removed because the real output - stage didn't have one. Secondly, the input signal had to be transformed from digital to electrical to facilitate the use of the same test bench for both output stages. Therefore, the new electrical input signal had to be converted internally back to a digital value for further use. When the voltage value exceeds a threshold, the local variable is set to high or low. The sample code for one input is shown in listing 1. The *above* function is true when the value is greater than zero. The *always block* means that this code will be execute all time as a part of the digital calculations.

```
always  @(above(V(IN_P)-V_HI+0.1))
        begin
                InP_logic <= 1'b1;
        end
```

Listing 1. sample the electrical signal

The third change deals with the output voltage levels. Original, they were modified by the digital filter depended on the supply voltage. For this simplification the output level will set by parameters independent from the supply voltage. Listing 2 shows the assignment to the expected voltage is shown.

```
case (InP_logic)
        1'b1:  begin
         VoutP=SW_HI; [...]
        end
        1'b0: begin
         VoutP=SW_LO;[...]
    endcase
```

Listing 2. set of expected positive output voltage

The analog sequence is shown in listing 3. It starts with *analog begin* to tell the simulator that the following operations are analog. The variable *RoutP* is for the changes of the output resistant. This is done logarithmically, so the value changes quickly, for example when the high impedance state (10MΩ) should be taken. The next statement, Ohm's law, assign the output current. The transition function realizes the expected slew rate by ramping up the voltage in the needed time. The parameters Td and Tr determine the delay, rise, and fall time. *OUT_P* is from type electrical and with *I(OUT_P)* and *V(OUT_P)* the current and voltage values are addressed.

```
analog begin
  // Rout shifts on log scale
  RoutP = exp(transition(RlogP,
                         0,
                         TrRP,
                         TrRP));
  I(OUT_P) <+
        (V(OUT_P) - transition(VoutP,
                               TdVP,
                               TrVP,
                               TrVP)
        )/RoutP; // linear Vout chg

end
```

Listing 3. apply Vout and Rout with transition

C. Simulations

1) Simulations - real output stage: The graphical output for the transient simulation is shown in figure 1. In the graph, the saturation effects are clearly visible. The top graph is the output voltage at the output stage. The bottom graph is at the wire termination with ground as its reference. The capacity load in this simulation is 300fF, the rise time is 26.67ps and the maximum current is 38mA. In figure 2 and 3 the PSRR and CMRR calculations are shown. The green and red lines are without load while the light blue and yellow lines are with load. When the capacity load changed to 600fF, the rise time was 36.7ps. This time is longer because of the current limitations in a real output stage. To simulate 4μs the simulation run time was 909.5s .

2) Simulations - AMS model: Figure 4 shows the transient simulation for the model. There is no saturation effect. To achieve the same voltage levels and slew rates, the parameters where adjusted in a few simulations. The PSRR und CMRR are shown in figure 5 and 6 (kdB). The CMRR is constant

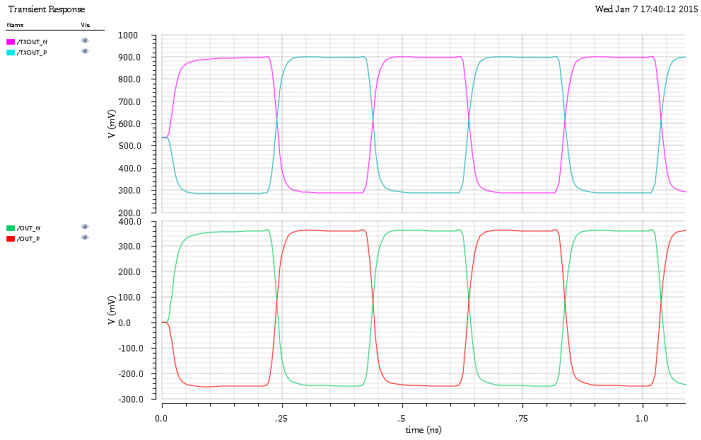

Fig. 1. transient response real output stage

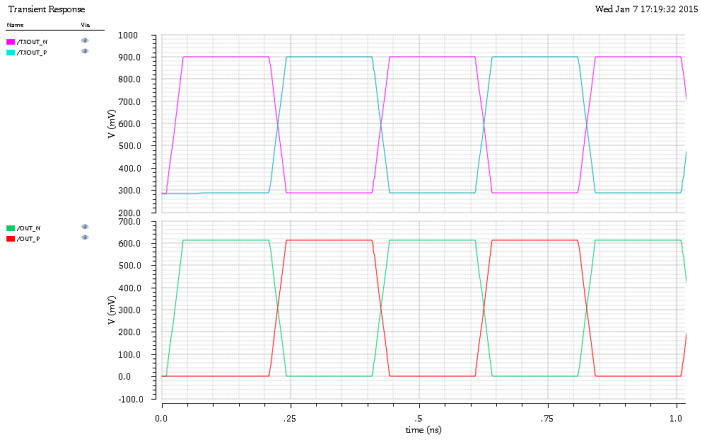

Fig. 4. transient response model output stage

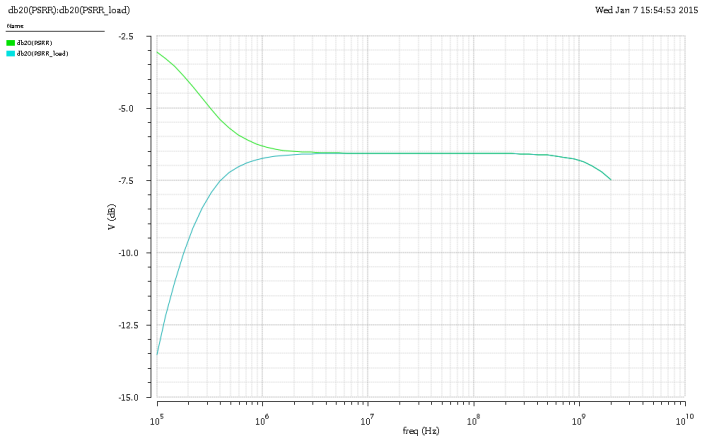

Fig. 2. PSRR real output stage

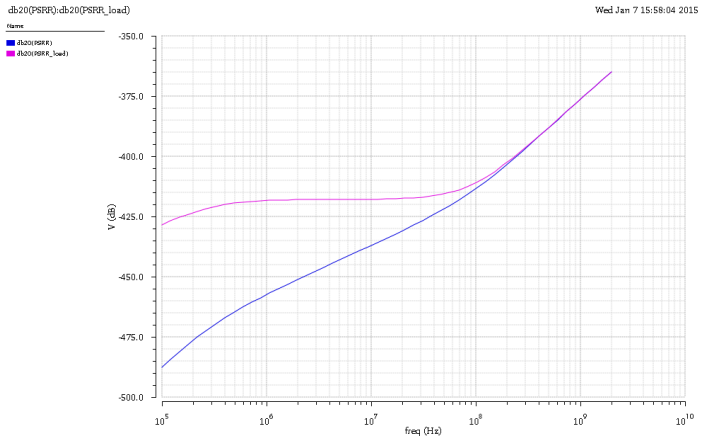

Fig. 5. PSRR model output stage

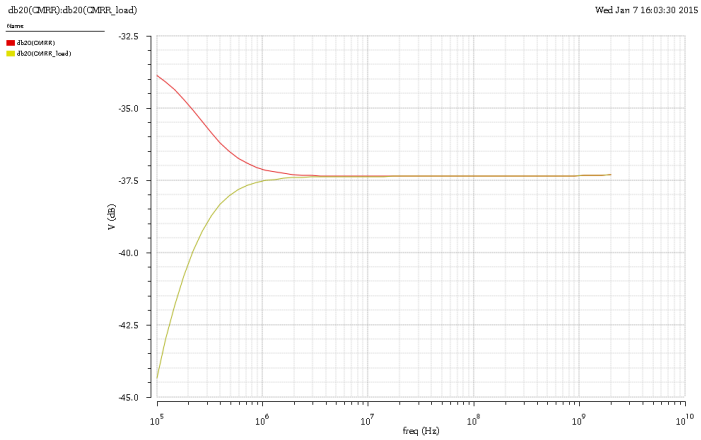

Fig. 3. CMRR real output stage

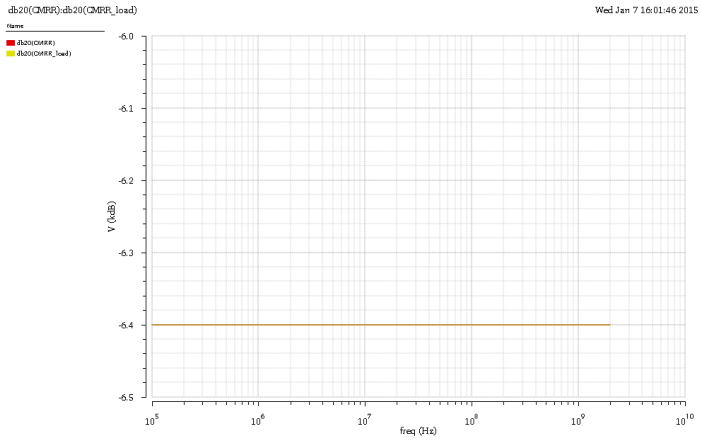

Fig. 6. CMRR model real output stage

at -6.4 kdB because input voltage is internally converted into digital signals and the parameters for the output voltage levels are used. The rise time is 26.57ps and the maximum current is 12.46mA. When the capacity load changed to 600fF the rise time stay the same. To simulate 4μs the simulation run time was 19.7s.

D. Comparing the results

Both output stages have the same voltage levels and rise times for a fixed capacity load. The difference dominate because the model is a strong simplification of the real output stage.
In the following points the output stages differ from each other:

- the model has no saturation effect
- the model has no current consumption and limitation
- the model output voltage levels are independent from the supply and input voltage levels
- calculation time for simulation 909.5s vs. 19.7s
- the PSRR and CMRR of the model are significant lower

By these facts, some simulation results can be explained. Because of the missing current limitation, the rise time is not affected if the capacity load changes. Furthermore, due to the independent output voltage levels, the model has no current consumption and the model CMRR is constant.
An important outcome is the speed-up. The model calculation for a 4 μs transient simulation is about 46 times faster.
For the real use of this model in development, it should be extended to achieve a more accurate simulation behavior. Of course, with a more accurate model, the speed-up decreases because of the higher complexity.
For example, to limit current, another transition in the analog block should be inserted. This limits the current similar to a current-limiting diode.

VI. Summary and Conclusion

The process of computational support in chip development in the past thirty years has changed dramatically. Tools, methodologies, and manufacturing technologies were further developed at a high pace. The requirements to develop ICs have increased. The pressure on companies to introduce products faster than their competitors leads to sophisticated team processes and verification optimizations such as modeling.
The demonstration has shown that the AMS modeling has potential of accelerating the verification process. However, the accuracy of the model and the speed of the simulation have to be balanced. The trend goes towards SoCs, which is only possible using mixed-signal design. In the abstraction of analog modules, RNM will be applied more often because the speed-up compared to AMS modeling is significantly higher.
Therefore, I think mixed-signal design with a unified workflow will be focused more in the next years.

References

[1] [Online]. Available: http://en.wikipedia.org/wiki/Programmable_Array_Logic

[2] [Online]. Available: http://en.wikipedia.org/wiki/Hardware_description_language

[3] [Online]. Available: http://en.wikipedia.org/wiki/Verilog

[4] V. Grimblatt, "Synthesis - state of art," in *Devices, Circuits and Systems, Proceedings of the 6th International Caribbean Conference on*, April 2006, pp. 327–332.

[5] [Online]. Available: http://en.wikipedia.org/wiki/Analog_signal

[6] J. Chen, M. Henrie, M. Mar, and M. Nizic, *Mixed-Signal Methodology Guide*. Cadence Design Systems, Incorporated, 2012. [Online]. Available: https://books.google.de/books?id=RKrfAwAAQBAJ

[7] "Soc verification, solving todays problem of soc verification," *eetasia*, 2010. [Online]. Available: http://www.eetasia.com/STATIC/PDF/201001/EEOL_2010JAN22_EDA_TA_01.pdf

[8] [Online]. Available: http://en.wikipedia.org/wiki/Comparison_of_EDA_software

[9] P. Reynaert, L. Callewaert, G. Gielen, G. Debyser, K. Lampaert, F. Leyn, G. van der Plas, W. Sansen, B. Schneider, R. Bloch, D. Orton, S. Boardman, J. Stent, J. Agaesse, J. Fischer-Binder, J. Riihiaho, and K. Tukkiniemi, "Admire: advanced mixed signal design environment," in *Electronics, Circuits, and Systems, 1996. ICECS '96., Proceedings of the Third IEEE International Conference on*, vol. 1, Oct 1996, pp. 428–431 vol.1.

[10] [Online]. Available: http://en.wikipedia.org/wiki/Electronic_design_automation

[11] (2012, Janurary) Custom and mixed-signal design solution unified solution for custom and cell-based design and verification. Synopsys. [Online]. Available: http://www.synopsys.com/Tools/Implementation/Documents/custom-design-solution-overview.pdf

[12] M. Nizic. (2011) Concurrent, oa-based mixed-signal implementation. cadence. [Online]. Available: https://www.si2.org/events_dir/2012/fall12/6-1_Cadence_final.pdf

[13] [Online]. Available: http://en.wikipedia.org/wiki/Cadence_Design_Systems

[14] [Online]. Available: http://en.wikipedia.org/wiki/Synopsys

[15] [Online]. Available: http://en.wikipedia.org/wiki/Mentor_Graphics

[16] [Online]. Available: http://www-soc.lip6.fr/en/

[17] [Online]. Available: http://en.wikipedia.org/wiki/GEDA

[18] [Online]. Available: http://wiki.geda-project.org/geda:faq

[19] [Online]. Available: http://opencircuitdesign.com/

[20] S. G. Sabiro, "Event-driven (rn) modeling for ams circuits," in *Specification Design Languages (FDL), 2013 Forum on*, Sept 2013, pp. 1–8.

[21] [Online]. Available: http://en.wikipedia.org/wiki/Power_supply_rejection_ratio

[22] [Online]. Available: http://en.wikipedia.org/wiki/Common-mode_rejection_ratio