# Microsystem CAD: From FEM to System Simulation

Peter Schwarz

Fraunhofer Institute for Integrated Circuits
Design Automation Department EAS Dresden
Zeunerstrasse 38, D - 01069 Dresden; http://www.eas.iis.fhg.de

**Abstract**

Microsystem technology is a highly interdisciplinary area. Therefore, a combination of different CAD methods and tools is necessary for supporting microsystem design. Process and device simulation are basic CAD methods but more and more higher levels of abstraction need to be applied in order to analyze microsystems adequately. This paper summarizes several modeling and simulation strategies for system simulation of microsystems on different levels of abstraction: generalized Kirchhoffian networks, black-box models, macromodels, the application of hardware description languages, and simulator coupling.

## 1. Abstraction levels in microsystem design

Computer-aided modeling and simulation are very topical in the design of microsystems. Other powerful CAD tools may be used in the construction process e.g. for handling geometrical data and design rule checking. Only first approaches exist for other tasks like specification [40], synthesis [37], and optimization [24]. Modeling and simulation methods depend strongly on the abstraction level. Basic mathematical description means for heterogeneous systems are partial differential equations (PDE) and ordinary differential equations (ODE) for continuous systems. Discrete-event systems (DES), Finite State Machines (FSM), PETRI nets or other automata models are suitable for digital or time-discrete systems:

| | |
|---|---|
| System level | ODE + DES + FSM + ... |
| Subsystem level | ODE, DES, FSM, ... |
| Device level | PDE |
| Process level | PDE |

Digital systems are not taken into account in this context because they are covered by traditional CAD methods which are not specific to microsystems. *Process simulation* is not included in this paper either. Well-known programs like SUPREM support this technological design step.

On the *device level*, the numerical solution of PDEs is also the basis of simulation and may be carried out with simulators like ABAQUS, ANSYS, CAPA, FLOTRAN, NASTRAN, SESES, ... . We have to start here with the development of system models.

On the next level of abstraction, ODEs are commonly used as mathematical models in continuous *subsystem* simulation. It depends on the simulator and on the subsystem's physical domain which of the mathematically equivalent representations of ODE's will be given preference:

| | |
|---|---|
| • mathematical expressions | Mathematica, Maple, Macsyma,... |
| • block diagrams, signal flow graphs | Matlab, MatrixX, Dymola, ACSL, Simplorer,.. |
| • multi-body systems | ADAMS, NEWEUL, ITI-SIM,... |
| • electrical (or non-electrical) networks | SPICE-related electronic simulators |

On the *system level*, complete microsystems consist of mechanical, electrical, magnetic, thermal, pneumatic, fluidic, optical, ... subsystems and devices. Adaptive control and numerical calibration may be realized by electronic signal processing. From the *system* simulation point of view, microsystems may be characterized by the following features:

- complex, heterogeneous, mixed-domain systems
- distributed and lumped (concentrated) elements and devices
- partially strong coupling between and in some subsystems (side effects, cross coupling)
- analog and digital electronic subsystems → mixed signal simulation
- continuous subsystems with extremely different time constants (stiff ODEs)

Typical microsystems can only be described considering coupled physical effects. The solution of coupled field problems is therefore intensively investigated [14], [20], [34], [36], [46]. Obviously there are many common features with mechatronics, robotics, and microelectronics. Therefore, CAD methods from these disciplines may be used successfully also in the design of microsystems.

The goal of system simulation is the evaluation of the *overall system behavior* rather than the very precise simulation of devices in all details. Therefore, *modeling by abstraction* is a major task. Due to the heterogeneity of microsystems it will not always be possible to construct models with a reasonable effort for one system simulator. *Coupling* of different specialized simulators may then solve the problem (see Fig. 1).

We would like to emphasize the criteria for models in an integrated design system: consistency, transparency, and tailored validity [47]. Especially the last aspect seems to be very important in microsystem design: „Model tailoring is driven by the necessity of economizing the computational effort in the simulation of complicated microstructures. Using a general-purpose simulator with sophisticated physically based models for the simulation of full microsystems would not only require excessive CPU time and memory but, besides, would yield much superfluous and redundant information. Instead, the number of degrees of freedom must systematically be adapted to the special properties of a given structure by selecting the proper state variables and relevant basic equations for each individual system component. Doing so we introduce not more dynamical degrees of freedom as necessary, but just as few as possible" [47]. Multi-level and mixed-mode models are valuable approaches to reach this goal. Macromodeling proved to be very successful approach in electronics and may be extended to microsystems.

## 2. Choice of system simulator

One major criterion which kind of simulator should be used is the basic choice of the modeling approach:

- equation-based modeling → equation-based simulators:
  *mathematical equations or block diagrams are used as simulator input*
- physically-based, structure-oriented modeling → simulators for conservative systems:
  a (generalized) network description is the main simulator input

Multi-level, mixed-mode simulators developed in the last years especially for simulating electronic systems (ELDO, SABER, Smash, Spectre, ViewSimA/D, ...) are very useful in the design process of heterogeneous microsystems as well. A basic theory for considering non-electrical systems as (generalized) networks exists and facilitates the usage of the specific advantages of these simulators:

- graphical input facilities which support a structure-oriented modeling style
- a network-based model structure is related closely to the original system structure - this is one important aspect of object-oriented modeling [3]
- first libraries with models of non-electrical components are available for the simulation of mechatronical systems or microsystems [15], [39]
- conservative as well as non-conservative signals for simulation of networks (bidirectional signal flow) and block diagrams (unidirectional signal flow) can be handled

- mixed-signal (analog-digital, continuous-discrete) simulation is supported
- behavioral models may be included: equation-based modeling is very convenient
- multi-disciplinary description languages are under standardization (esp. VHDL-AMS)

Subsystems from different physical domains (mechanics, electronics, hydraulics, ...) may be combined to a complex system in one simulation run. There is no longer a large difference between „system simulation" and „subsystem simulation". All these advantages are reasons for our orientation on modern circuit simulators which are much more powerful than the well-known SPICE-type simulators with their restricted modeling capabilities.

Applying these modern simulators, a „modeling flow" from device level (distributed element description) to subsystem level (lumped or concentrated elements) must be supported. In Fig. 2, often used modeling methods are sketched out. In practice, all of these methods may be combined pragmatically according to the specific needs of a given problem (that's why modeling looks more like an art than a science). Most of these transformation steps are carried out manually today because only a few tools exist but they will be done automatically in future.

## 3. Modeling: from device models to (sub-)system models

We will especially look at the step from device-level models to subsystem models following the lines sketched out in Fig. 2. The next step from subsystem models to an overall system model is also very important but not considered here since it is a general problem and typical not only for microsystems. At least three questions need to be discussed:

- which modeling method may be applied?
- which algorithms and tools exist for supporting the modeling process?
- how is the input information for the simulator generated?

### 3.1 Generalized Kirchhoffian networks

Generalized Kirchhoffian networks are adequate model types when a structure-oriented modeling approach and the above-mentioned system simulators are used. Starting with the classical four-pole theory, n-poles proved to be very useful as models of complicated static or dynamical components in electronics. Using analogies between electrical, mechanical, acoustical, thermal, ... domains, network-based modeling approaches were developed in non-electrical domains long time ago [19], [21], [31] and are now applied to microsystems [9], [18], [27], [35], [44]. The main idea is to decompose a complex system into components (or subsystems) coupled together by quantities which may be distinguished into *flow* and *differences* quantities (other names are *through* and *across* quantities). Field quantities are „concentrated" to 1-point or 2-point quantities which are related to the interconnections between the subsystems. Such quantities are e.g. forces, currents, and moments as well as velocities, voltages, and angular velocities, respectively. Conservation laws exist for these two kinds of quantities (generalized Kirchhoff's laws). The components are completely characterized by *relations* between flow and difference quantities at their terminals in consequence of very general basic principles originating from irreversible thermodynamics [46] or other field-theoretic approaches [41]. Similar decomposition principles are well-known in mechanics [16], [29] but are usually not conceived as a network approach. Bond graphs [3] are closely related to the network approach. In [4], [10] we proposed a unified system of equations based on the *terminal description* of generalized n-poles, see Fig. 3.

Many components may be modeled simply as two-poles (spring, translational or rotational mass, resistor, ...) or two-ports (transformer, gyrator). The more general approach decomposes a complicated microsystem into n-poles. They may be decomposed further into an interconnection of basic components (structural modeling) or may be described by algebraic-differential equations (behavioral modeling) or both of them. Usually, we prefer a combined *structural-behavioral modeling approach*. In contrast to electronics, multidimensional quantities (e.g. forces in x-, y-, and z-direction) are used. The models of reusable components may be stored in a block library. These basic building-blocks may be used to compose models of other microsystems.

In Fig. 4, this approach is sketched out for a complicated mechanical translational acceleration sensor [27]. The same principle applied to a rotational sensor is described in [22]. The behavioral description of the basic elements were derived analytically from FEM theory. But network models (or their equivalent behavioral descriptions) may also be constructed by discretizing the PDEs describing the elements [30], [38].

### 3.2 Black-box models

The application of basic ideas from system theory and control theory also leads to effective modeling methods. In all cases, we use detailed results from simulations on lower levels of abstraction to construct simplified and much more abstract models. For special structures, the analytic derivation of models is possible, of course.

a) Nonlinear static models

Various mathematical optimization, approximation, and interpolation methods exist for curve fitting and parameter adaptation [1]. Apart from classical methods with polynomials, rational functions, and splines, the application of *radial-basis functions* (RBF) is very promising [13]. RBFs may be used for interpolation as well as for approximation, they are very robust and relatively independent of the order of the problem (many spline algorithms are restricted to 1- or 2-dimensional problems).

b) Linear dynamic models

These models are often formulated as transfer functions (pole-zero or rational function representation in the Laplace domain). The system behavior is first simulated on the device level in the time or frequency domain. Standard algorithms from control and system theory may then be used to calculate the transfer function [42]. The response of a linear system may be calculated by convolution of the impulse response with the actual input signal. But this numerical process is very time-consuming and erroneous for long input sequences. *Recursive convolution* is an improved method often applied in the analysis of lossy electrical transmission lines [28]. After approximating the impulse response by exponential functions, the convolution integral may be solved analytically. We used this approach in modeling an electro-mechanical acceleration sensor and to simulate it together with the control circuit [45].

It should be pointed out that some of these approaches are based on the calculation of step or impulse responses. But the application of more complicated input functions, e.g. random binary sequences, may render necessary if not enough a-priori knowledge about the system is available [2], [11]. In 2- or 3-dimensional mechanics, the adequate choice of stimulus signals in different spatial directions is necessary to enforce all internal vibration modes.

These methods lead directly to non-conservative models (to be described e.g. as block-diagrams representing transfer functions) but may be generalized to multi-port descriptions of conservative subsystems. In this case, different transfer characteristics (e.g. the four-pole parameters y11, y12, y21 and y22) have to be modeled simultaneously.

c) Nonlinear dynamic models

Only a few approaches exist for this very general case. Process identification methods [12] are available from control theory. But these methods are mostly restricted to low-order models. Some of their advantages (e.g. the applicability on randomly disturbed signals) are of minor importance when applied on FEM simulation results obtained with high accuracy. So, most nonlinear dynamic models are often based on presumptions about the internal model structure composed of basic functional blocks. Their parameter values are calculated

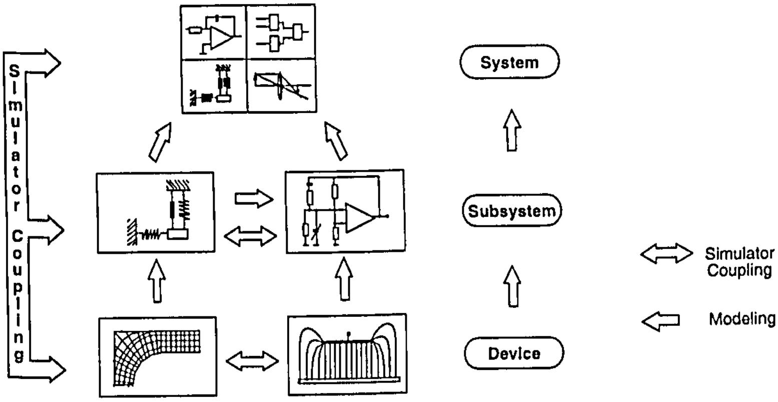



**Fig. 1:** Modeling and simulator coupling in microsystem simulation

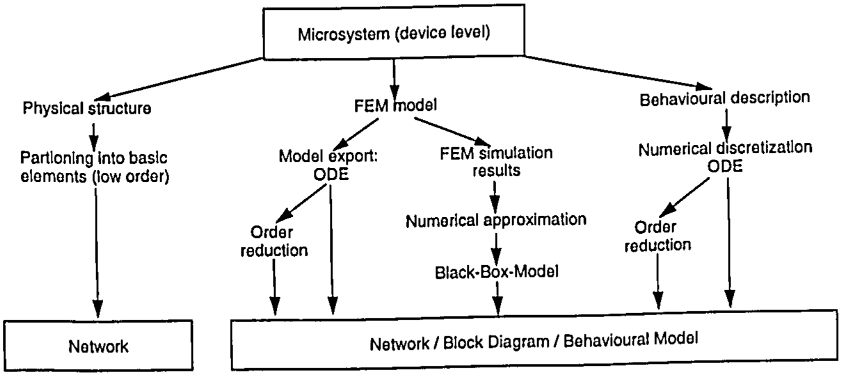
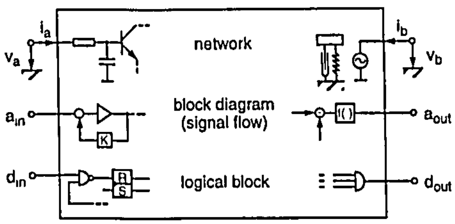



**Fig. 2:** Modeling Flow




$$i_a = f_1(v_a, \dot{v}_a, i_b, \dot{i}_b, a_{in}, \dot{a}_{in}, s, \dot{s}, d_{in}, p, t)$$
$$v_b = f_2(v_a, \dot{v}_a, i_b, \dot{i}_b, a_{in}, \dot{a}_{in}, s, \dot{s}, d_{in}, p, t)$$
$$a_{out} = f_3(v_a, \dot{v}_a, i_b, \dot{i}_b, a_{in}, \dot{a}_{in}, s, \dot{s}, d_{in}, p, t)$$
$$0 = f_4(v_a, \dot{v}_a, i_b, \dot{i}_b, a_{in}, \dot{a}_{in}, s, \dot{s}, d_{in}, p, t)$$
$$d_{out} = f_5(v_a, \dot{v}_a, i_b, \dot{i}_b, a_{in}, \dot{a}_{in}, s, \dot{s}, d_{in}, p, t)$$

s = vector of additional state variables
p = vector of parameters
t = time

**Fig. 3:** Unified terminal description

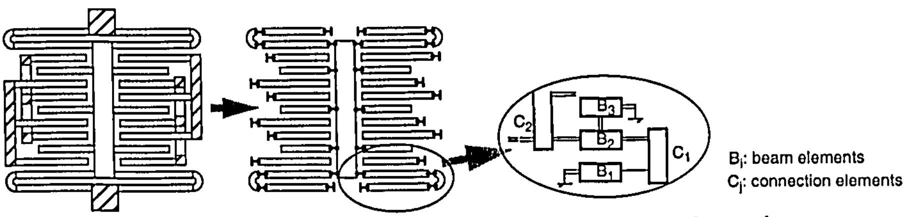



**Fig. 4:** Principle of an acceleration sensor and its decomposition into beam elements

with dimensioning formulas or with optimization programs. Especially in electronics such an approach is well-known as *macromodeling.*

### 3.3 Macromodeling

The basic idea is to compose a system model from various basic blocks. Each of them reflects at least one important property of the system to be modeled. A simple but typical example is the macromodel of an operational amplifier consisting of 3 stages in a chain: a nonlinear input stage, a transfer stage (linear dynamic), and a nonlinear output stage.

Of course, this idea is not restricted to such simple structures and to electrical systems. Traditionally, macromodels were realized as electronic circuit models especially for SPICE-like simulators [5]. But the concept is very general and may be realized also with pure behavioral or with mixed structural-behavioral models. It is an interesting approach to change the structure (and not only the parameter values) of a macromodel with optimization procedures [11] or in a knowledge-based framework [23].

### 3.4 Model generation from FEM models

A lot of modeling work is invested in FEM simulators which are widely used in microsystem simulation. This knowledge should be re-used in system-level modeling [10], [26], [36]. A straightforward way to system-level models could be „exporting" these simulator-internal models and embedding these model cores into a shell necessary for system simulators. Newer versions of ANSYS support this model export for restricted model classes in form of the „substructuring feature". The generated „superelements" are originally intended for re-use in ANSYS but may be the basis for system-level models, especially if this approach is combined with order reduction methods.

The method is particularly interesting if the source code of the FEM simulator is available and not only the general user-interface. The *fully automated generation* of a static linear n-pole behavioral model of a thermal system, starting with its simulator-internal FEM model, is reported in [49]. (A similar method based on FDM approximation is described in [38].) This thermal behavioral model may be coupled with the electrical model of the circuit and then the electro-thermal interactions may be simulated using a circuit simulator.

### 3.5 Description Languages

Behavioral description of devices and subsystems is supported by „Hardware Description Languages" (HDLs) in electronics. HDL-A, MAST, SpectreHDL and the forthcoming VHDL-AMS (now under IEEE standardization [43]) are such languages of practical importance that are also successfully applied in microsystem modeling [11], [32]. Modern HDLs fulfill requirements necessary also for microsystem simulation:

- multi-domain description: electrical/mechanical/magnetic, ...; time/frequency/Laplace; time-continuous/time-discrete; analog/digital, ...
- clear distinction between interface and algorithmic kernel of the model (VHDL-AMS: language constructs ENTITY and ARCHITECTURAL)
- interface to embedded C programs
- mechanisms for handling nonlinear ODE to be solved by the simulator's algorithm

Other languages not originally focused on electronics are of increasing importance for microsystems, e.g. the object-oriented language Modelica [8] which will have broad areas of application like mechatronics and control systems.

## 4. Simulator coupling

Another efficient way for handling the complexity and heterogeneity of microsystems is to couple the simulators which are specialized in one modeling level or one physical domain.

This considerably simplifies the task of modeling but the price is the implementation of coupling software (which is not always available from CAD vendors) and, in general, the investment of much more computation time (especially in coupling ODE simulators or network simulators with FEM simulators). Different aspects have to be considered:

- algorithms: iteration methods; relaxation or Newton-like; predictor/corrector methods,...
- implementation: PVM, TCP/IP, UNIX sockets, ... for communication between computers; extension of simulator interfaces for data exchange
- overcoming simulator's restrictions: some simulators may be rolled-back over the last iteration interval, others must be re-started at t = 0; some simulators have user interfaces which may be used for exchanging actual simulation results, others need modifications of the input language (esp. the stimulus description).

Unfortunately, a standardized simulation backplane does not exist. Very useful results might be gained, however, with some realized simulator coupling solutions for the analysis of coupled effects (e.g. in controlled electro-mechanical acceleration sensors and in electro-thermal interactions of integrated circuits) [7], [17], [33], [48].

## 5. Conclusion

It was shown in the paper that efficient approaches exist for handling the modeling and simulation tasks in microsystem design. However, a lot of work remains to be done in

- the development of a widely accepted modeling methodology
- the development of user-friendly modeling tools to support this methodology, e.g. automatic construction of low-order models based on FEM simulator models,
- solving coupled field problems and their consideration in system-level models
- generating mixed-domain libraries of building-blocks

**Acknowledgement**

I would like to thank many colleagues for fruitful discussions in the broad field of microsystem modeling: U. Becker, H. Eggert, P.-C. Eccardt, G. Gerlach, R. Laur, A. Maurer, K.-D. Müller-Glaser, R. Neul, K. Reinschke, S. Schulte, G. Wachutka and especially my colleagues in our institute: C. Clauß, J. Haase, E. Huck, S. Reitz, P. Schneider, and S. Wünsche (now at Siemens, Burlington). The funding by the German Ministry of Research (BMBF) in different projects and in SFB 358 „Automated System Design" is also gratefully acknowledged.

**References**

[1] Box, G.E.P.; Draper, N.R.: Empirical Model Building and Response Surfaces. Wiley, 1987.

[2] Casinovi, G.; Sangiovanni-Vincentelli, A.: A macromodeling algorithm for analog circuits. IEEE Trans. CAD-10(1991)2, 150-160.

[3] Cellier, F. E.: Continuous System Modeling. Berlin: Springer-Verlag, 1991.

[4] Clauß, C.; Haase, J.; Schwarz, P.: An approach to analogue behavioural modelling. Proc. VHDL User Forum Europe, Dresden, May 1996, pp. 85-96.

[5] Conelly, J.A.; Choi, P.: Macromodeling with SPICE. Prentice Hall, Englewood Cliffs 1992.

[6] Crary, S. B.; Zhang, Y.: CAEMEMS: An integrated computer-aided engineering workbench for microelectro mechanical systems. Proc. MEMS '90, 113-115.

[7] Eccardt, P.C. et al.: Coupled finite element and network simulation for microsystem components. Proc. MICRO SYSTEM Technologies (MST'96), VDI-Verlag, Potsdam 1996, 145-150.

[8] Elmqvist, H. et al.: Modelica - A Unified Object-Oriented Language for Physical Systems Modeling. Version 1, September 1997. http://www.Dynasim.se/Modelica/Modelica1.html

[9] Gerlach, G.; Dötzel, W.: Grundlagen der Mikrosystemtechnik. Hanser-Verlag, München 1997.

[10] Haase, J.; Schwarz, P.: Modeling and simulation of heterogenous systems. Proc. Workshop on System Design Automation SDA'98, Dresden 1998, 103-110.

[11] Hofmann, K. : Differential model generation for microsystem components using analog hardware description languages. Dissertation, Darmstadt 1997.

[12] Isermann, R.: Identifikation dynamischer Systeme. Springer, Berlin 1992.

[13] Jackson, M. F.; Chua, L. O.: Device modeling by radial basis functions. IEEE Trans. CAS-I 39(1992)1, pp. 19-27.

[14] Kaltenbacher, M.; Landes, H., Lerch, R.; Lindiger, F.: A finite-element/boundary-element method for the simulation of coupled electrostatic-mechanical systems. J. Phys.III France 7(1997), pp.1975-1982.
[15] Karam, J.M. et al.: Low cost access to MST: manufacturing techniques and related CAD tools. Proc. MICRO SYSTEM Technologies (MST'96), VDI-Verlag, Potsdam 1996, 127-132.
[16] Kecskemethy, A.; Hiller, M.: An object-oriented approach for an effective formulation of multibody dynamics. 2nd US Natl. Congress Computational Mechanics, Washington, 1993.
[17] Klein, A.; Schroth, A.; Blochwitz, T.; Gerlach, G.: Two approaches to coupled simulation of complex microsystems. Proc. EUROSIM '95, Vienna 1995, 639 - 644.
[18] Klein, A.; Gerlach, G.: System modelling of microsystems containing mechanical bending plates using an advanced network description method. Proc. MST'96, Potsdam 1996, 299-304.
[19] Koenig, H. E.; Blackwell, W. A.: Electromechanical System Theory. McGraw-Hill, 1961.
[20] Korvink, J.G. et al.: SESES: a comprehensive MEMS modeling system. Proc. MEMS'94, 22-27.
[21] Lenk, A.: Elektromechanische Systeme (3 vol.). Verlag Technik, Berlin 1971 - 1973.
[22] Lorenz, G.; Neul, R.: Network-type modeling of micromachined sensor systems. Proc. MSM98.
[23] Mammen, T. et al.: MASE - Werkzeug zur Generierung von Makromodellen für Mikrosystemkomponenten. Proc. 3. Workshop „Methoden und Werkzeuge ...", Frankfurt 1996, 138-145.
[24] Meinzer, S. et al.: Simulation and design optimization of microsystems based on standard analog simulators and adaptive search techniques. VHDL User Forum Europe, Dresden 1996, 169-180
[25] MEMCAD 3.1, see http://www.memcad.com/products.html
[26] Nagler, O.; Folkmer, F.: FEM-Simulation piezoresistiver Meßwandler am Beispiel eines mikromechanischen Beschleunigungssensors. 13. CAD-FEM User's Meeting, 1995.
[27] Neul, R. et al.: A modeling approach to include mechanical microsystem components into system simulation. Proc. Design, Automation & Test Conf. (DATE'98), Paris, 1998, 510-517.
[28] Nguyen, T. V.: Recursive convolution and discrete time domain simulation of lossy coupled transmission lines. IEEE Transactions on CAD 13 (1994)10, pp. 1301-1305.
[29] Otter, M.: Objektorientierte Modellierung mechatronischer Systeme am Beispiel geregelter Roboter. Dissertation, Bochum 1994.
[30] Pelz, G. et al.: MEXEL: Simulation of microsystems in a circuit simulator using automatic electromechanical modeling. Proc. MICRO SYSTEM Technologies, VDE-Verlag, Berlin 1994, 651-657.
[31] Reinschke, K.; Schwarz, P.: Verfahren zur rechnergestützten Analyse linearer Netzwerke. Akademie-Verlag, Berlin 1976.
[32] Romanowicz, B. et al.: Microsystem modeling using VHDL 1076.1. Proc. Microsim'97, 179-188.
[33] Schrag, G. et al.: Device- and system-level models for micropump simulation. Proc. MicroMat'97, Berlin 1997, 941-944.
[34] Schulte, S.: Simulation of cross-coupled effects in physical sensors. Proc. MST'94, 833-842.
[35] Schwarz, P.: Simulation von Mikrosystemen. 2. GME/ITG-Workshop, Ilmenau 1993, 247-256.
[36] Senturia, S.; Aluru, N. R.; White, J.: Simulating the behavior of MEMS devices: computational methods and needs. IEEE Trans. Computational Science & Engineering, January 1997, 30-54.
[37] Sigmund, O.: Design of material structures using topology optimization. Diss., Lyngby 1994.
[38] Szekely, V.; Rencz, M.: Fast field solver for thermal and electrostatic analysis. Proc. DATE'98, Paris 1998, 518-523.
[39] Tanner Tools MEMS Pro, Tanner EDA, Pasadena, CA 91107, USA.
[40] Tanurhan, Y. et al.: System level specification and simulation for microsystem design. Proc. MICRO SYSTEM Technologies, VDE-Verlag, Berlin 1994, 849-860.
[41] Tonti, E.: The reason for analogies between physical theories. Appl. Math. Modelling 1(1976), 37-50.
[42] Unbehauen, R.: Netzwerk- und Filtersynthese. R. Oldenbourg, München 1992.
[43] VHDL-AMS: IEEE DASC 1076.1 WG Documents. See http://www.vhdl.org/analog/
[44] Voigt, P.; Wachutka, G.: Electro-fluidic microsystem modeling based on Kirchhoffian network theory. Sensor and Actuators A 66 (1998)1-3, pp. 6-14.
[45] Voll, I; Haase, J.: Rekursives Faltungsmodell für ein allgemeines Netzwerksimulationsprogramm. Proc. 40. IWK TH Ilmenau, 1995, vol. 3, 269-274.
[46] Wachutka, G.: Tailored modeling of miniaturized electrothermalmechanical systems using thermodynamic methods. DSC-Vol. 40, Micromechanical Systems, ASME, New York 1992, 183-198.
[47] Wachutka, G.: Tailored modeling: a way to the 'virtual microtransducer fab' ? Sensor and Actuators A 46-47 (1995), pp. 603-612.
[48] Wünsche, S.; Clauß, C.; Schwarz, P.; Winkler, F.: Electro-thermal circuit simulation using simulator coupling. IEEE Trans. VLSI-5(1997)3, 277-282.
[49] Wünsche, S.: Ein Beitrag zur Einbeziehung thermisch-elektrischer Wechselwirkungen in den Entwurfsprozeß integrierter Schaltungen. Dissertation TU Chemnitz, 1998.

# Methods for model generation and parameter extraction for MEMS

Peter Voigt, Gabriele Schrag and Gerhard Wachutka

Institute for Physics of Electrotechnology
Munich University of Technology
Arcisstr. 21, D-80290 München, Germany

Abstract

A strategy for compact modeling is presented, that takes into account the specific problems associated with the development and production of microelectromechanical devices and systems (MEMS). For a MEMS compact model, a set of basis functions is derived from a continuous field model description by proper approximations. This set should represent all relevant coupling effects that determine the device operation, and the most important parasitic effects. The remaining less important effects are taken into account by means of data fitting. The corresponding model parameter extraction process uses direct extraction techniques and well tuned local optimization steps for those parameters of the basis functions, which represent physical, technological or geometrical properties. Local and global optimization steps are used to determine the remaining fit parameters. This results in an effort-optimized modeling approach, which is suitable for statistical modeling and yield analysis, and can be automated in a CAD-toolset.

## 1. Motivation

There are several aspects in which microsystems technology differs vastly from integrated circuit (IC) technology. Integrated circuits are composed of a quite limited number of elementary device structures, fabricated by means of well established and quasi-standardized design rules and process technologies. In the field of microsystem technology, however, an ever growing variety of different device types has emerged, based on rather unconventional design methods and a large number of widely differing (and sometimes brandnew) fabrication technologies. Hence, today's challenge in the computer-aided IC design consists in mastering very complex system topologies built up by a huge number of simple basic elements, whereas in the computer-aided design of microelectromechanical systems we face the problem of describing systems with simple topology built up by a comparably small number of constituent devices which, however, exhibit a high functional complexity based on quite sophisticated and involved physical operating principles.

The complexity of microsystems originates in particular from the often complicated coupling between different energy and signal domains which, on the one hand, is the inherent and much desired property of any sensor or actuator element in a microsystem and, on the other hand, is a detrimental property when it occurs as parasitic cross coupling between the system components. Therefore the accurate analysis of all kinds of physical coupling effects has a major impact on the optimization of microsystems and is thus the most important issue that has to be tackled in the computer-aided design of microtransducers and microsystems. In this sense, the physical-based modeling of microsystems is widely recognized as necessity, even though it becomes increasingly complex, so that the effort as well as the time spent into model development, validation and parameter extraction have to be carefully adjusted to the actual needs, as it has been proposed by the concept of "tailored modeling" [1].

As the state of the art in modeling, simulation, and design optimization is far advanced in the world of microelectronics, it makes sense to revisit the well-tried methodologies for model generation and parameter extraction used in IC technology. In this context, the particular demands on the models must be identified which arise from their dedicated application to microsystems. This will provide us with decision criteria what modeling approach is most appropriate for a given problem and, consequently, should be implemented in a CAD toolbox for MEMS, which then will allow for an efficient and time-economizing computer-aided microsystem development.

## 2. Model development

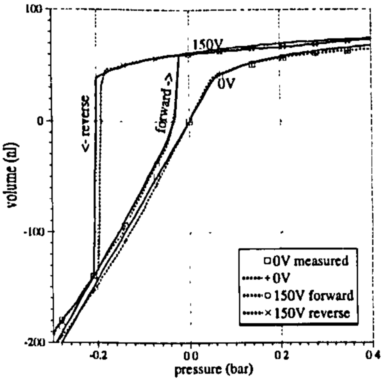
The most rigorous approach to develop a device model is a basic physical analysis of its operation principles. Starting from a continuous field model (CFM) description, the degrees of freedom in the model have to be reduced by proper approximations. This results in a compact model that still reproduces all important physical effects of the device operation correctly, but allows, due to its relative simplicity, the simulation of the device behavior on the system level [1]. The resulting model equations contain parameters, which represent physical, geometrical or technological quantities. They can serve as basis functions and, with the help of additional fit parameters, are used to reproduce the device characteristics. However, for various reasons (complexity of the device geometry, complicated coupling effects) it may turn out impossible to consequently follow this way.

A possible alternative then is the numerical simulation of the device behavior by direct use of the original CFM model, employing FEM tools, for instance. This approach yields accurate information about the operation principles of the device, but the FEM model needs to be set up and interpreted carefully. Otherwise, important effects may easily be overlooked. The required effort ranges from moderate to prohibitive, depending on the device complexity and the availability of adequate software. Notably the coupling effects require special numerical methods.

The easiest way to generate compact models is pure curve fitting, based on measured data or on CFM simulation results. Such a model, if not supported by a physical description of the device operation, lacks all predictive capabilities and can hardly be used to inter- or extrapolate in the space of design parameters and operating conditions.

The choice for the optimum modeling approach depends on technological contraints, the system application of the device, and other factors. If a device is realized in just one version using one given technology, the model may be based on curve fitting procedures. But if a device would have many variants (geometry variations, e.g.), then a generic model is required that correctly reproduces the dependences from technology parameters, operating conditions, geometry etc.

In most practical cases, modeling is an overlap of physical considerations, CFM-based problem analysis, and data fitting. An instructive example is the model of an electrostatically actuated pump membrane (presented in [2]). It demonstrates a compromise between cost and benefits in the modeling process. The electrostatic force between membrane and



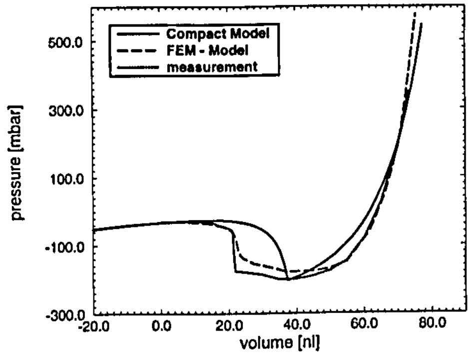




Figure 1: Static characteristics of an electrostatically deflected membrane. The left figure shows volume deflection vs. pressure. The right figure shows the resulting pressure, when volume is forced. A voltage of 150V is applied to the membrane, causing the snap effect.

rigid counterelectrode is modelled physically with only little, well justified approximations. This treatment reproduces correctly the hysteresis phenomenon of the membrane snap-down due to an applied voltage. The shape of the membrane deflection, however, is simply fitted by an sinusoidal function. The effect of shape snapping, as calculated by FEM, is not included in the compact model and therefore cannot be reproduced (see Fig. 1). It is the responsibility of the modeling engineer to decide if this approximation is sufficient for the system applications in mind.

Other examples of such a mixed approach are given in [3], [4] and [5], among others. Here the device behavior is described by linear combinations of appropriately chosen basis functions. These functions are derived from an analytical approximation of the device and/or CFM and, therefore, represent to a certain extent the physical properties of the device. This way of model generation is amenable to automation in a dedicated software environment.

## 3. Model verification and parameter extraction

Model development cannot be considered independently of model validation and parameter extraction. Each modeling approach requires an appropriate extraction technique and, in return, the limitations of different extraction methods may influence the decision for a specific modeling strategy.

In case of a modeling strategy based on data fitting, the model verification is simply a test if the simulated data reproduces the input data from measurement or CFM simulation within the required accuracy. The appropriate extraction technique would be a global optimization algorithm. The advantage is a very fast setup of the extraction, but one pays with typically lengthy optimization runs. The extracted parameters can hardly be used for statistical process monitoring and a physical interpretation is quite often impossible.

If the developed model is based on the physical analysis of the device, a global optimization scheme could easily erode the physical content of the model parameters. Therefore, an extraction scheme utilizing direct extraction steps and carefully tuned local optimization needs to be employed. Only the remaining fit parameters can be extracted by means of global optimization. Model validation becomes a very challenging task since it must be verified, if the model reproduces all relevant physical effects together with their process and geometry dependence. In that case, both steps, model validation and extraction setup, are very time consuming, but the resulting model parameters can be used for physical interpretation, yield analysis, process monitoring, etc. The time required for an actual parameter extraction run is typically short, thus allowing the processing of a large amount of data as it is required for statistical analysis.

The parameters with a physical meaning are (more or less) independent from an actual device structure, thus allowing to extract them by measuring dedicated test structures. A parameter database may be established, that makes simulations possible in an early development stage, before real devices exist.

## 4. Conclusions

A synopsis of the special demands on MEMS device models including the strengths and weaknesses of alternative modeling and parameter extraction techniques led us to the conclusion that a hybrid modeling approach is best suitable in most situations. This approach uses basis functions to fit the device characteristics. These basis functions represent an approximation of the underlying physical effects of the device operation and can be derived analytically from CFM analysis. Their advantage compared to pure fit functions is the fact that they contain 'physical functionality'. The adequate parameter extraction process utilizes direct extraction and local optimization steps for all physically based parameters and global optimization for the fit parameters. Once a library of device classes and their corresponding basis functions have been developed, the modeling and parameter extraction process can be automated, thereby minimizing the required development effort. The suggested modeling approach is suitable for process monitoring and statistical analysis.

## References


[1] G. Wachutka, "Tailored modeling: a way to the 'virtual microtransducer fab'?", *Sensors and Actuators*, vol. A46–47, pp. 603–612, 1995.

[2] P. Voigt, G. Schrag, and G. Wachutka, "Micropump macromodel for standard circuit simulators using HDL-A", in *Proc. of the 10th European Conf. on Solid-State Transducers (EuroSensors X)*, R. Puers, Ed., Leuven, 1996, pp. 1361–1364, Timshel BVBA, Leuven.

[3] E. S. Hung, Yao-Joe Yang, and S. D. Senturia, "Low-order models for fast dynamical simulation of MEMS microstructures", in *Dig. of Tech. Papers of Transducers'97*, Chicago, June 16–19, 1997, pp. 1101–1104.

[4] S. Kurth and W. Dötzel, "Experimental adaption of model parameters for microelectromechanical systems (MEMS)", *Sensors and Actuators*, vol. A62, pp. 760–764, 1997.

[5] N.R. Swart, S.F. Bart, M.H. Zaman, M. Mariappan, J.R. Gilbert, and D. Murphy, "AutoMM: Automatic generation of dynamic macromodels for MEMS devices", in *Proceedings of MEMS'98*, Heidelberg, 1998, pp. 178–183.