Overcoming Roadblocks in Widespread Use of Hardware Description Languages for RF/Analog Design

Steven Bibyk and Edward Murphy
Information Electronics
The Ohio State University
Columbus, OH 43210

Abstract—— **RF/Analog System Design often requires many more constructs than are present in Verilog and VHDL-AMS language extensions. These additional constructs are vendor specific and thus constrain the design database to a specific vendor. This paper describes the design of an LNA as part of a Software Radio architecture to highlight some of the issues in using an HDL approach for system design.**

I. INTRODUCTION

This paper focuses on system development that standardizes the simulation procedure using an RF system design, with the detailed design of a critical LNA module as an example. The key point is that the detailed requirements of critical modules, such as the LNA, can only be finalized by either being a highly experienced LNA designer, or by performing detailed design, the latter proceeding all the way to the physical layout level of a design flow.

At the early stages of the design flow, the main concentration is the system methodology, which emphasizes using high-level equation based models in a system language. At the later stages of the design flow, what is needed is a realizable system architecture that contains detailed models for spice integration and verification.

We have found that standardization of RF/Analog design in a Mixed Signal System requires many more constructs than are present in both Verilog and VHDL Hardware Description Language extensions for Analog and Mixed Signal (HDL-AMS). These added constructs are added by CAD tool vendors and are different among vendors[1], negating the ability of HDL-AMS standards to develop design content that is CAD tool independent.

II. SYSTEM DEVELOPMENT

A. *The Art of Mixed Signal System Design*

System models are needed in two places in the design flow. First, in the early stages, system model (A) is used to make high level design decisions by evaluating architectural tradeoffs. Secondly, after the circuit level design phase (after spice), system model (B) is used in the system integration and verification stage of the design to speed up simulation. The A type models are only needed to make some design decisions and any high level language can be used to help the designer make these quick decisions, before proceeding to detailed schematic entry. The main requirement for the system B type model is the integration of system simulations and spice simulations where the spice simulations are used for the detailed schematic design. Thus, some form of hardware descriptive language (HDL) is often used. The most useful part of the B type models is the creation of test-benches that can calibrate the B type models to the detailed schematic design. Unfortunately, there is no standard language for these test-benches, although what is often used is some combination of spice control/post processing commands and vendor specific constructs.

B. *HDL-AMS use for System Development of Wireless Transceivers*

The system model A of an RF front end can be based on communication concepts shown in the block diagram of Figure 1. The system consist of an Antenna, LNA, Quadrature Mixers, Sigma Delta Modulators, and PLL synthesizers. One of the major design objectives is to increase the programmability of the receiver, as in a Software Radio architecture. Due to the limitations of the HDL-AMS approach, low risk system architectures are often chosen in which the RF, analog, and digital portions can be

Supported by TI and WPAFB

designed separately and flexibly while still meeting system requirements. The conservative architecture often sacrifices performance, which then needs to be made up using the most expensive IC fabrication technologies.

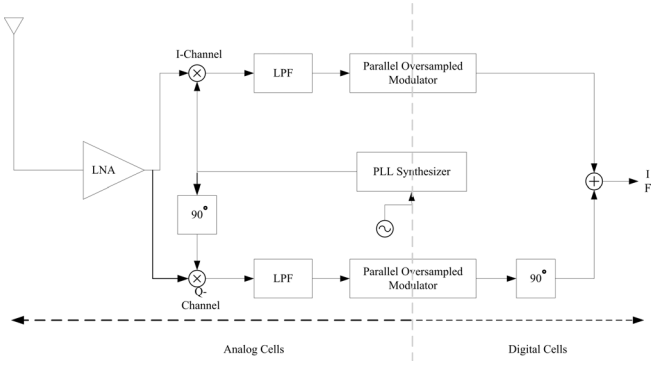

Figure 1. Block diagram representing a system A type model of software radio architecture.

A high risk architecture is the zero-IF architecture in Figure 1., in which the mixers shift the RF signal directly to baseband or near baseband frequencies. At the start of a design flow, a system modeling language is used which enables the design team to reduce risk and quickly get on to the schematic design. We note that the LNA is a critical module in this software radio based architecture, since a more complex RF signal is at the LNA input than is typical of heterodyne architectures, the latter including an expensive (off-chip) RF image reject filter prior to the LNA. Thus, the performance limits of the LNA must be well characterized, and in most cases, this can only be accomplished after the detailed design stages, even taking into account layout and transmission line effects in an RFIC.

Since the detailed design of the LNA and other modules is a crucial part of the design flow, any hurdles and delays with the development and simulation of the system models is a major roadblock to their use. For this reason, we see a wide diversity in modeling languages used for the system A models, such as Matlab, Excel, C++ tools, HDLs,[2] and combinations of these tools. However, most software radio architectures include a large digital core and software as part of the system design. Large digital cores and their software are often modeled in HDLs or SystemHDLs in digital processing systems. Therefore, it is highly desirable to build the total system models in an HDL-AMS. However, because of the inefficiencies of HDL-AMS for RF system level design, the other languages mentioned earlier are often used. In many cases, HDL-AMS approaches have not yet implemented many of the capabilities needed by RF system designers, and it is not clear when this will be rectified [3],[4].

In the following sections of this paper, we described the detailed design of the LNA, and our experience with incorporating the RF design flow of the LNA into an HDL-

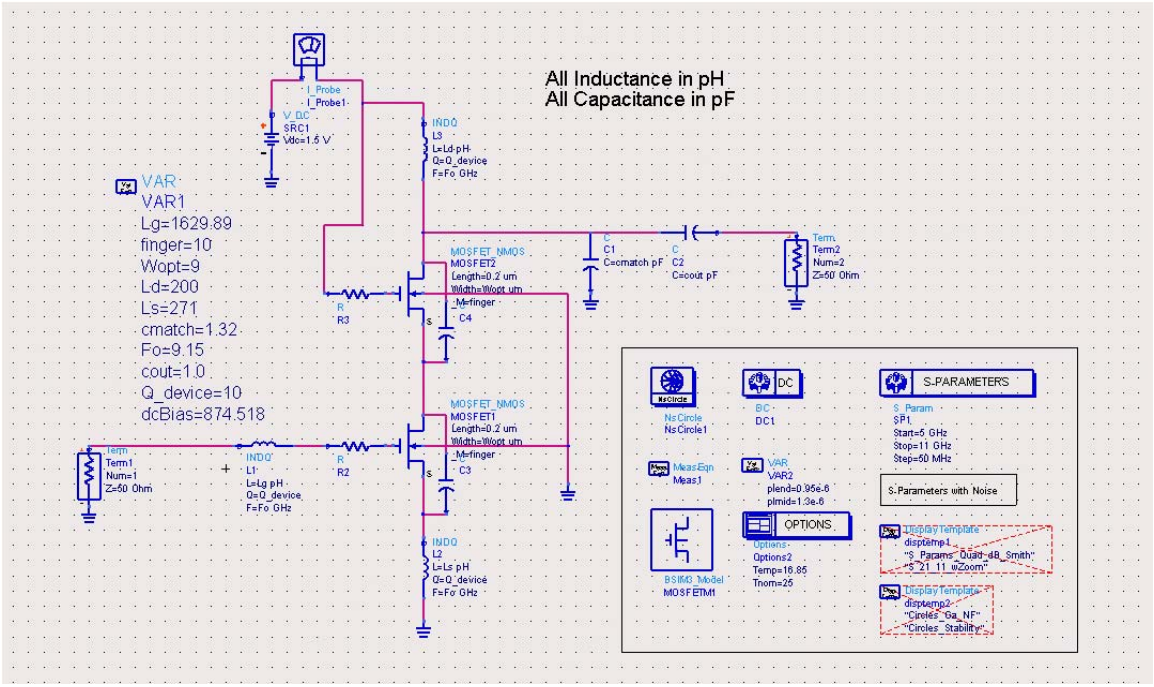

Figure 2. ADS schematic of X band LNA.

AMS approach, using the SystemVision tool from Mentor.

There were a variety of difficulties with using HDL-AMS for LNA design. To illuminate these difficulties, we summarize some of the needed procedures that manifest themselves in the detailed design.

III. RF LNA DESIGN.

Both an S band and X band LNA design were performed in an fully depleted SOI process. The main CAD tools used for the detail LNA design were Agilent's ADS for circuit optimization and Cadence for layout. An ADS schematic for the X band LNA is shown in Figure 2. The Cadence layout is shown in Figure 3. The ADS design flow environment has two features that make it highly effective for RF system design. One, ADS has many library elements to build testbenches for simulation driven design decisions. Second, there are many library elements to calculate performance parameters (eg. noise figure, nonlinearity, scattering parameters) from the simulation results.

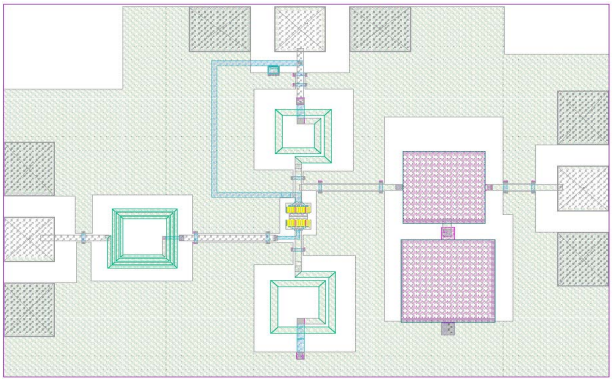

Figure 3. Cadence Layout of X band LNA.

With regards to the analog portion of the mixed signal system, the system models should be done minimally, since they are not as useful as the digital system models. The reason for this is, in a full tapeout design flow, the detailed analog schematic will still need to be generated by the designer, in order to perform layout vs schematic (LVS) and other cell level design verification operations. This is not the case for digital design, in which the detailed schematics for LVS are generated automatically, that is once the system HDL as been created, the circuit schematic and layout are generated automatically. These detailed schematics with the attached transistor models can then be used to perform system simulations using computer clusters. Hence, Figures 2 and 3 depict an experienced view of the LNA design.

IV. VHDL-AMS IMPLEMENTATION

All of the VHDL-AMS simulators we have evaluated, to date, focus on system level representations. Mentor Graphic's SystemVision is, of course, no different. It is clear that this tool and others like it, can have a significant impact on modeling mixed-signal/mixed-domain systems. However, when we attempt to press these tools for System B type modeling, we face a number of challenges.

In order to obtain simulation results useful at this stage of design, the transistor models must be of high fidelity, that is, they must match the characterizations of the foundry. At this point, most of these characterizations are in the form of BSIM models. Unfortunately the current version of SystemVision does not support IC spice models. This isn't an unfamiliar hurdle when dealing with spice model support. The number of "flavors" and levels of spice make co-simulation between EDA tools difficult.

Yet another issue observed, when dealing with AMS tools geared towards system level design is the representation of simulation results. In the digital domain, transient analysis waveforms are all that are needed in most cases. Analog and RF domains, on the other hand, also require frequency analysis and extensive post-processing to yield pertinent information about the design under test.

In order to get S parameter measurements in SystemVision, two AC analyses were needed along with waveform expression manipulations, if strictly VHDL-AMS was used. Luckily, SystemVision has some Eldo capabilities built-in which allow for spice commands to instantiate these measurements in one AC analysis, which leads to a key issue.

VHDL-AMS does not have simulation and output control built-in like spice does. In our opinion this is unfortunate. There are many advantages to under-defining the language. This affords vendors the opportunity to exercise some freedom in implementation among others. Conversely, it doesn't standardize some useful functionality, such as the ability to control the simulations i.e. .AC, .TRANS or define expressions and control output i.e. .DEFINE, or .PLOT, .PROBE. These commands, along with others of this ilk, are well used in the analog world. It would be nice to see VHDL-AMS extended to support this functionality and perhaps even extend it to include the sequential control of simulations or a "design of experiments".

V. CONCLUSION

In any new system development, the system designer should be able to quickly integrate the functional blocks of the design to explore the conceptualization process and reduce risk. However, this requires large libraries of pre-built and tested blocks. Initially it requires lots of time and effort to develop and test libraries within a uniform platform, and to document the design. In addition, the questions that system engineers face are what are the sufficiency requirements for the system model and how useful are the HDL-AMS models in early system design development?

These questions are usually answered more efficiently with experience. Ironically, the system B type models are developed as part of a bottom up design activity and HDL-AMS finds its most common use in the bottom up portion of the design flow, where the integrations take place.

The methodology discussed in this paper requires both top-down and bottom-up design flow, that is the use of simulations and verification process at both the system and circuit level of abstraction. This type of methodology allows for 1) the exploration of RF system design at the same time setting up parametric libraries using HDL-AMS 2) creation of test-benches for design re-use and 3) the integration of analog and digital circuitry.

REFERENCES

[1] Geoffrey Ying, "Start at the Top to Reduce Re-Spins for Analog-Digital Chips," Chip Design Magizine, pp. 24-27, June/July 2005.

[2] Wei Yang, Hal Carter, and Jianping Yan, "A high-level VHDL-AMS model design methodology for analog RF LNA and Mixer," Behavioral Modeling and Simulation Conference, 2004. BMAS 2004. Proceedings of the 2004 IEEE International, pp. 125-129, October 21-22, 2004.

[3] Gennady Serdyuk and Doug Goodman, "VHDL-AMS Extensions For Frequency Domain Harmonic Balance Simulation," 11th International Conference "Microwave & Telecommunications Technology," pp. 108-111, September 10-14, 2001.

[4] John Willis, Ph.D. and Joshua Johnson, "Language Design Requirements for VHDL-RF/MW" IEEE MTT-S Digest, pp. 2093-2095, May 2002.