The Model and Algorithm Prototyping Platform (MAPP)

NEEDS 2012-2017 (needs.nanohub.org)
PI: Jaijeet Roychowdhury, U.C. Berkeley
May 2017

Background

Our motivation for starting this project in 2012 was the fact that device compact modeling, although an old and seemingly mature discipline, suffered from a very basic problem: the development time taken for a new device to become genuinely useful for simulation in circuits and systems was typically measured in years. The norm was for device models to be released, then to prove inadequate in simulation, and much time to elapse before reasons were identified, addressed and the model updated to the point that it became robust in simulation and useful for design. A big reason for this state of affairs was that there was no easy way to develop, test and debug a device's numerical characteristics or physical soundness in a standalone fashion, i.e., in a manner independent of simulator implementation (which makes it difficult to narrow down the causes of problems in models). Another motivation for the development of MAPP was the increasing number of people developing compact models as a part of exploratory research on new devices and applications. The people doing such work are not compact modeling experts, their interest lies in the materials, devices, or applications, but a compact model is an increasingly important part of their work. Helping people who are new to compact modeling, or who do it infrequently, develop robust compact models that are useful in design was another motivation for this work.

What were the goals?

Our goal was to develop and provide a modern, well-structured, open-source infrastructure that would make it possible to develop/debug high-quality device models that were "simulation ready" – i.e., models that would work well for a broad range of simulations on all properly written simulators – quickly and easily. We felt that this was an especially good time to build this capability, since there is much effort currently ongoing in exploring new types of nanoscale devices, good compact models for which are essential for understanding their value for circuits and systems.

What was accomplished?

Originally termed NEEDS-SPICE, the infrastructure we developed was later re-christened the Model and Algorithm Prototyping Platform (MAPP). MAPP allows a user to quickly and naturally define a device model, test it standalone (i.e., without using a simulator), build small circuits/systems with the model, and simulate them using various analyses – all in MATLAB/Octave. We chose MATLAB/Octave to make MAPP more easily accepted by users without formal programming skills, such as many in the device physics and modelling communities. Moreover, MATLAB/Octave's excellent visualization facilities, fully leveraged by

MAPP, are instrumental in helping device modellers validate and debug their models.

MAPP was designed to be highly modular from the ground up. What this means is that all its components (including its ModSpec device models, circuit/system equation formulation engines, and various analyses) are separate modules (consisting of one, or a few, files) that interact in clearly defined ways with each other. This modularity is crucial for MAPP to be easy to program in, extend and modify in a sustainable manner for the future. Such modularity has been the key factor missing from, e.g., the open source SPICE simulator from Berkeley, and is the main reason why it is so much faster and easier to prototype in MAPP than in other available simulators. MAPP comes with a number of analyses: DC, AC, transient, homotopy, shooting, harmonic balance, and DC parametric sensitivities. New analyses are very easy to implement, typically taking from a few hours to a few days. Today, MAPP consists of about 2000 files containing about 529,000 lines of code and documentation/comments.

How device models should best be represented was, of course, a key question in the design of MAPP. Unlike all prior low-level modelling techniques, we wanted to devise a device model standard that was: a) executable standalone (i.e., no simulator needed), b) easy to examine or write by hand, c) not limited just to electrical devices, but completely general in that it could serve for any physical domain, and for multi-physics devices, d) mathematically well defined, e) easily and directly usable by any simulator, and f) usable by any analysis (simulation algorithm) - not only existing analyses but future ones (see Fig 1). Furthermore, we wanted a general way of specifying details of Newton-Raphson aids/hooks (initialization and limiting), an important feature that has been lacking.

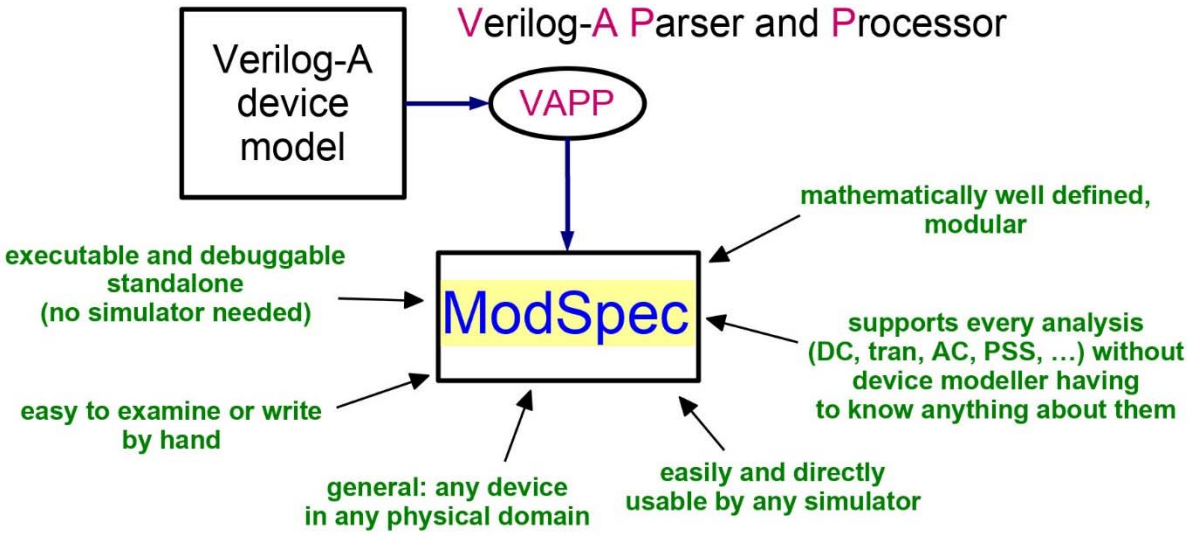


Fig. 1: ModSpec, the model specification language used by MAPP, has several important features that prior alternatives do not provide.

Our ModSpec device model specification system has all these features, which have been instrumental in our being able to quickly and easily detect and diagnose problems in various NEEDS models (described below). We believe that its multi-physics modeling facilities, in particular, will be important in enabling proper modeling and simulation of a new generations of devices; we have prototyped devices that involve thermal, spin, chemical and mechanical constituents. Moreover, the mathematical clarity ModSpec brings to device modeling has already led to new and exciting developments in modeling (also noted below). We have recently proposed to the Compact Modeling Coalition (CMC) that ModSpec be adopted as a low-level modeling standard; the proposal is under consideration. A C++ version of the ModSpec API has been integrated with Sandia's Xyce open-source simulator, enabling it to use ModSpec devices seamlessly. This functionality is expected to be included in Xyce's next release.

Another important component of our effort has been the development of an automatic translator from Verilog-A compact model descriptions to ModSpec. Verilog-A is a high-level language that has been in use in the industry for describing compact models. However, unlike ModSpec, it is not a low-level standard for model description; for example, Verilog-A model descriptions cannot be executed or validated standalone, since it is up to each simulator to implement a Verilog-A engine to interpret it. This has led to compatibiity problems - such as the same Verilog-A model behaving differently in different simulators. Furthermore, the Verilog-A language itself is not best suited for compact models. It was originally developed for a much broader purpose, analog behavioral modelling and simulation, thus provides facilities that are confusing or even dangerous for compact modelling, and are often misused. For example, it is possible using perfectly legal Verilog-A syntax to write a device model that makes no mathematical sense, and it is also easy to write models that are defined for only one analysis and make no sense for other analyses. Indeed, the various deficiencies of Verilog-A as a compact modelling vehicle was one of the chief motivations for us to develop ModSpec. These deficiencies complicated our task of writing a translator to ModSpec significantly, for it had to detect problems with models written in legal Verilog-A syntax, and work around them if possible.

Nevertheless, we recently succeeded in completing and releasing a Verilog-A Parser and Processor (VAPP) that is modularly structured and easily extensible with new capabilities (such as optimizing the generated code). VAPP is able to translate a large slew of Verilog-A models (including BSIM versions 3, 4 and 6, the MIT MVS model, PSP, VBIC, R3 and the Purdue NCFET) into ModSpec, with more to come. It serves as the engine behind a graphical-user interface developed by Purdue, called VAlint, that allows model developers to quickly and conveniently find problems in their Verilog-A model descriptions. VAPP currently consists of 367 files containing about 688,800 lines of code and comments.

All our tools have been released as open source at https://github.com/jaijeet/MAPP and https://github.com/jaijeet/VAPP, with links provided on nanoHUB. Keeping in mind that

MATLAB is a commercial software that not everyone can afford, we spent considerable effort ensuring that our tools also run in the free, open source, MATLAB-language-compatible program Octave – thus providing a completely free and open source toolchain for developing and debugging high-quality models. We have devised a detailed flow for developing well-posed, simulation ready models using our tools, depicted in Fig 2.

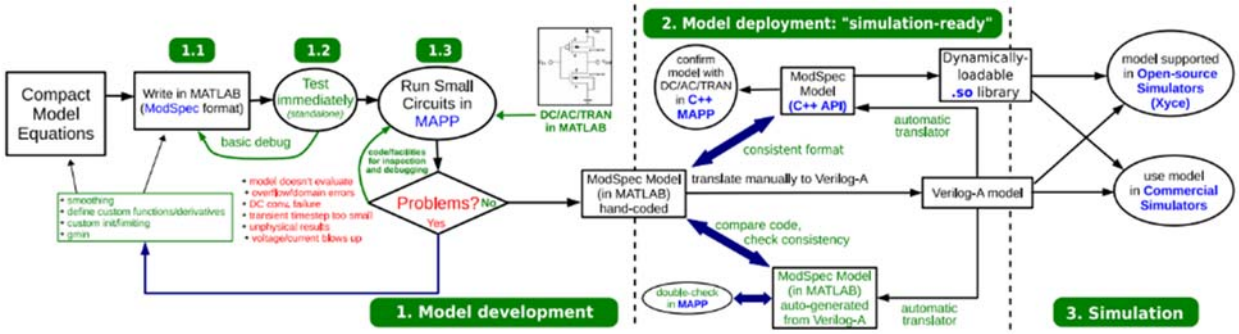
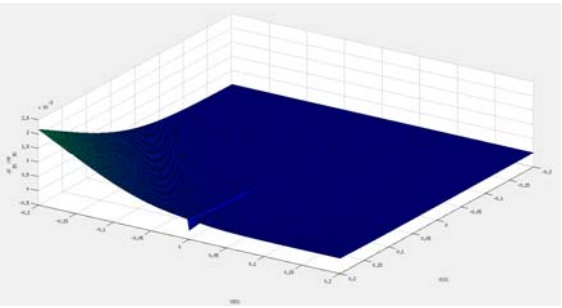


Figure 2: Compact model development flow using MAPP and VAPP.

Why was it important?

The MAPP/VAPP suite of tools was instrumental in the development of several new device models, including MIT's MVS and FEFET models, the Purdue NC-FET model, the Berkeley photodiode model, the MIT RBT model, the Berkeley multi-gate FET model, etc. For example, discontinuities in the MVS model (see Fig 3) were detected rapidly with MAPP and their causes diagnosed. Similar problems had persisted in a released version of BSIM3 for years before detection and correction. MIT's FEFET model, first formulated using MAPP, was characterized and tested in small circuits in a few days (see Figs. 4 and 5). A serious error in Purdue's NC-FET model was found rapidly in MAPP; thanks to this, the error was fixed. MAPP was used to aid the development of the Berkeley multi-gate FET and VCSEL models, and the MIT RBT and optical ring filter models. The group at Auburn University developing the MEXTRAM model has also been exploring MAPP for RF characterization.



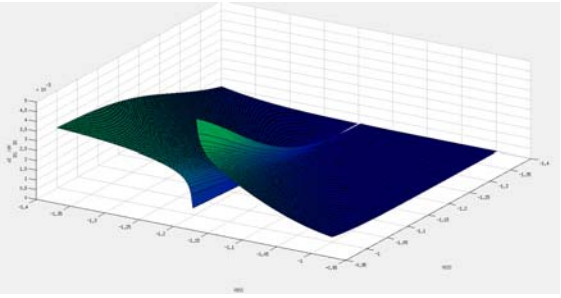


Figure 3: Discontinuities in MVS model diagnosed by MAPP. Left: Discontinuity in I_{DS} at $V_{DS} = 0$. Right: Discontinuity in dI_{DS}/dV_{DS} at $V_{DS} = 1.2$ V.

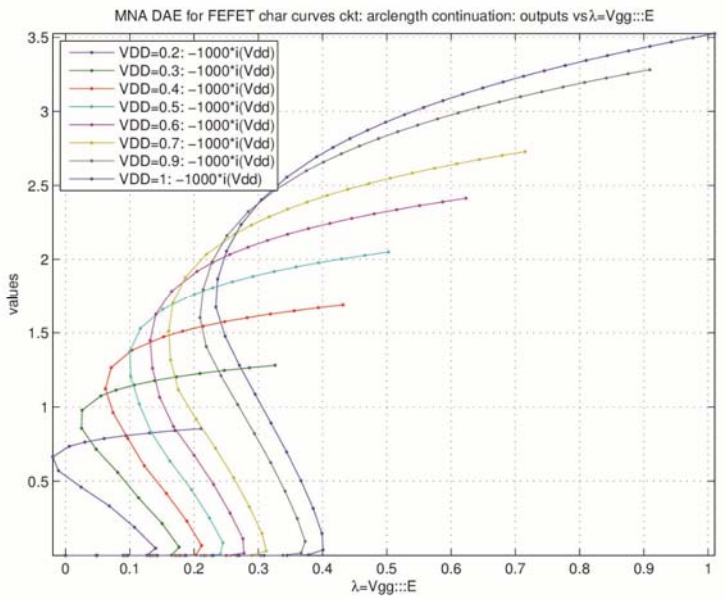


Fig. 4: MIT FEFET model: Drain current vs. gate-source voltage characteristics featuring negative transfer-resistance. Such characteristics cannot be obtained using standard DC sweeps; these were found using MAPP's homotopy analysis.

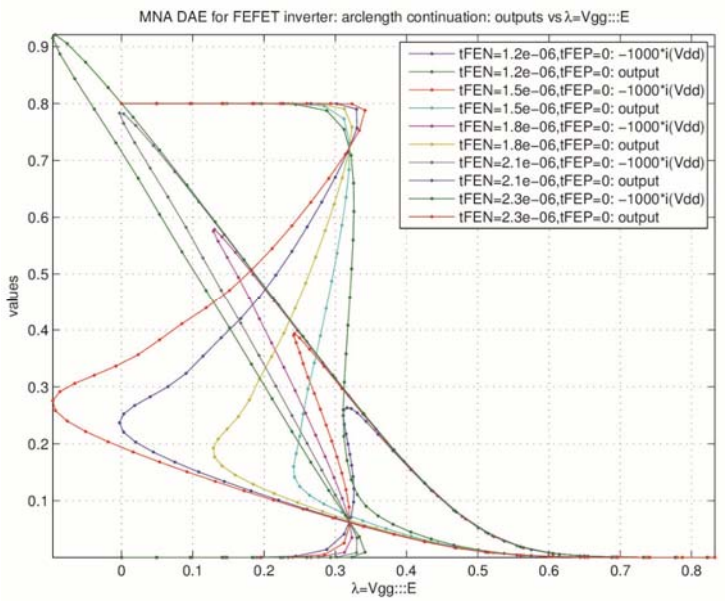


Fig. 5: MIT FEFET model: inverter current and transfer characteristics illustrating multiple operating points and bend-back (folding) regions (generated using MAPP's homotopy analysis).

Spin-offs

The MAPP/VAPP project led to several spin-offs that were not originally anticipated. These include some of the most exciting developments that resulted from this effort.

- Using MAPP, we discovered that not a single existing model for memristive devices (including RRAM) worked properly in simulation. Our attempts to diagnose and repair them led us to develop a suite of 30 well-posed, simulation-ready memristor models based on smooth hysteritic primitives [11]. These models, and the well-posed modelling philosophy behind them, have been rapidly noticed and lauded by the community. For example, the memristor fabrication company, Known Inc., has devoted an article (http://knowm.org/well-posed-memristor-modeling-with-xyce-and-verilog-a) on its blog. These models are being prepared for release on github as the Berkeley Memristor Models (BMM).
- We were able to apply the same technique (modelling hysteresis using smooth primitives) to the long-standing problem of modelling electrostatic discharge devices, widely used in industry for electrostatic protection. With the help and encouragement of Colin McAndrew of NXP, we were able to devise the Berkeley ElectroStatic Discharge (BESD) model [14], currently being prepared for release on github. NXP is in the process of adapting this model for their internal use.
- Although not a part of our original proposal, we were able to provide MAPP with some advanced analysis capabilities (beyond DC, AC, and transient): homotopy/continuation, shooting and harmonic balance. Homotopy is an analysis that is able to solve devices/systems for transfer characteristics that include folds and hysteresis. It was instrumental in the development of ferro-electric FET models and circuits using them (see Figs. 4 and 5). Shooting and harmonic balance, so-called "RF analyses", are used to find periodic or quasi-periodic steady states. These are useful for RF device and circuit characterization, including for finding distortion and intermodulation characteristics. Harmonic balance, in particular, was instrumental in our interactions with the Auburn MEXTRAM group, as well as for developing novel table-based modeling and distortion analysis methods, described below.
- A direct result of MAPP's making it quick and easy to prototype new ideas was an exploration of table-based modeling, initially as a class project at Berkeley. While the notion of using tabulated samples to substitute for equation-based compact models was proposed decades ago, we were unable to find any clear, reproducible technique in the literature for modeling arbitrary devices using tables. But using MAPP, we were able to rapidly devise a technique to do so (ModSpec turned out to be key for clarifying and de-mystifying table-based modeling) and, using spline interpolation, show that tabulated device models can be accurate to 0.1%, while providing speedups of an order of magnitude or more [15]. Recently, we have extended this technique to improve accuracy

to near machine precision – ie, the tabulated model is numerical indistinguishable from the original, hence can be used as a drop-in replacement for any kind of analysis, including RF analyses with stringent dynamic range requirements [23]. This work has debunked a long-held assumption in the community, that table-based models are unsuitable for high-accuracy applications like analog and RF. We expect that it will result in the use of table-based models for a variety of new devices and applications.

- MAPP also enabled a new per-element distortion analysis technique [6] we developed that relies on Harmonic Balance sensitivities rather than traditional Volterra series. The new technique has a number of advantages, including the fact that it provides meaningful results even for circuits with strong distortion (where Volterra series fails). It is also markedly easier to implement in simulators than Volterra series.
- MAPP was also critical in enabling progress on a separate NSF-funded project, PHLOGON, that we have been working on. PHLOGON is a technique to use nonlinear oscillators to implement general-purpose Boolean computation. The oscillator analysis based design techniques we devised for PHLOGON [9] were implemented in MAPP, without which it would have been far more difficult for us to devise and implement PHLOGON computation schemes.

Dissemination, teaching and interactions

The most important dissemination component of our project has, of course, been the open-source release of MAPP (which contains ModSpec) and VAPP on github (with links from nanoHUB) [17, 18].

- Throughout the NEEDS project, we gave a number of talks and tutorials on good modelling techniques, MAPP, ModSpec and VAPP. Most of these have been put up on nanoHUB [1-5, 13, 16, 19-21]. We recently put up a first version of a set of guidelines for good ("NEEDS-compatible") device modelling [19] that we hope will have a far-reaching positive impact on the practice of the art in the years to come.
- Our project and the spin offs described above led to the publications [6-12, 14, 15, 22, 23]. We are also in the processing of preparing a book on the "Fundamentals of Compact Modeling," to be published by World Scientific as part of the NEEDS-led *Lessons from Nanoscience* Lecture Notes Series.
- As part of the dissemination and feedback process for our project, we held three well-attended workshops on MAPP and VAPP – in Feb 2014 (in Berkeley), May 2015 (at MIT) and Feb 2016 (in Berkeley). We also gave a presentation to the SRC on the MAPP project in April 2013, and presentations to the Compact Modelling Coalition (CMC) in August 2016 and March 2017.
- Preliminary versions of MAPP were instrumental for teaching modelling and simulation concepts (including through in-class demos and coding HWs) in three classes taught at

Berkeley: EECS 219A (Numerical Modelling and Simulation), EECS 290A (Numerical Modelling and Simulation 2), and EECS 144/244 (Fundamental Algorithms for System Modeling, Analysis, and Optimizations). A new undergraduate class, EE148 (Fundamentals of Continuous Modelling and Simulation) is in preparation for a first offering in Fall 2017.

- A large number of technical and scientific interactions – with other NEEDS researchers, people from the industry and the general community – occurred throughout the project. These interactions were extremely helpful in guiding the development of MAPP and VAPP. We would especially like to thank Colin McAndrew (NXP), Eric Keiter (Sandia), Jason Verley (Sandia), Geoffrey Coram (Analog Devices), Kiran Gullapalli (NXP), Dimitri Antoniadis (MIT), Bichoy Bahr (TI), Peter Bermel (Purdue), Suprio Datta (Purdue), Guofu Niu (Auburn) and Luca Daniel (MIT). But the most important interaction benefitting the MAPP/VAPP project was unquestionably with the NEEDS project lead, Mark Lundstrom, without whose superlative leadership, vision, encouragement, enthusiasm and help we would not have achieved even a fraction of what we were able to.

References (* denotes work supported or partially supported by NEEDS)

[1] T. Wang, J. Roychowdhury (2013), "Introduction to Compact Models and Circuit Simulation" https://nanohub.org/resources/18616.

[2] T. Wang, J. Roychowdhury (2013), "Guidelines for Writing NEEDS-certified Verilog-A Compact Models" https://nanohub.org/resources/18621.

[3] J. Roychowdhury (2014), "Dealing with Common Numerical Issues in Compact Models (Part 1)" https://nanohub.org/resources/21262.

[4] J. Roychowdhury (2014), "Dealing with Common Numerical Issues in Compact Models (Part 2)" https://nanohub.org/resources/21263.

[5] T. Wang, J. Roychowdhury (2014), "The Berkeley Model Development Environment: A MATLAB-based Platform for Modeling and Analyzing Nanoscale Devices and Circuits" https://nanohub.org/resources/20137.

[6] B. Wu and J. Roychowdhury, "Efficient Per-Element Distortion Contribution Analysis via Harmonic Balance Adjoints" Proc. IEEE Custom Integrated Circuits Conference, September 2014.

[7] J. Yao, T. Wang and J. Roychowdhury, "An efficient time step control method in transient simulation for DAE systems" Proc. ICECS, Dec. 2014.

[8] (invited paper) T. Wang, A. V. Karthik, B. Wu, J. Yao and J. Roychowdhury, "MAPP: The Berkeley Model and Algorithm Prototyping Platform" Proc. IEEE Custom Integrated Circuits Conference, September 2015.

[9] T. Wang and J. Roychowdhury, "Design Tools for Oscillator-based Computing Systems" Proc. IEEE Design Automation Conference, June 2015.

[10] T. Wang, K. Aadithya, B. Wu and J. Roychowdhury, “MAPP: A platform for prototyping algorithms and models quickly and easily” Proc. NEMO, Aug. 2015.

[11] T. Wang and J. Roychowdhury, “Well-posed Modelling of Memristive Devices” arXiv:1605.04897 [cs.ET], May 2016.

[12] T. Wang and J. Roychowdhury, “Multiphysics modelling and simulation in Berkeley MAPP” Proc. NEMO, July 2016.

[13] T. Wang; A. V Karthik; J. Roychowdhury (2016), "MAPP: The Berkeley Model and Algorithm Prototyping Platform" https://nanohub.org/resources/23370.

[14] T. Wang, “Modelling Multistability and Hysteresis in ESD Clamps, Memristors and other Devices” Proc CICC 2017, to appear.

[15] A. Gupta, T. Wang, A.G. Mahmutoglu and J. Roychowdhury, “STEAM: Spline-based Tables for Efficient and Accurate (Device) Modelling” Proc. ASPDAC, Jan. 2017.

[16] J. Roychowdhury (2017), "The Berkeley Model and Algorithm Prototyping Platform" https://nanohub.org/resources/26051.

[17] J. Roychowdhury, T. Wang, K. Aadithya, “The Berkeley Model and Algorithm Prototyping Platform (MAPP)”, 2017-02-15 release of software, https://github.com/jaijeet/MAPP/wiki

[18] A. Gokcen Mahmutoglu, K. Aadithya, J. Roychowdhury, “The Berkeley Verilog-A Parser and Processor (VAPP)”, 2017-02-27 release of software, https://github.com/jaijeet/VAPP/wiki

[19] A. Gokcen Mahmutoglu, T. Wang, A. Gupta, J. Roychowdhury (2017),"Well-Posed Device Models for Electrical Circuit Simulation" https://nanohub.org/resources/26199.

[20] J. Roychowdhury, T. Wang, A. G. Mahmutoglu, A. Gupta (2017), "ModSpec: an Open, Universal, Low-Level Model API: Proposal for designation as a CMC-approved standard" https://nanohub.org/resources/26195.

[21] A. Gupta, K. Aadithya, A. G. Mahmutoglu, J. Roychowdhury, T. Wang, B. Wu (2017), "Formulating ModSpec or: what is a device, exactly?" https://nanohub.org/resources/26197.

[22] A. G. Mahmutoglu, “Graph Theory Based Conversion of Verilog-A Device Models into a Differential Algebraic Equation Formulation” submitted to ICCAD 2017.

[23] A. Gupta, A. G. Mahmutoglu and J. Roychowdhury, “RF-accuracy Table-Based Device Modelling” submitted to ICCAD 2017.