MECS/DesParO: Hierarchical Simulation and Robust Design Aspects

Bernhard Klaassen[1], Tanja Clees[1], Caren Tischendorf[2]

[1]Fraunhofer Institute SCAI, Sankt Augustin, Germany

[2]Cologne University, Mathematical Institute, Köln, Germany

Abstract. We are presenting intermediate results of the Fraunhofer funded project HIESPANA running from 2008 to 2011. Partners are five Fraunhofer institutes and two universities (Cologne and TU Vienna). Main goal is to investigate the influence of parameter changes over the whole design sweep from manufacturing process up to system behavior. The focus in this presentation lies on close coupling of electrical and device simulation for integrated circuits in connection with parameter sensitivity. Coupling here does not mean co-simulation of several different simulators. In contrary, we decided to use a full-system approach which means an inclusion of the relevant equations into one large nonlinear system of partial differential algebraic equations (PDAEs).

Classical circuit simulation (sometimes also called Spice-level) uses device models in form of pseudo-circuits. Such a method is limited in its accuracy by nature. Moreover, it needs some effort to generate these models and calibrate its up to several hundreds of parameters for new semiconductor devices. In order to obtain stable and/or more accurate results, it is often advantageous to insert the PDEs describing the electrical behavior directly into the set of circuit equations. Thus, after space discretization of the PDE parts, we come up with a large set of differential algebraic equations. This approach has been implemented in the experimental simulation framework MECS. MECS comprises python implementations of DAE solvers based on multistep methods (BDF) and general linear methods.

MECS has the ability to include device models from several existing simulators as e.g. Spice, MinimosNT, or Magwel^{RT}. This was not implemented by copying or re-implementing parts of code but by using the above-mentioned simulators in form of shared libraries. Since MECS is written in Python, there are easy-to-use mechanisms to import functions from shared libraries originally generated from e.g. C++ or Fortran. This is not only a practical way of re-using code but also a powerful method to speed-up a Python program.

The large nonlinear system resulting after time discretization has to be linearized and finally solved by a sparse-matrix-solver. Fraunhofer SCAI has a long lasting tradition in solving sparse systems of a certain structure (the software for this so-called algebraic multigrid approach is SAMG). Some moderate effort has to be done to transform the usual matrix structure of MECS into a suitable form for SAMG. First numerical results are briefly discussed.

Finally, we investigate the parameter sensitivity of such a design process by slightly changing the semiconductor parameters (e.g. device doping) and monitoring the effects. For this purpose, the Fraunhofer software DesParO was developed. It assists the designer in the task of generating a robust design in the sense that the resulting chip should react as moderate as possible on certain parameter changes within a defined range. Applications of DesParO to the field of circuit design with mixed-level simulation are presented.