Microsystem Design Using Simulator Coupling

S. Wünsche, C. Clauß, P. Schwarz, F. Winkler

Fraunhofer-Institute for Integrated Circuits, Department EAS Dresden

Zeunerstraße 38, D-01069 Dresden, Germany

Tel.: +49 351 4640 710; Fax: +49 351 4640 703; e-mail: wuensche@eas.iis.fhg.de

Abstract

The microsystem design process is characterized by interdisciplinary approaches and close interactions between different domains. A methodology for simulating the performance of complex microsystems using simulator coupling is presented. The technique is based on the coupling of the FEM program ANSYS with the circuit and system simulator SABER. In difference to other known simulator couplings a time step algorithm is employed. Its methodology is reported and the implementation into simulation tools is explained. The system simulations of an acceleration sensor system as well as the simulation of thermal interactions in integrated circuits prove the suitability of the coupling. Finally, simulation results are discussed and advantages of the implemented coupling are concluded.

1 Introduction

The current trend in microsystem design towards the integration of electronic, mechanical, thermal and other components on chip or on small dimensions requires system analyses during the whole design process. Increasing interactions between the components due to higher integration rates have to be considered. System simulation to perform system analyses support the microsystem design process from the early stage of the specification to the first prototype.

There are two approaches to simulate the behaviour of microsystems - the modelling and simulation with a single simulation tool and the simulator coupling. The first method is based on the modelling of electronic, mechanical, thermal and other components for a single simulation tool. The behaviour of the components can be described with an analog behavioural language like VHDL-AMS or can be modelled by circuits (e.g. macromodels described by netlists). The simulation can be processed by circuit and system simulators like Spectre, ELDO, or SABER. Due to the required abstraction from partial differential equations to ordinary differential equations this approach includes often a complicated modelling process.

The other method - simulator coupling - uses special simulation tools for each kind of microsystem components. So the electronic system part can be simulated with a circuit simulator. A mechanical sensor or thermal effects can be analysed by employing a finite element program. System simulation is processed by the coupling of different simulation tools. Normally the modelling process is well supported but the coupling mechanism with data transfer, synchronization, and convergence control has to be implemented. The advantage of coupling is the opportunity to model special components very exactly, to include special effects into system simulation, and to use the most suitable simulator for the different kinds of subsystems.

The paper focuses on the simulator coupling method. Some approaches to system simulation using single simulation tools are discussed in [1], [2]. The methodology of the simulator coupling and its mathematical background is reported in section 2. Section 3 discusses the implementation of the coupling in the FEM program ANSYS [3] and the circuit and system simulator SABER [4]. The realization of the coupled simulation is shown at the example of an acceleration sensor system and thermal interactions in an operational amplifier circuit in Section 4. Section 5 concludes the results and points out some advantages and disadvantages of the realized coupled simulation.

2 Methodology of simulator coupling

The simulators employed for simulator coupling should meet some typical requirements [5]. The tools should have a time incremental simulation principle. The repetition of calculated time steps should be supported. Moreover, the simulators should have the properties to receive from and to send calculated values to another simulator. The FEM program ANSYS and the circuit and system simulator SABER can be used to realize coupled simulation since they meet these requirements. Time domain simulation is supported and behavioural languages and interfaces to languages like C or FORTRAN are offered. Thus user's own communication interfaces can be implemented. The repetition of a calculated time step is available in both simula-

tors. However, in SABER a restart of the simulator cannot be initiated from a model.

In difference to other known simulator couplings [6], [7] the method employed here is based on an automatic time step algorithm. It works with a variable size of the time steps which depends on the convergence of the system equations. Roughly spoken small changes of the solution values result in large time steps and large changes lead to small time steps. The majority of circuit simulators uses a similar method efficiently.

The implemented algorithm is sketched out in figure 1. Assumed that the results are processed until t_b. $t_x > t_b$ may be chosen. Six steps are required to calculate the next time interval:

① Extrapolation of the SABER results in interval $[t_b, t_x]$

② ANSYS calculation with the extrapolated SABER values (interval $[t_b, t_x]$)

③ Processing of SABER using the results of the ANSYS simulation (interval $[t_b, t_x]$) until 4. is active

④ During simulation the difference between the extrapolated and the calculated SABER value is measured. If it is greater than error criterion Δ_1 the end of the time interval is defined (interval $[t_b, t_e]$).

⑤ ANSYS uses the SABER results to calculate interval $[t_b, t_e]$ again, which leads to Δ_2

⑥ The calculation of the time interval $[t_b, t_e]$ is completed. The new time interval is chosen.

The implemented algorithm of time step control was chosen because it does not need a time step repetition in SABER. This was necessary because a MAST model can not initiate the repetition of a calculated time step in SABER. Thus only one repetition in ANSYS and one SABER calculation per interval $[t_b, t_e]$ are performed. The error criterion is defined by a relative and an absolute convergence parameter. The values of these parameters depend on the simulated system and have to be chosen by the user.

The heuristic procedure of stepsize control may be interpreted as relaxation method [8] performing only one iteration. Therefore it is related to the simple noniterative timing analysis [9]. The arising disadvantages [10] will be compensated by the automatic $[t_b, t_e]$ time window control. The characteristics of the procedure following below have been determined by numeric test calculations:

- Convergence should not be expected if carrying out the iterations with constant stepsize diverged.
- Knowing the consistent initial values is a fundamental precondition for good convergence behaviour.
- The convergence behaviour may be positively affected by the choice of the following timestep $[t_b, t_x]$, which should not be much larger than the last one.

A main advantage of the algorithm is that no iterations are necessary. A disadvantage is the lack in error control. Δ_2 is not always a sufficient error measure. Anyway, for well-posed problems and carefully chosen stepsizes t_x there may be obtained acceptable results, as shown in the enclosed examples. Another simulator coupling method using both time step control and iteration is under investigation.

3 Implementation of the simulator coupling

To implement the time step control algorithm several approaches are feasible. The control algorithm could be realized as a stand-alone software which calls both simulation tools. Alternatively the control routine could be implemented in one of the simulation tools. Due to the availability of programming languages in SABER and ANSYS the second approach was chosen. The simulator coupling pointed out in this paper is based on SABER as
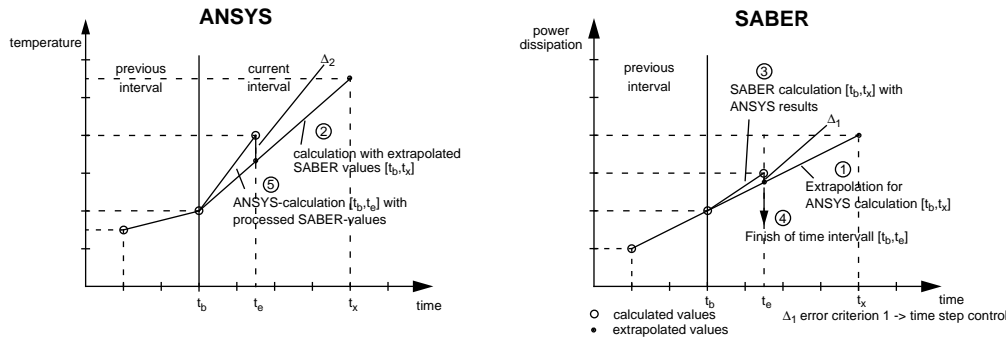

Fig. 1. Algorithm of automatic time step control

the master. This means the control routine is implemented in a SABER model and SABER controls the coupled simulation, defines the time steps, and processes the convergence test. A realization by employing ANSYS as master was also implemented. It is described in [11].

```
template ansys5 e1 e2 e3 e4
thermal e1,e2,e3,e4
 {#Declaration of numbers, values, states, variables, foreigns
   when(dc_init) {
       # receive ANSYS data}
   when(tr_start) {
       # receive ANSYS data}
       }
   when(event_on(tick)&(time<=tend)&(time_domain) ) {
       # receive ANSYS data}
       # calculate time step
       }
   when(threshold(Difference,CC_abs+CC_rel*max,before,after)){
       # finish current time intervall
       # send data to ANSYS}
       }
 parameters {
       # Declaration of parameters}
       }
 values   {
       # Reading of across values
       # Calculation of time dependent temperatur source
       }
  equations {
       # Calculation of through values   }
 }
```

Fig. 2. Structure of the MAST model

The control routine which includes the time step control algorithm was implemented by using the analog behavioural language MAST from SABER. In MAST an analog-digital interface is available which allows the combination of time domain analog models with event driven digital models. As mentioned in section 2 the current time interval is finished if the difference between the extrapolated and the calculated SABER value climbes over error criterion Δ_1. This is realized by creating a digital event at this time. The whole control routine is included into a network model, which is called in the SABER netlist. The behaviour of this model is represented by the ANSYS simulation. A routine written in APDL (ANSYS Parametric Design Language) controls the simulation and the data transfer in ANSYS. Figure 2 shows a simplified structure of the MAST model.

Different methods for data communication between ANSYS and SABER were implemented. The simplest one is the data transfer via file coupling. This means one simulator writes its results into a file and the other simulator reads this data. A more comfortable and faster solution is the use of PVM (Parallel Virtual Machine) [12]. This public-domain software supports the data exchange between different programs on several platforms. Libraries of commands for initialization, sending, receiving, etc. are available in C as well as in FORTRAN. The SABER data transfer procedure was implemented in C and is included into the MAST model. The data transfer in ANSYS was

```
*DECK,USER04        USER  PARALLEL                     SASI
        FUNCTION  USER04 (INTIN,DPIN,CH4IN,CH8IN)
        implicit none
        EXTERNAL  WRINQR
        INTEGER   WRINQR
        include '/home/sim/kosim/pvm3/include/fpvm3.h'
        INTEGER  USER04,INTIN(12), IOTT
        DOUBLE PRECISION  DPIN(12)
        CHARACTER*4  CH4IN(12)
        CHARACTER*8  CH8IN(12)
C
C       INTIN(2) switch FILE-(=0)/PVM(=1)-coupling
C       INTIN(3) number of values to send (ANSYS -> SABER)
C       INTIN(4) number of values to receive (SABER -> ANSYS)
C       DPIN(i)  data to send/receive
C       Declaration of parameters, variables, ...

C       FILE-coupling
        if(INTIN(2) .eq. 0) then
C           write data to file
C           create ready signal
C           waiting for data from SABER
C           read SABER data

C       PVM-coupling
        else if (INTIN(2) .eq. 1) then
            if (begin) then
C               create connection to pvm-daemon
C           send data to SABER
C           receive data from SABER
            if (finish) then
C               delete connection to pvm_daemon

C       write received values to ANSYS
        RETURN
        END
```

Fig. 3. Structure of the ANSYS user comand

realized by writing a user command in FORTRAN. The user command is called by the ANSYS control routine. The structure of the ANSYS user command is depicted in figure 3.

4   Test systems and simulation results

The simulator coupling was employed to analyse the system behaviour of an acceleration sensor system as well as self-heating effects in an operational amplifier circuit. Both systems are characterized by a close interaction between different physical domains.

4.1   Acceleration sensor system

The acceleration sensor system is shown in figure 4. The sensor consists of two cantilever beams and a seismic mass which is the central electrode of a differential capacitor. The evaluation circuit - a synchronous rectifier - extracts the deflection of the mass. The sensor system works in a closed loop and a PID controller delivers the electrostatic force to hold the mass in the middle between the opposite electrodes [13]. Thus the electrostatic force is a measure for the acceleration.

The block diagram of the complex microsystem is sketched out in figure 5. To accomplish the system simula-
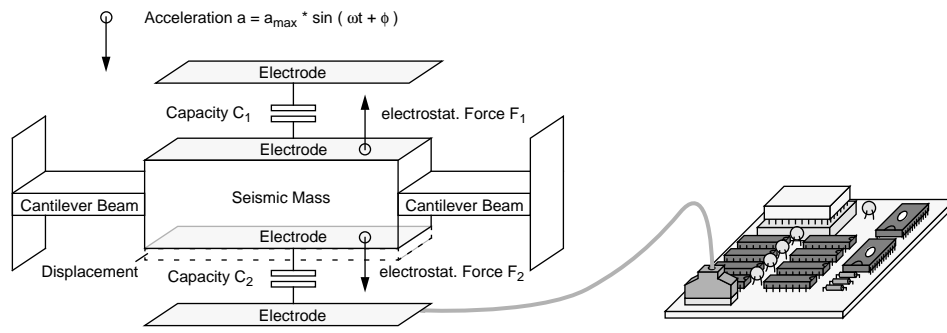
Fig. 4. Acceleration sensor system

tion a system partitioning for the modelling process was necessary. The mechanical component acceleration sensor was modelled using ANSYS. However, the behaviour of all other components was described by the analog behaviour language MAST and analysed using SABER.
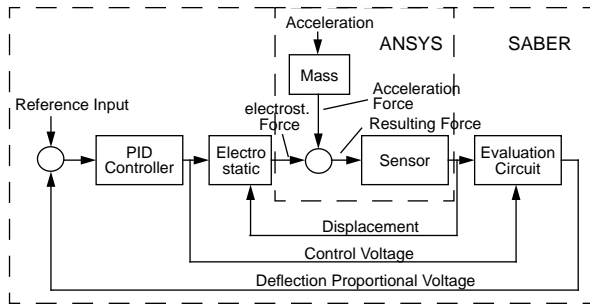

Fig. 5. Block diagram

The ANSYS sensor model delivers the displacement of the seismic mass which is influenced by the acceleration and electrostatic force. The synchronous rectifier transforms the displacement to a proportional voltage. The PID-controller produces the electrostatic voltage which is controlling the displacement. Finally, the electrostatic

model performs the electrostatic force as a function of the electrostatic voltage and the displacement of the seismic mass.

The values to be transferred are defined by the partitioning of the system. Hence, during the coupled simulation SABER gets the displacement of the seismic mass from ANSYS and sends the electrostatic force to the FEM program. The system behaviour was tested by applying a sinusoidal acceleration to the sensor. Figure 6 shows the electrostatic force and the oscillation of the seismic mass around zero. The simulation of the acceleration system requires 137 time steps and takes 5 hours.

4.2 Thermal interactions in integrated circuits

Today higher packaging density, higher speed, and mixed analog-digital solutions in circuit design process result in increasing power density which leads to different analog behaviour and higher failure rates. Examples of thermal induced performance failures include input offset voltage and offset voltage drift in differential amplifiers, reference voltage shifts in regulators and data converters, and nonlinearities in analog multipliers, instrumentation amplifiers, and data converters. The implemented simulator coupling can be employed to analyse such electro-ther-
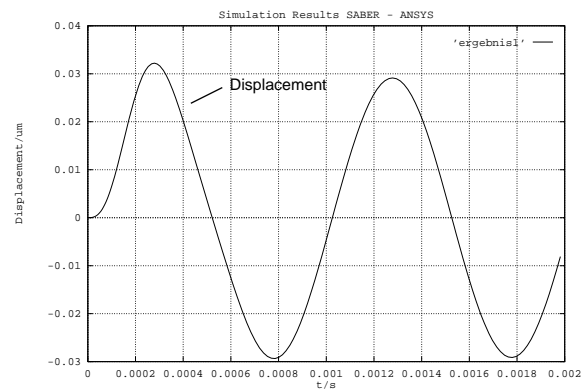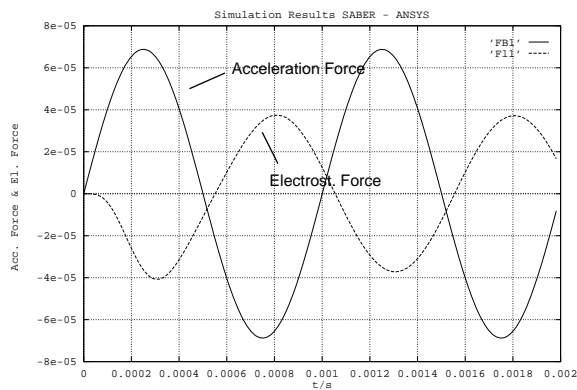

Fig. 6. Simulation results of the acceleration sensor system

mal interactions in integrated circuits. The suitability of the technique is reported at the example of electro-thermal interactions which influences the DC transfer curve of the classical 741 operational amplifier.
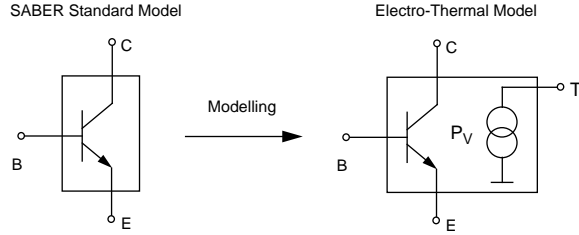

Fig. 7. Electro-thermal model

Currently there are no commonly used SPICE-based simulators which have the capability to set individual junction temperatures during an operating point computation or alter them as transient thermal effects would dictate. This limitation is caused by standard device models which handle the temperature as a parameter. To overcome the limitation an electro-thermal model of the bipolar junction transistor was implemented using the analog behavioural language MAST. A thermal pin T was added which uses the temperature as across value and the power dissipation as through value (figure 7).
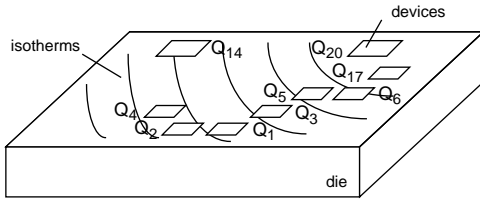

Fig. 8. Simplified layout of 741

Figure 8 shows the simplified layout of the circuit. The simulated schematic of the 741 is depicted in figure 9 [14], [15]. The dissipation characteristic of the class AB-output stage with zero dissipation at V_o=0 and dissipation peaks
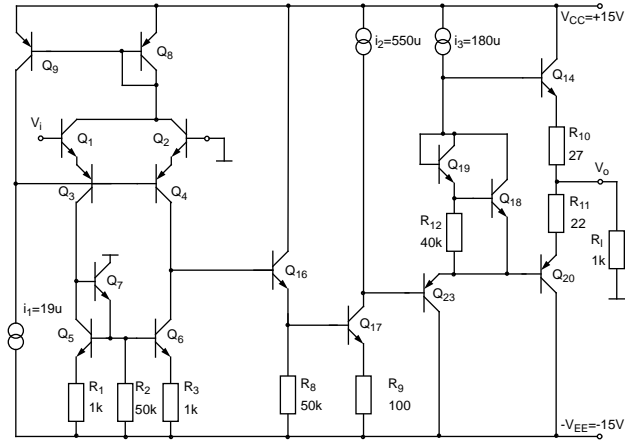

Fig. 9. Simplified circuit schematic of op amp 741

at V_o=+V_{CC}/2 and -V_{EE}/2 lead to small temperature gradients on chip [16]. Thus the critical transistor pairs Q_{1-2}, Q_{3-4}, and Q_{5-6} of the input stage experience different temperatures. Assumed a $1k\Omega$ resistor loading the output a severe distortion of the transfer characteristic is produced by electro-thermal interactions. The distortion can be analysed by processing electro-thermal simulation.

To simulate the electro-thermal behaviour the die/header structure was modelled for thermal analysis with ANSYS. The three-dimensional FEM model was composed of 5120 thermal finite elements. The circuit simulation of the electronic part uses a SABER netlist and MAST models for the transistors, the coupling algorithm and additional components. The implementation considers the power of the output stage and the thermal sensitivity of the transistor pair Q_{1-2}. During the coupled simulation ANSYS calculates the temperature T_{Q1} and T_{Q2} of the transistor devices and sends it to SABER. SABER simulates the circuit behaviour using the received device temperatures. The power dissipations of Q_{14} and Q_{20} calculated by SABER define time dependent loads in ANSYS.
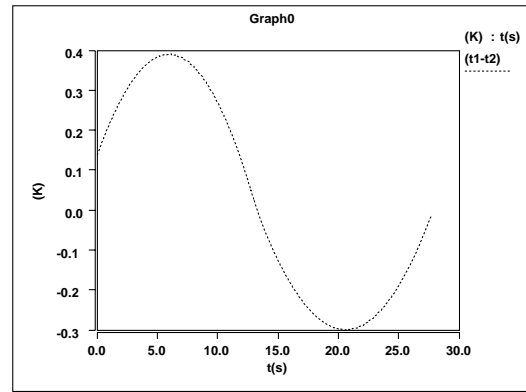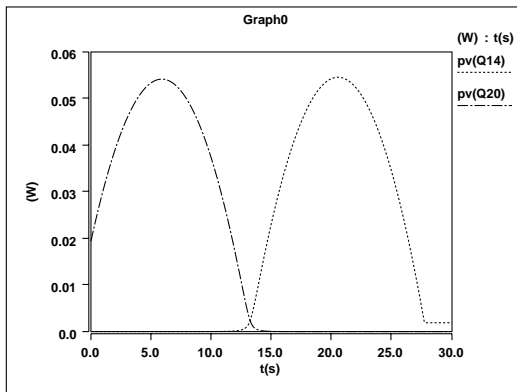

Fig. 10. Power dissipated in the output stage (left), temperature difference between Q_1 and Q_2 (right)

The coupled simulation was employed to analyse the thermal influenced DC characteristic of the circuit. A quasi static transient simulation of the electro-thermal system with increasing input voltage was processed. The simulation results in figure 10 show the power of the output stage transistors. The temperature difference between transistor Q_1 and Q_2 is also depicted in figure 10. Furthermore the DC curve is compared to the DC curve without thermal interactions (figure 11). The simulation results agree with simulation and measured data published in [17]. The coupled simulation takes 6 hours to calculate the behaviour of the electro-thermal system.
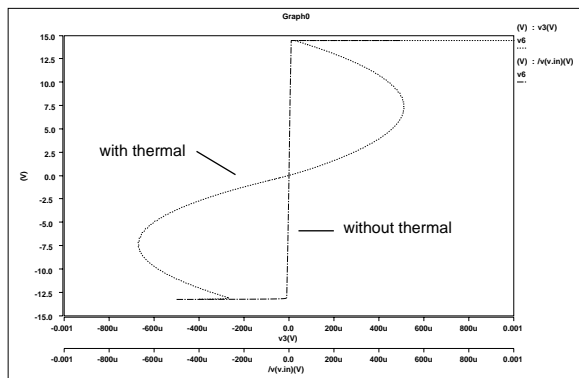

Fig. 11. DC transfer characteristic of the 741 op amp

5 Conclusions

The paper introduces a simulator coupling method to simulate the behaviour of complex microsystems. The method employs a time step control algorithm. The simulation of typical microsystem problems indicates the suitability of the technique. The method offers the advantage to combine different types of models (FEM, netlists, behavioural models) into one simulation run. Consequently, it is a contribution to include special microsystem effects and typical interactions into system simulation. Finally, standard simulation tools like the general purpose FEM program ANSYS and the circuit simulator SABER which are often integrated in the microsystem design process can be used to perform simulations in microsystem design. Moreover SABER can be replaced by other widely used simulation tools like ELDO or SPECTRE. However, the disadvantage of the realized simulator coupling is the time consuming FEM simulation.

References

[1]   Hofmann, K.; Karam, J.M., Glesner, M., Courtois, B.: Entwurf und Simulation von Mikrosystemen. Workshop der G1-Fachgruppe 3.5.6 Mikrosystemtechnik, Schloss Dagstuhl, Germany, 21./22.11. 1994

[2]   Szekely, V. u.a.: Self-Consistent Electro-Thermal Simulation: Fundamentals and Practice. Therminic'95 Grenoble, pp. 188-194

[3]   ANSYS USER GUIDE, Swanson Analysis Systems, Inc. September 15, 1994

[4]   Vlach, M.: Modelling and Simulation with SABER. 3rd Annual ASIC Seminar ans Exhibition. Rochester 1990, T-11.1 - T-11.9.

[5]   Eccardt, P.-C.; Knoth, M.; Ebest, G.; Landes, H.; Clauß, C.; Wünsche, S.: Coupled finite element and network simulation for microsystem components. MICRO SYSTEM Technologies '96, Potsdam 1996, pp. 145-150

[6]   Kao, W.H.; Chu, W.K.: ATLAS: An Integrated Thermal Layout and Simulation System for IC. European Design and Test Conference, Paris 1994

[7]   Petegem, W.V.; Geeraerts, B.; Sansen, W. et al.: Electro-thermal Simulation and Design of Integrated Circuits. IEEE Journal of Solid-State Circuits, vol. 29, No. 2, Februar 1994

[8]   White, J.; Sangiovanni-Vincentelli, A. et al.: Waveform Relaxation: Theory and Practice. Transactions of the Society for Computer Simulation, Vol. 2 (1985), Nr. 1, pp. 95-133

[9]   Newton, A.R.: Techniques for the simulation of large-scale integrated circuits. IEEE Trans. Circ. Syst., CAS 26(1979)9, pp. 741-749

[10]   Newton, A.R.; Sangiovanni-Vincentelli, A.: Relaxation-based electrical simulation. IEEE Trans. Electron. Devices, vol. ED-30, no. 9, pp. 1184-1207 (1983)

[11]   Clauß, C.; Gruschwitz, R.; Schwarz, P.; Wünsche, S.: Simulation mikrosystemtechnischer Aufgaben mit gekoppelten Simulatoren. Proc. 2. Chemnitzer Fachtagung „Mikrosystemtechnik - Mikromechanik & Mikroelektronik, TU Chemnitz - Zwickau, 16./ 17.10.1995, pp. 92-101

[12]   Geist, A.; Beguelin, A. u.a.: PVM: Parallel Virtual Machine. A Users Guide and Tutorial for Networked Parallel Computing. The MIT Press, Cambridge 1994

[13]   Mohr, J.; Keller, W.; Eberle, K.; Lindemann, K.; Fromhein, O.; Krömer, O. Kühner, T.: Präzisionsbeschleunigungssensoren. KfK Nachrichten 1/95, pp. 28-39

[14]   Gray, R.G.; Meyer, R.G.: Analysis and Design of Analog Integrated Circuits. John Wiley & Sons New York 1994

[15]   Fukahori, K.: Computer Simulation of Monolithic Circuit Performance in the Presence of Electro-Thermal Interactions. Ph.D.-Thesis, University of California, Berkley 1977

[16]   Solomon, J.-E.: The Monolithic Op Amp: A Tutorial Study. IEEE Journal of Solid-State Circuits, vol. sc-9, No. 6, December 1974, pp. 314-332

[17]   Lee, S.-S.; Allstot, D.J.: Electrothermal Simulation of Integrated Circuits. IEEE Journal of Solid-State Circuits, vol. 28, No. 12, December 1993