# A Computer-Aided Design System for Microelectromechanical Systems (MEMCAD)

Stephen D. Senturia, *Senior Member, IEEE,* Robert M. Harris, *Student Member, IEEE,* Brian P. Johnson, *Member, IEEE,* Songmin Kim, Keith Nabors, Matthew A. Shulman, and Jacob K. White, *Member, IEEE*

*Abstract*—**Although microelectromechanical systems (MEMS) are generally fabricated using enhancements of lithography-based planar technologies developed for the integrated circuit (IC) industry, the three-dimensional nature of MEMS implies that the required computer design aids are very different from those used by IC designers. Specifically, of most immediate use to MEMS designers are programs which provide visualization and electromechanical analysis of three-dimensional structures. In this paper, we describe the MIT Microelectromechanical Computer-Aided Design system (MEMCAD), in which selected commercial software packages are linked with specialized database and numerical programs to allow designers to quickly perform both mechanical and electrical analysis of structures either described directly, or derived from the design specification (mask data plus process flow). The system architecture, the various modules, and their present status are described, and present system performance is demonstrated with several examples.**

## I. Introduction

Although there are very sophisticated computer-aided design (CAD) tools for integrated circuits (IC's) that dramatically increase engineering productivity, many of these tools are inapplicable to aiding the design of microelectromechanical systems (MEMS). The most important requirement for a MEMS designer is that the CAD system be able to analyze realistic three-dimensional structures whose geometry is either provided directly by the designer, or is derived from simulation of the fabrication process using a process flow and mask description. Being able to analyze structures whose geometry is specified directly is obviously useful for examining alternatives early in the design process, but the ability to use the actual design specification, that is, the process flow and masks, to derive a structure makes the CAD system much more powerful. With such a tool, a MEMS designer can verify process descriptions and masks before beginning fabrication; insure a robust design by examining performance sensitivities to process and mask variation, including the very important dependence of material properties on process conditions; and isolate experimental artifacts from real phenomena by correlating measured data with simulated results.

At Transducers '87 [1], we identified the need for a new kind of CAD system for microelectromechanical devices and systems, which we now call MEMCAD. Our vision is that eventually it should be possible for a MEMS product designer with an idea of what is needed to interact with two other experts, one on MEMCAD and the other on process technology, to create a practical design for a new product. In the absence of MEMCAD tools, the opportunity for advances in the MEMS technology is limited to fabrication experts who often must work in cut-and-try fashion to create new structures that perform as originally conceived.

Since 1987, several groups, including our own, have reported progress in this area [2]–[9]. Koppelman, in his OYSTER program, has concentrated on the problem of creating a 3-D solid geometric model from an integrated-circuit process description and mask data [2]. Buser [7], Koide [8], and Séquin [9] have also focused on 3-D geometric modeling, but specifically for silicon anisotropic etching. Crary [5] and Zhang [6] have concentrated on developing an overall user interface which can access modules for simulating the mechanical behavior of specific structures, such as diaphragms. Our approach to the MEMCAD problem has been both at the system level, seeking to define an architecture which could, in principle, address the entire coupled set of CAD problems, and also at the specific tool-development level.

A schematic block diagram of our MEMCAD system

Manuscript received August 12, 1991; revised October 28, 1991. Subject Editor H. Fujita. This work was supported in part by the Defense Advanced Research Projects Agency under Contracts MDA972-88-K-0008 and N00014-87-K-825, a National Science Foundation Presidential Young Investigator award, the Department of Justice under Contract J-FBI-88-067, and a grant from Digital Equipment Corporation.

S. D. Senturia and R. M. Harris are with the Microsystems Technology Laboratories, Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Cambridge, MA 02139.

B. P. Johnson is with the Power Products Division, Systems Technology Division, IBM Corp., Endicott, NY, and the Microsystems Technology Laboratories, Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Cambridge, MA 02139.

S. Kim, K. Nabors, and J. K. White are with the Research Laboratory of Electronics, Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Cambridge, MA 02139.

M. A. Shulman was with the Microsystems Technology Laboratories, Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Cambridge, MA. He is now with the Microsoft Corp., Redmond, WA 98052.

IEEE Log Number 9105426.

1057-7157/92$03.00 © 1992 IEEE

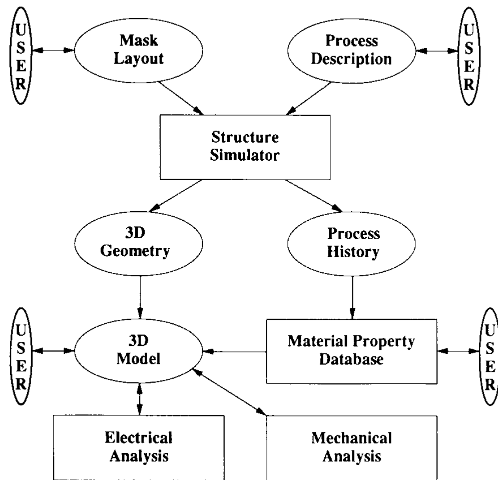


Fig. 1. MEMCAD architecture.

is shown in Fig. 1. The basic idea is that starting with a mask layout and process description, a 3-D solid geometric model is constructed by the *Structure Simulator*. For purposes of early design investigation the 3-D structure may also be entered directly. Then, depending on the process history, the material properties appropriate to each component of the structure are retrieved from a *Material Property Database* and added to the geometric model, thereby creating a complete 3-D solid model of the structure. This structure can then be analyzed for electrical properties and mechanical behavior with suitable modeling tools. In the next section we examine the MEMCAD architecture in more detail, and then describe our approach to geometric construction in Section III. In Sections IV and V, the material property database and the specialized numerical analysis techniques are described. Examples using the system to perform electrical and mechanical analyses are presented in Section VI, and conclusions in Section VII.

## II. MEMCAD System Architecture

### A. Architectural Constraints

We have imposed three important constraints on the MEMCAD architecture and implementation:

*1) Work from the actual design.* It is essential that the designer be able to simulate and verify the *actual design*, i.e., the actual mask set and process flow. In IC design, for example, simulation and design verification tools routinely accept the actual mask set and/or process flow as inputs. However, IC CAD generally does not provide tools for automatic generation of three-dimensional models which are needed for subsequent mechanical and electrical analysis. Hence, the structure simulator link between the IC layout-and-process world and the three-dimensional model is an essential MEMCAD feature. Further, since the material properties depend on the specific process sequence, a process-dependent material-property database is required for simulation of actual designs.

*2) Use well-supported commercial software when available.* The mechanical CAD area is rich in tools which provide support for solid modeling, user-driven mesh generation for solid models, numerical simulation of interior problems with finite-element modeling (FEM), and three-dimensional visualization. It is neither necessary nor desirable to recreate such modules when they are available and effective. Therefore, whenever possible, the MEMCAD implementation has relied on linking together suitable commercial software packages. Commercial packages must be chosen carefully, as they can impact the entire architecture. For example, a solid modeler imposes its own constraints on the underlying data representations for the solid model and may limit the modeling primitives available for creating new shapes. This obviously impacts the Structure Simulator but also influences the numerical tools which perform the analyses, as they must now be compatible with the solid-model representation and with the mesh types which can be generated.

*3) Provide fast and easily used analysis.* Simulation of three-dimensional MEMS typically involves solving not just for quantities in the interior of the structure, like displacement due to applied loads, but also for quantities in the exterior of the structure, like electric fields. It is possible to use FEM-based methods to analyze both interior and exterior problems. However, for exterior problems, volume meshes with large numbers of nodes must be created, which is burdensome to the user, and then these large meshes must be solved, which is computationally very expensive. To avoid these problems, accelerated boundary-element methods (BEM) and mixed FEM–BEM methods, which require only surface and interior meshes, are being developed for the MEMCAD system (see Section V). In light of these constraints, it is interesting to compare the MEMCAD system with other approaches to CAD for MEMS currently under way. Koppelman's OYSTER program functions as a structure simulator; it does an excellent job of creating a 3-D shape from a mask set and description of the effects of various process steps. Koppelman selected a boundary representation for his solid model. However, despite recent progress, generation of a satisfactory 3-D volume mesh from a boundary representation is still an open problem [10]. The MEMCAD approach is to create a volume representation directly, for which meshing is straightforward [11], but model construction may be difficult. Crary's CAEMEMS framework appears to provide a highly effective user interface for linking between modules which perform specialized tasks, such as simulating the mechanics of a diaphragm, and is attempting to address the complex problem of a material property database within that framework. However, lacking the equivalent of a structure simulator, the CAEMEMS framework can only deal with structures for which specialized modules have been developed. Our MEMCAD system is attempting to provide both a structure simulator for geometric flexibility, and effective links to general three-dimensional analysis tools, all within the same environment.

### B. Implementation Decisions

There are many specific implementation decisions that must be made early in a system development, and which have significant impact on how the system develops. In the case of the MIT MEMCAD system, portability and compatibility with existing integrated circuit CAD tools required that the system be developed under the Unix operating system and use the X11 Window System. Mask layout is created in CIF format [12] using KIC [13]. Process sequences are specified in the MIT-developed process flow representation (PFR), a hierarchical process language which integrates all of the information about a process step needed for design, simulation, manufacturing, and scheduling [14]. Parsers for the PFR have been developed which can create a linear list either of the *treatments* to which the device is subjected during fabrication (such as times, temperatures, gas environments, etc.), or of the *process effects* (such as amount of oxide grown, etc.). A simulation manager [15] is used to create an input file for the process simulators (SUPREM-III, SAMPLE) from the process PFR. The resulting process simulator output becomes input for the Structure Simulator, which is a special module being developed for MEMCAD (see Section III). It combines mask and process information to create the 3-D geometry, and also creates a process history linked to this geometry which is used to access the material property database (Section IV).

The primary interface for mechanical and electrical modeling, in the present version of MEMCAD, is through PATRAN [16], a mechanical CAD package which provides interactive construction of 3-D solid models, meshing of the models, application of loads and boundary conditions, graphical display, and interfaces to FEM packages (we are currently using ABAQUS [17]). While there are limitations in the implementation of boolean construction operators in the solid modeling primitives supported by PATRAN, the underlying data representations have proved both accessible and useful. Specifically, the 3-D geometry resides in the PATRAN neutral file which uses a specific volume representation, analytic solid modeling [18]. This neutral file format is also used for insertion of material property information from the material property database based on the process history (Section IV), and to provide an interchange format between the mechanical and electrical analysis modules (Section V).

It must be stressed that the MEMCAD architecture permits substitution of other software modules, subject to the overall architectural constraints. For example, other FEM packages, such as ANSYS [19] or NASTRAN [20] could be used. It is also true that other mechanical CAD packages could be used in place of PATRAN. However, a substitution here impacts every other module because of the central role played by the solid-modeling primitives and data representations of the mechanical CAD module. Current efforts to develop standardized TCAD frameworks, particularly a standard semiconductor wafer representation, should make the substitution of a new solid modeler much easier [21], [22]. We are currently examining alternatives to PATRAN which have more versatile solid-modeling capabilities. This paper will deal only with our PATRAN-based system, and its present status and performance.

## III. Structure Simulator

### A. Design

The Structure Simulator must merge the mask layout and process information to construct a three-dimensional solid model [4]. Two kinds of information must be tracked, the *geometry* of the structure (position, shape, and connectivity of each component), and the *material type* and associated process conditions used to create each component of the structure. The material type and associated process conditions for each component are stored in the process history file. The 3-D geometry is made available for subsequent analysis by creating a PATRAN neutral file. The 3-D geometry is later combined with material properties from the material property database based on the process history to form a complete 3-D solid model.

The Structure Simulator is designed to work on a process-step-by-process-step basis, modifying the solid model after each process step. This mimics the physical fabrication sequence, in which each process step causes a change in the wafer. The final structure is therefore the result of a sequence of such changes. The step-by-step operation of the Structure Simulator also permits simulation of intermediate results of the process sequence. This can be used, for example, to simulate whether the structure will maintain mechanical integrity throughout the entire process sequence.

For each process step, the Structure Simulator begins by reading the current step from the PFR, and the effect of this step is determined. (For example, the process *step* "oxidize in wet $O_2$ for 4 h at 1000°C" is converted into the process *effect* "grow 1 $\mu$m silicon dioxide on bare ⟨100⟩ silicon.") When appropriate, the PFR information is passed to process simulators (e.g., SUPREM-III, SAMPLE) and the simulation results are used to determine the process effect.

The process effect is then decomposed into construction and selection operators. The construction operators are responsible for actually modifying the solid model from the previous step, both the geometry (in the neutral file), and the material-type information (in the process history file). For microelectromechanical design, the following construction operators constitute a useful minimal set: film deposition and growth, film etching, impurity introduction and movement, and wafer joining. These construction operators are combined into a complete process effect using selection operators that restrict the operation of a construction operator to a certain region of the solid model. Selection may be done on the basis of layout (masking) or material type. The updated model is output for use with the next process step. The above sequence is

TABLE I
SUMMARY OF STRUCTURE SIMULATOR OPERATION

1) Determine effect of process step.
   Consult layout information and process modelers, as necessary.
2) Decompose process effect into construction and selection operators.
3) Modify solid model using construction operators.
4) Output results
   Geometry information ⇒ Neutral File
   Material information ⇒ History File
5) Repeat steps 1-4 for next process step.

then repeated for the next process step. The operation of the Structure Simulator is summarized in Table I.

Modification of the geometry must be implemented to ensure that the resulting model is physically valid (i.e., describing a reasonable approximation to the actual structure). Without careful attention to the robustness of the modification algorithms, invalid solid models could result (e.g., an unphysical topology such as a Klein bottle, or two objects occupying the same place). Implementation of a robust modification algorithm is considerably simplified if the modification is done using a small set of robust construction operators. Additionally, using a small set of construction and selection operators insulates the Structure Simulator from the effects of changing the underlying solid modeler.

### *B. Present Status*

A simple Structure Simulator capable of handling a single mask layer with a single feature has been reported previously [4]. We are currently extending the structure simulator to handle multiple multifeatured masks.

Multifeatured masks present two problems. First, the CIF description of the mask may contain abutting or overlapping features. These are not permitted in the solid model. Therefore, the CIF file must be converted into a *planar map* in which each group of abutting and overlapping features in the CIF description have been merged into a single feature. We are using the "quad-edge" data structure of Guibas and Stolfi [23] to represent the planar maps.

Secondly, each mask (represented as a planar map) must be overlaid on the existing surface (which can also be represented as a planar map). The mask overlay problem is equivalent to finding the intersections between two sets of segments, $S$ and $T$, where no two $S$-segments and no two $T$-segments intersect. An $S$-segment and $T$-segment, however, may intersect. The set of edges bounding the faces in each map correspond to the sets $S$ and $T$. The overlaid faces are bounded by $S$- and $T$-segments with new vertices at $S$-$T$ intersections. We are using an extension of an $S$-$T$ intersection reporting algorithm due to Mairson and Stolfi [24] to do the overlay.

## IV. THE MATERIAL PROPERTY DATABASE

### *A. Design*

The problem in establishing a Material Property Database (MPD) for microelectronic materials is that the properties of these materials, particularly thin-film materials, may depend significantly on the detailed process used to form or deposit them (see, for example, the literature cited in [1]). In our previous work on this subject [3], we reported on the idea of using an object-oriented database to capture the process dependence of material properties. A brief description of the first implementation of the database was presented at Transducers '91 [25], including the rationale behind its design, and the specific schema being used to implement the object-oriented features. The following is based on that conference report.

An object-oriented database (in contrast to a record-oriented database, which is like a spreadsheet with a set of predefined rows and columns) can be thought of as a looseleaf notebook which is divided into sections, each of which can be repeatedly (i.e., recursively) divided into subsections. The entries in the notebook, i.e., the data, can be in any format which fits on a page, e.g., numbers, graphs, functions, references, etc. The advantage of an object-oriented structure is that the form of the data for each entry can be tailored to fit the actual physical behavior being described. For example, it is known that the residual stress of silicon nitride depends dramatically on the processing conditions, but the density does not. Thus, a single number entry in a spreadsheet might be suitable for the density, but a multivariable function expressing the dependence on process variables (gas pressure, flow rate, substrate temperature, etc.) is needed for residual stress. Implementation of the simultaneous storage of these two quantities in a record-oriented format would be extremely cumbersome.

In an object-oriented architecture, the same data structures, or schema, can be used regardless of the specific data type, as long as the type can be recognized and evaluated by the user interface. Fig. 2 illustrates the schema being used in the MPD. Fig. 3 illustrates a sample user request for data on the density of a PECVD oxide. A key feature of the design is that the user can select a default value at the desired level of detail. At the top menu level, the user selects a material (silicon dioxide) and a property (density). The user interface then requests more specific information on the material. If the user "defaults," the database selects amorphous silicon dioxide grown as a thermal oxide with a set of default oxidation parameters (time, temperature, ambient). The database then returns the density for the default selection of material type, process, and process parameters. However, if the user is interested in PECVD, "amorphous" is specified, and when asked for a process, the user selects PECVD (had the default been selected, the thermal oxidation path would have been used as shown in the figure). The user then has a choice of either accepting the default PECVD parameters (gas flow, pressure, temperature, power), or entering specific values. Regardless of which path is used, the database returns to the user 1) the specific data, 2) an indication of whether this data came from interpolating within the experimentally documented range, extrapolating outside that range, or as a default value, and 3) a reference

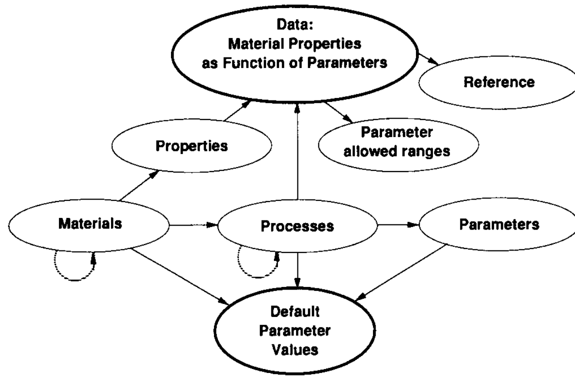


Fig. 2. Schema for the object-oriented Material Property Database. The arrows indicate pointers; the dashed loops indicate recursive subdivision.

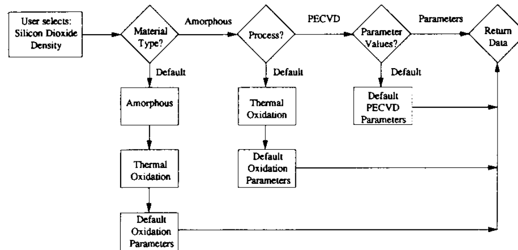


Fig. 3. Illustration of a typical user interaction with the Material Property Database to find the density of PECVD silicon dioxide. If at any point, the user selects a default, the indicated pathway to the Return Data block is followed.

citation to permit the user to track the validity and applicability of the data.

### *B. Implementation*

These schema have been implemented in the MIT-developed GESTALT program [26], which provides an object-oriented interface to an underlying relational database, in this case INGRES [27]. Implemented data structures include single numbers, arrays and tables, and—most useful—algebraic functions which can capture the process-dependence of data. The user interface includes an algebraic-function evaluator borrowed from GIRAPHE [28], so that when a functional form is used for the data, the user is always presented with a specific result. The user is also provided documentation of the data source.

We now present an illustrative example of how something as complex as the temperature and gas-flow dependence of the residual stress in silicon nitride can be represented as a single data object. Fig. 4 is a graph of data on the residual stress extracted from the literature [29]. Also shown are solid lines obtained from a linear regression of the following function to the data:

$$s = 10.7r^2 + 1.22r - 0.219rT - 1.86T + 2260 \quad (1)$$

where $s$ is the residual stress (MPa), $T$ is the temperature (Celsius), and $r$ is the ratio of dichlorosilane to ammonia. It is seen that (1) is a reasonable representation for the

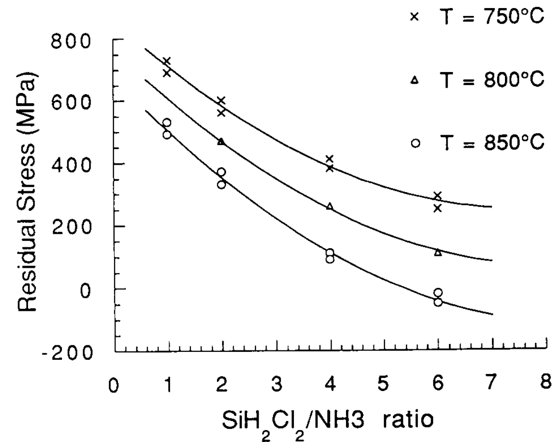


Fig. 4. Stress versus process conditions for silicon nitride. The points are from [29]. Solid lines are the fit of (1).

data, and thus provides an interpolation function for the entire parameter space spanned by $750 < T < 850$ and $1 < r < 6$. As long as the user requests processing parameters within this space, the expression is considered valid.

When the user queries the database for nitride residual stress, this entire expression is returned as the data object, along with the reference. The user is then asked to provide values for the parameters. If the parameters are within the interpolation space, the expression evaluator substitutes these values and returns a numerical result. If the user does not provide a parameter value, midrange parameter values are used to provide a default stress value.

The database is also constructed to allow limited extrapolation outside the range of parameters rigorously supported by the interpolation function. At this time, we do not know whether this feature will prove useful for device design; however, it is expected to be useful for estimating the property values needed in the design of new experiments to find material properties in regions of the process-parameter space not previously explored.

### *C. Present Status*

A text-based user interface is complete, both for data entry and data retrieval. Improvements, however, are needed to permit efficient display of the full tree of materials and properties without excessive menu overhead. A programmatic interface has also been implemented which permits the database to be accessed as a function call from another program. Johnson [30] has recently implemented a module, which based on the process history (see Fig. 1), automatically accesses the database and inserts the material properties into the 3-D solid model.

Top-level default data on a set of seven materials (silicon, polysilicon, aluminum, aluminum oxide, silicon dioxide, silicon nitride, and polyimide), and seven properties (density, dielectric constant, index of refraction, Young's modulus, Poisson ratio, thermal expansion coefficient, and residual stress) have been installed. Work

continues on collecting process-dependent data from the literature for a set of microelectronic materials, and placing each material/process combination within the appropriate tree structures. As a result of this literature review, we have identified two problems which are not addressed by the present schema: how to represent morphology, and how to record and track the effects of subsequent thermal processing, such as annealing. We are exploring adding an attribute to the schema on morphology (such as grain size, texture, and, for a single crystal, orientation). The annealing problem is more difficult, since the final state of a material depends on its complete thermal history. The concept of "wafer state," and changes in wafer state brought about by a process step, may prove useful in handling the annealing problem [31]. In the present implementation, subsequent process steps are ignored when accessing the database.

## V. Structure Analysis

Researchers in microsensors already investigate mechanical properties of microstructures by computer analysis, typically solving stress-strain and diffusion equations numerically using finite-element methods. For example, Christel [32] and Pourahmadi [33] have used FEM to model mechanical stresses and thermomechanical behavior of silicon microstructures. Also, Mullen [34] has used FEM to predict load-deflection and buckling behavior of beams with step-up boundary conditions. Pan [35] and Maseeh [36] have used FEM to perform quantitative studies of models of diaphragm structures used in an experiment to determine material mechanical properties.

Although researchers have succeeded in analyzing relatively simple microstructures using existing FEM-based software, this approach does not easily extend to performing simulations involving the combined effects of mechanical, electrical, and thermal stimuli. Simulating complete behavior of most microstructures requires solving not just for quantities in the interior of the structure (such as displacement) but also for exterior quantities (such as electric fields). If FEM-based methods are used for the exterior quantities, then an exterior volume mesh is needed. Although there are methods for generating *interior* volume meshes [11], volume meshes for the *exterior* are *not* readily generated. Even if a user is willing to construct the exterior mesh, such meshes require a large number of nodes, and the resulting system of equations would be computationally very expensive to solve.

Furthermore, to determine the electrostatic force on a structure, it is *not* necessary to compute electric fields throughout the entire exterior volume; only the field at the structure surface is relevant. This suggests that electrostatic force can be computed using boundary-element methods, in which only surfaces are discretized and only surface quantities computed [37]. The advantage of using BEM is that generating exterior volume meshes is unnecessary, and only relevant unknowns are computed. However, BEM can be used directly only to solve linear homogeneous problems; finite-element techniques are needed to treat inhomogeneous materials as well as nonlinearities. For the MIT MEMCAD system, we are developing specialized BEM and mixed finite-element/boundary-element methods, to make it possible to easily and efficiently simulate coupled mechanical and electrical effects.

### A. Mechanical Analysis

Mechanical analysis is performed in the MEMCAD system using ABAQUS, a commercially available finite-element based program [17]. The finite-element approach involves subdividing the domain of a structure to be analyzed into a number of elements, where the displacement over each element is approximated by shape functions. The coefficients of these shape functions are determined by solving force equilibrium equations with supplied boundary conditions. Details on the finite-element approach to mechanical system analysis can be found in many sources, for example in [38].

### B. Electrostatic Analysis

In the MIT MEMCAD system, electrostatic analysis is performed using the boundary-element based program FASTCAP [39]. Since this program is tailored to the MEMCAD system, it is described in some detail here. Specifically, in FASTCAP, computing capacitance and charge density, which is directly related to electrostatic force, is made tractable by assuming the conductors are ideal and are embedded in a piecewise-constant dielectric medium. To compute the surface charge densities, Laplace's equation is solved numerically over the charge free region with the conductors providing potential boundary conditions. One standard numerical approach is to apply a boundary-element technique to the integral form of Laplace's equation, as in [40],

$$\psi(x) = \int_{\text{surfaces}} G(x, x')\, \sigma(x')\, da' \qquad (2)$$

where $\sigma$ is the surface charge density, $da'$ is the incremental surface area, $\psi$ is the surface potential and is known, and $G(x, x')$ is the Green's function, which in free space is $1/\|x - x'\|$.

To numerically solve (2) for $\sigma$, the conductor surfaces are broken into $n$ small panels or tiles. It is assumed that on each panel $k$, a charge, $q_k$, is uniformly distributed. Then for each panel, an equation is written that relates the potential at the center of that $k$th panel, denoted $p_k$, to the sum of the contributions to that potential from the $n$ charge distributions on all $n$ panels [41]. The result is a dense linear system,

$$Pq = \bar{p} \qquad (3)$$

where $P$ is the matrix of potential coefficients, $q$ and $\bar{p}$ are the vectors of panel charges and given panel potentials, respectively, and

$$P_{kl} = \frac{1}{a_l} \int_{\text{panel}_l} \frac{1}{\|x' - x_k\|} da' \quad (4)$$

where $x_k$ is the center of the $k$th panel and $a_l$ is the area of the $l$th panel.

The dense linear system of (3) can be solved to compute panel charges from a given set of panel potentials, and the capacitances can be derived from the panel charges. If Gaussian elimination were used to solve (3), the number of operations would be order $n^3$. Clearly, this approach becomes computationally intractable if the number of panels exceeds several hundred implying only simple MEMS geometries could be analyzed.

In [39], a fast algorithm for computing the surface charge of three-dimensional structures of rectangular conductors in a homogeneous dielectric was presented. The computation time for the algorithm, which was based on the hierarchical multipole algorithm [42], was shown to grow nearly as $mn$, where $n$ is the number of panels used to discretize the conductor surfaces, and $m$ is the number of conductors. In [43], several improvements to that algorithm were described, including the extension to arbitrary geometries. In addition, a variety of examples were presented to demonstrate that the new method is accurate and can be as much as *two orders of magnitude faster* than standard direct factorization approaches. Incorporating the FASTCAP program based on the multipole-accelerated boundary-element approach into the MEMCAD system allows for very complicated MEMS geometries to be quickly analyzed on a standard scientific workstation.

### C. Electromechanical Analysis

When analyzing an electrostatically deformed structure, the mechanical and electrostatic problems must be solved self-consistently. Simply computing the electrostatic forces on the undeformed structure may not be accurate. In general, mechanical deformation of the structure will cause charge redistribution, and the electrostatic forces on the deformed structure will differ from those on the undeformed structure. This effect is particularly significant in electrostatically deformed microactuators, where without a self-consistent calculation pull-in cannot be simulated correctly.

We are currently investigating the following iteration to provide self-consistent electromechanical simulation. First, the electrostatic force is computed and used to deform the structure. Then the electrostatic force is recomputed on the deformed structure and this new force is used to redeform the structure and so forth. Results of this investigation will be reported separately.

## VI. Demonstration Examples

In this section, we present examples that demonstrate the current state of our MEMCAD system and show its ability to perform both electrical and mechanical analysis. Such a combined capability is essential when analyzing either sensors based on capacitance measurement [44] or microactuators positioned by electrostatic forces [45]–[49]. In particular, it is now possible for a designer to use the MEMCAD system running on a workstation to: specify and mesh an arbitrary three-dimensional solid model, specify boundary conditions and applied loads, query a Material Property Database for material property insertion, run a finite-element based mechanical simulation to determine solid model deformation due to specified loads, and then determine the capacitance of the mechanically deformed structure. The software links and specialized electrostatic solver make it easy for a MEMCAD user to compute an accurate capacitance versus pressure curve for a completely new microstructure geometry.

For the examples below, the commercial solid modeler, PATRAN, is used to capture the geometry of the structure and to impose finite-element meshing, boundary conditions and mechanical loads. This structure is represented in the PATRAN neutral file. A software module has been implemented to access the material property database [30], and insert the material properties directly into the neutral file. For the examples presented in this paper, the specific material properties used are the silicon material elastic constants (i.e., Young's modulus = 169 GPa and Poisson's ratio = 0.3). The FEM solver, ABAQUS, is run to calculate mechanical displacements, and then our electrostatic solver, FASTCAP, is used to calculate the resultant capacitance of the deformed structure. In order to facilitate this transfer between modules, careful attention to data formats and to the detailed meshes appropriate to the separate mechanical and electrical analyses was required. These issues are explained with the aid of the examples in the following sections.

### A. Electrical Analysis

In this section, results from computational experiments are presented to demonstrate the efficiency of the specialized preconditioned, adaptive, multipole-accelerated (PAMA) 3-D capacitance extraction algorithm used in the program FASTCAP, and incorporated into the MEMCAD system. The CPU times required to compute the capacitances of five different examples using the two methods are given in Table II. The examples Cube and Sphere are discretizations of a unit cube and a unit sphere; 5 × 5 woven bus is a 5 × 5 version of the 2 × 2 woven bus shown in Fig. 5. These structures were created using problem-specific programs prior to existence of the PATRAN/FASTCAP link. The example *via*, shown in Fig. 6, models a pair of connections between integrated circuit pins and a chip-carrier; and the example *diaphragm* is a deformed diaphragm (to be described below). These examples were created directly using PATRAN. From Table II, it can be seen that the PAMA algorithm is more than two orders of magnitude faster than direct methods for the larger problems, and that by using FASTCAP, the capacitance of *all* the examples can be computed in less than 20 min on an IBM RS6000 workstation.

TABLE II
CPU TIMES IN MINUTES ON AN IBM RS6000/540. TIMES IN PARENTHESES ARE EXTRAPOLATED USING THE $n^3$-GROWTH OF THE MATRIX SOLUTION TIME

| Method | Cube 294 Panels | Sphere 1200 Panels | Via 6185 Panels | Diaphragm 7488 Panels | 5 × 5 Woven Bus 9630 Panels |
|---|---|---|---|---|---|
| Direct | 0.08 | 4.4 | (490) | (890) | (1920) |
| PAMA | 0.03 | 0.4 | 3.1 | 3.0 | 12 |

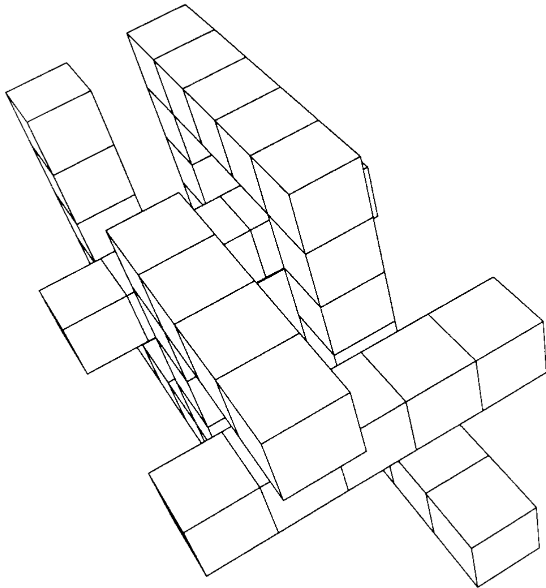

Fig. 5. The 2 × 2 woven bus problem: bars have 1 m × 1 m cross sections.

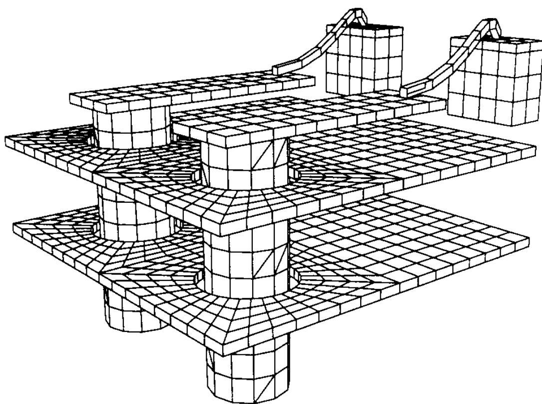

Fig. 6. Two signal lines passing through conducting planes; via centers are 2 mm apart.

### B. Square Diaphragm Capacitance

The next example is a square diaphragm deflected by differential pressure [30]. Deflections due to various pressures can be measured as capacitance changes between the diaphragm and a ground plane. The square diaphragm presents a severe test for the capacitance calculation of the MEMCAD system because the deformation due to pressure can be made sufficiently large to force the capacitor plates to touch. In this example, the initial separation between the diaphragm and ground electrode is 1 $\mu$m with a diaphragm edge length of 1000 $\mu$m and a diaphragm thickness of 5 $\mu$m. At a pressure of 1.5 kPa, the diaphragm separation reduces to 0.02 $\mu$m in the center of the diaphragm geometry. With the gap reduced to 2% of its original value, we are still able to obtain adequate precision in both mechanical and electrical behavior, provided appropriate meshing is used. This particular diaphragm touches down at a load of 1.53 kPa.

Due to symmetry, only a quarter of the diaphragm geometry is meshed. ABAQUS is run to find the diaphragm deformation due to the pressure loads imposed. The FEM output and 3-D solid model of the diaphragm are then passed to FASTCAP to find the resultant capacitance in the presence of the applied pressure. FASTCAP requires the entire geometry of the diaphragm and ground plane to be meshed, but this can be done efficiently within PATRAN. Four-node shell elements were used for both the modeling of the mechanical deformation and the modeling of the capacitance calculation to simplify the analysis. This is not a fundamental limitation of the MEMCAD system, but the data compatibility between FASTCAP and ABAQUS is more complex when solid elements are used for the FEM analysis.

A variety of elements and mesh sizes were used to study convergence for the diaphragm geometry. Lin performed an element and mesh analysis on the square diaphragm and reported a mesh of 256 ratioed shell four-node elements was sufficient to calculate mechanical deformations [50]. The ratioed elements, see Fig. 7, are meshed so that there are a larger number of elements at the edge of the diaphragm as compared to the number of elements in the center of the diaphragm.

We found that these ratioed elements do not model accurately the capacitance calculation of the square diaphragm structure when the deflection is large, as shown in Table III. The 256 ratioed mesh calculates a capacitance value that is smaller by 11% then the value calculated by smaller mesh sizes. The reason is that when the deflection is large, the capacitance is dominated by the diaphragm center, which has too few elements to capture the curvature of the diaphram relative to the gap thickness. The last two row entries in Table III show calculations based on a double-ratioed mesh (see Fig. 7). This mesh allows a larger number of elements to be placed both in the center of the diaphragm which dominates the ca-

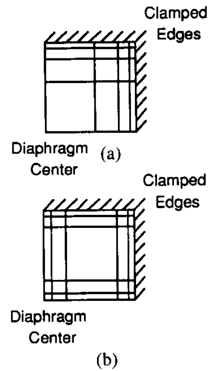


Fig. 7. Schematic drawings of (a) ratioed elements and (b) double ratioed elements over a quarter of the square diaphragm geometry [30].

TABLE III
MESH STUDY FOR THE SQUARE DIAPHRAGM AT AN APPLIED PRESSURE OF 1.5 kPa, FOR WHICH THE GAP IS REDUCED FROM 1.0 TO 0.02 $\mu$m [30]

| Number of Elements for ABAQUS | Center Displacement ($\mu$m) | C (pF) |
|---|---|---|
| 256 Ratioed | 0.980 | 19.40 |
| 625 | 0.980 | 21.56 |
| 900 | 0.980 | 21.91 |
| 1225 | 0.980 | 21.88 |
| 400 D. Ratioed | 0.980 | 21.85 |
| 900 D. Ratioed | 0.980 | 21.92 |

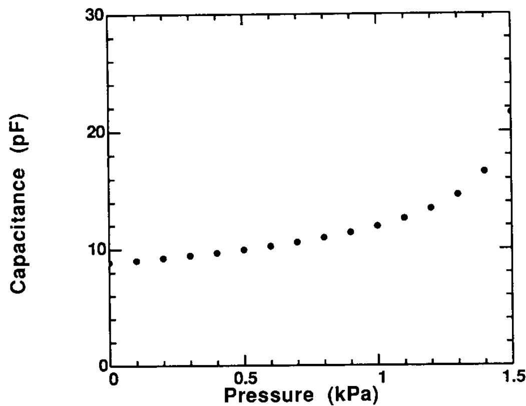


Fig. 8. Simulated capacitance versus pressure of a square diaphragm [30]. Points have been simulated using ABAQUS to determine mechanical deformation and FASTCAP to determine capacitance of deformed structure.

pacitance calculation, and at the edge of the diaphragm, to account for mechanical deformation.

Based on the meshing study, 900 four-node shell elements were used to calculate the capacitance-pressure curve of Fig. 8. Computer times for ABAQUS and FASTCAP on a Sun4 platform were 6.0 and 18.6 min for this example, respectively, at the 1.5 kPa data point. A comparison of the diaphragm deflections simulated using ABAQUS to a Fourier series calculation [51] of the deflections agree to within 1.0% for pressures which deflect the beam to 98% of the gap.

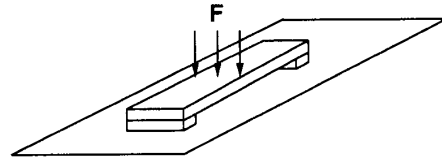


Fig. 9. Beam structure showing applied force for capacitance calculation. The relative permittivity of the supports is assumed to be 1.0.

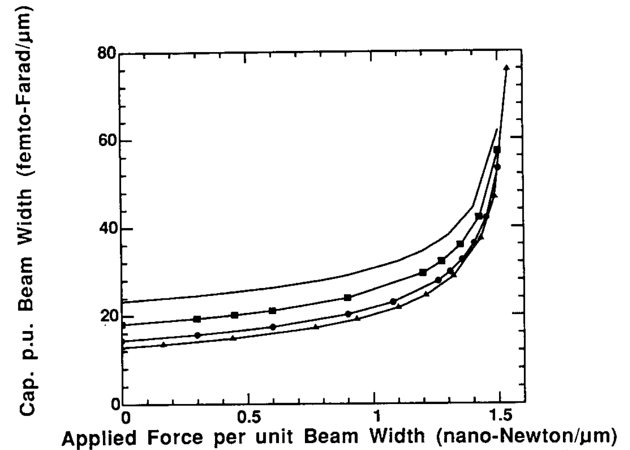


Fig. 10. Simulated capacitance per unit width versus applied force per unit width for rectangular beams; length = 1200 $\mu$m, thickness = 1.0 $\mu$m, and electrode gap = 1.0 $\mu$m [30]. Solid line (——), squares (■), circles (●), and triangles (▲) are for beamwidths of 1-, 2-, 5-, and 10-$\mu$m, respectively.

### *C. Beam Microstructure Capacitance*

The last example is a beam that is clamped on both ends and suspended 1 $\mu$m above a ground plane [30]. A force is applied at the center of the beam across the width perpendicular to the beam's length (see Fig. 9). Such doubly clamped beam structures have been used to measure material properties of thin films [46]. Four-node shell elements were used for mechanical and electrical simulations with mesh sizes of 400 elements. Capacitance per unit beamwidth is calculated for changes in applied force per unit beamwidth, and is shown in Fig. 10. The beamwidths are scaled so that the effects of fringing electric fields become important as the beamwidth decreases, hence the increase in capacitance per unit width for small applied forces. There is an 81% increase in capacitance per unit width at small loads between beamwidths of 10 and 1 $\mu$m. Because FASTCAP is so fast, analyses of this type, where fringing fields become important, can now be done routinely as part of a design.

## VII. CONCLUSION

We have presented an overview of the MIT MEMCAD system, the philosophy behind its architectural design, the

status of its present implementation, and several examples illustrating its capabilities. It is an evolving system, with a long term goal of providing complete MEMS design capabilities within one workstation environment. The system is currently limited in its ability to generate 3-D geometries directly from mask sets, a situation that will improve when the implementation of the Mairson–Stolfi intersection algorithm is completed. For the present, when supplemented by direct input of the 3-D geometry, the MEMCAD system now has capabilities for querying the material property database for material property insertion and performing both mechanical analysis (with FEM) and electrical analysis (with FASTCAP). The rectangular beam example, in particular, demonstrates the utility of the MEMCAD system in analyzing the electrical behavior of mechanically deformed structures in cases where 3-D fringing fields are important.

At present, MEMCAD is being extensively used within MIT for the design and analysis of microelectromechanical devices. Provision for access to users outside MIT is currently under consideration. The FASTCAP module has already been released for general distribution. Release of other MEMCAD modules will take place when they reach stable and documented configurations.

## Acknowledgment

Brian P. Johnson is in the Resident Study Program with the IBM Corporation in Endicott, NY. The authors are grateful for the assistance of several people. Duane Boning critically reviewed early versions of this manuscript. Malini Ramaswamy, Laura McDonald, and Aleksandar Pfajfer assisted in the collection of data for the Material Property Database. Michael Heytens provided assistance with the GESTALT object-oriented database. Leonidas Guibas suggested the use of the quad-edge data structure and the Mairson-Stolfi intersection algorithm. The authors also gratefully acknowledge technical discussions with Fariborz Maseeh on the MEMCAD architecture.

## References

[1] S. D. Senturia, "Microfabricated structures for the measurement of mechanical properties and adhesion of thin films," in *Proc. 4th Int. Conf. Solid-State Sensors and Actuators (Transducers '87)*, Tokyo, 1987, pp. 11-16.

[2] G. M. Koppelman, "OYSTER, a three-dimensional structural simulator for microelectromechanical design," *Sensors and Actuators*, vol. 20, pp. 179-185, Nov. 1989.

[3] F. Maseeh, R. M. Harris, and S. D. Senturia, "A CAD architecture for microelectromechanical systems," in *Proc. IEEE Micro Electro Mech. Syst.*, Napa Valley, CA, 1990, pp. 44-49.

[4] R. M. Harris, F. Maseeh, and S. D. Senturia, "Automatic generation of a 3-D solid model of a microfabricated structure," in *Tech. Dig. IEEE Solid-State Sensor and Actuator Workshop*, Hilton Head, SC, June 1990, pp. 36-41.

[5] S. B. Crary and Y. Zhang, "CAEMEMS: An integrated computer-aided engineering workbench for micro-electro-mechanical systems," in *Proc. IEEE Micro Electro Mech. Syst.*, Napa Valley, CA, Feb. 1990, pp. 113-114.

[6] Y. Zhang, S. B. Crary, and K. D. Wise, "Pressure sensor design and simulation using the CAEMEMS-D module," in *Tech. Dig. IEEE Solid-State Sensor and Actuator Workshop*, Hilton Head, SC, June 1990, pp. 32-35.

[7] R. A. Buser and N. F. de Rooij, "CAD for silicon anisotropic etching," in *Proc. IEEE Micro Electro Mech. Syst.*, Napa Valley, CA, Feb. 1990, pp. 111-112.

[8] A. Koide, K. Sato, and S. Tanaka, "Simulation of two-dimensional etch profile of silicon during orientation-dependent anisotropic etching," in *Proc. IEEE Micro Electro Mech. Syst.*, Nara, Japan, Feb. 1991, pp. 216-220.

[9] C. H. Séquin, "Computer simulation of anisotropic crystal etching," in *Proc. 6th Int. Conf. Solid-State Sensors and Actuators (Transducers '91)*, San Francisco, CA, June 1991, pp. 801-806.

[10] D. C. Cole, E. M. Buturla, S. S. Furkay, K. Varahramyan, J. Slinkman, J. A. Mandelman, D. P. Foty, O. Bula, A. W. Strong, J. W. Park, T. D. Linton Jr., J. B. Johnson, M. V. Fischetti, S. E. Laux, P. E. Cottrel, H. G. Lustig, F. Pileggi, and D. Katcoff, "The use of simulation in semiconductor technology development," *Solid State Electron.*, vol. 33, pp. 591-623, 1990.

[11] P. Conti, N. Hitschfeld, and W. Fichtner, "Omega—An octree-based mixed element grid allocator for the simulation of complex 3-D device structures," *IEEE Trans. Computer-Aided Design*, vol. 10, pp. 1231-1241, Oct. 1991.

[12] L. Conway and C. Mead, *Introduction to VLSI Systems*. Reading, MA: Addison-Wesley, 1980, pp. 115-127.

[13] G. C. Billingsley, "Program reference for KIC," Tech. Rep. UCB/ERL M83/62, Univ. California, Berkeley, CA, Oct. 1983.

[14] M. B. McIlrath and D. S. Boning, "Integrating semiconductor process design and manufacture using a unified process flow representation," in *Proc. 1990 Int. Conf. Comput. Integrated Manuf.*, May 1990.

[15] D. S. Boning, "Semiconductor process design: Representations, tools, and methodologies," Ph.D. dissertation, Massachusetts Inst. Technol., Cambridge, MA, Jan. 1991.

[16] PDA Engineering, Costa Mesa, CA.

[17] Hibbit, Karlsson, and Sorensen, Inc., Providence, R.I.

[18] M. S. Casale and E. L. Stanton, "An overview of analytic solid modeling," *IEEE Comput. Graphics and Appl.*, vol. 5, no. 2, pp. 45-56, 1985.

[19] Swanson Analysis Syst., Inc., Houston, PA.

[20] McNeal-Schwendler Corp., Los Angeles, CA 90041.

[21] A. Wong, W. Dietrich, and M. Karasick, Eds., "Semiconductor wafer representation architecture," Doc. 162, CAD Framework Initiative, Austin, TX, 1991.

[22] A. Wong, Ed., "Semiconductor wafer representation procedural interface," Doc. 166, CAD Framework Initiative, Austin, TX, 1991.

[23] L. Guibas and J. Stolfi, "Primitives for the manipulation of general subdivisions and Vornoi diagrams," *ACM Trans. Graph.*, vol. 4, pp. 74-123, Apr. 1985.

[24] H. G. Mairson and J. Stolfi, "Reporting and counting intersections between two sets of line segments," in *Theoretical Foundations of Computer Graphics and CAD*, R. A. Earnshaw, Ed. New York: Springer-Verlag, 1987.

[25] M. Shulman, M. Ramaswamy, M. Heytens, and S. D. Senturia, "An object-oriented material-property database architecture for microelectromechanical CAD," in *Proc. 6th Int. Conf. Solid-State Sensors and Actuators (Transducers '91)*, San Francisco, CA, June 1991, pp. 486-489.

[26] M. L. Heytens and R. S. Nikhil, "GESTALT: An expressive database programming system," *ACM SIGMOD Rec.*, vol. 18, pp. 54-67, Mar. 1989.

[27] M. Stonebraker, G. Wong, P. Kreps, and G. Held, "The design and implementation of INGRES," *ACM Trans. Database Syst.*, vol. 1, no. 3, pp. 189-222, 1976.

[28] R. M. Harris and D. S. Boning, "Giraphe v3.3: A user's manual with examples," VLSI Memo 88-486, Massachusetts Inst. Technol., Cambridge, MA, Nov. 1988.

[29] M. Sekimoto, H. Yoshihara, and T. Ohkubo, "Silicon nitride single-layer X-ray mask," *J. Vac. Sci. Tech.*, vol. 21, p. 1017, 1982.

[30] B. P. Johnson, S. Kim, J. K. White, and S. D. Senturia, "MEMCAD capacitance calculations for mechanically deformed square diaphragm and beam microstructures," in *Proc. 6th Int. Conf. Solid-State Sensors and Actuators (Transducers '91)*, San Francisco, CA, June 1991, pp. 494-497.

[31] D. S. Boning, M. B. McIlrath, P. Penfield, Jr., and E. M. Sachs, "A general semiconductor process modeling framework," submitted to *IEEE Trans. Semiconductor Manuf.*

[32] L. Christel, K. Peterson, P. Barth, F. Pourahmadi, and J. Mallon, "Single-crystal silicon pressure sensors with 500X overpressure protection," in *Proc. 5th Int. Conf. Solid-State Sensors and Actuators (Transducers '89)*, Montreux, Switzerland, June 1989, pp. 84-88.

[33] F. Pourahmadi, P. Barth, and K. Peterson, "Modeling of thermal and mechanical stresses in silicon microstructures," in *Proc. 5th Int. Conf. on Solid-State Sensors and Actuators (Transducers '89)*, Montreux, Switzerland, June 1989, pp. 84-88.

[34] R. Mullen, M. Mehregany, M. Omar, and W. Ko, "Theoretical modeling of boundary conditions in microfabricated beams," in *Proc. IEEE Micro Electro Mech. Syst.*, Nara, Japan, Feb. 1991, pp. 154-159.

[35] J. Pan, P. Lin, F. Maseeh, and S. D. Senturia, "Verification of FEM analysis of load-deflected methods for measuring mechanical properties of thin films," in *Tech. Dig. IEEE Solid-State Sensor and Actuator Workshop*, Hilton Head, SC, June 1990, pp. 70-73.

[36] F. Maseeh, M. Schmidt, M. Allen, and S. D. Senturia, "Calibrated measurements of elastic limit, modulus, and the residual stress of thin films using micromachined suspended structures," in *Tech. Dig. IEEE Solid-State Sensor and Actuator Workshop*, Hilton Head, SC, June 1990, pp. 84-87.

[37] C. A. Brebbia and J. Dominguez, *Boundary Elements: An Introductory Course*. Southampton: Computational Mechanics Publications, 1989. (Co-published with McGraw-Hill, New York.)

[38] K. J. Bathe, *Finite Element Procedures in Engineering Analysis*. Englewood Cliffs, NJ: Prentice-Hall, 1982.

[39] K. Nabors and J. White, "FastCap: A multipole-accelerated 3-D capacitance extraction program," to appear in *IEEE Trans. Computer-Aided Design*.

[40] A. Ruehli and P. A. Brennen, "Efficient capacitance calculations for three-dimensional multiconductor systems," *IEEE Trans. Microwave Theory Tech.*, vol. MTT-21, pp. 76-82, Feb. 1973.

[41] R. F. Harrington, *Field Computation by Moment Methods*. New York: Macmillan, 1968.

[42] L. Greengard, *The Rapid Evaluation of Potential Fields in Particle Systems*. Cambridge, MA: MIT Press, 1988.

[43] K. Nabors, S. Kim, J. White, and S. D. Senturia, "Fast capacitance extraction of general three-dimensional structures," in *Proc. Int. Conf. Comput. Design*, Cambridge, MA, Oct. 1991.

[44] W. H. Ko, "Solid-state capacitive pressure transducers," *Sensors and Actuators*, vol. 10, pp. 303-320, 1986.

[45] M. A. Huff, M. S. Mettner, T. A. Lober, and M. A. Schmidt, "A pressure balanced electrostatically-actuated microvalve," in *Tech. Dig. IEEE Solid-State Sensor and Actuator Workshop*, Hilton Head, SC, June 1990, pp. 123-127.

[46] K. Najafi and K. Suzuki, "A novel technique and structure for the measurement of intrinsic stress and Young's modulus of thin film," in *Proc. IEEE Micro Electro Mech. Syst.*, Salt Lake City, UT, Feb. 1989, pp. 96-97.

[47] R. T. Howe and R. S. Muller, "Resonant-microbridge vapor sensor," *IEEE Trans. Electron Devices*, vol. ED-33, pp. 499-506, 1986.

[48] L. S. Fan, Y. C. Tai, and R. S. Muller, "IC-processed electrostatic micromotors," in *Proc. IEDM*, San Francisco, CA, 1988, pp. 666-669.

[49] M. Mehregany, S. Bart, L. S. Tavrow, J. H. Lang, S. D. Senturia, and M. F. Schlecht, "A study of three microfabricated variable capacitance motors," *Sensors and Actuators*, vol. A21-A23, pp. 173-179, 1990.

[50] P. Lin, "The in-situ measurement of mechanical properties of multilayer coatings," Ph.D. dissertation, Massachusetts Inst. Technol., Cambridge, MA, 1990.

[51] C. Wang, *Applied Elasticity*. New York: McGraw-Hill, 1953, pp. 286-291.

**Stephen D. Senturia** (M'77-SM'91) received the B.A. degree from Harvard University, Cambridge, MA, and the Ph.D. degree from the Massachusetts Institute of Technology, Cambridge, both in physics, in 1961 and 1966, respectively.

He is the Barton L. Weller Professor of Electrical Engineering at the Massachusetts Institute of Technology. He has been active in applying novel measurement methods to a variety of fundamental and applied problems, including semiconductor physics, high-precision temperature measurement, automated solid-waste reclamation, oil-well logging, and microdielectrometry, which uses a silicon microsensor for in-process dielectric measurements of cure reactions of resins. His most recent work includes the use of micromachined structures for the determination of mechanical properties and adhesion of thin polymer films and coatings, and the development of silicon micromotors.

Dr. Senturia is an Editor of the JOURNAL OF MICROELECTROMECHANICAL SYSTEMS, an Associate Editor of the IEEE TRANSACTIONS ON ELECTRON DEVICES, the author or co-author of about 130 scientific publications, and is co-author of an elementary electronics text. He is the founder of Micromet Instruments, Inc., and serves on its Board of Directors.

**Robert M. Harris** (S'90) was born in Minot, ND, on July 20, 1960. He received the S.B. and S.M. degrees in electrical engineering and computer science from the Massachusetts Institute of Technology, Cambridge, in 1982 and 1986, respectively. He is currently a Ph.D. candidate at the Massachusetts Institute of Technology.

His research interests are in the design, modeling, and simulation of silicon microfabricated structures.

**Brian P. Johnson** (S'83-M'85) was born in El Paso, TX, in 1961. He received the B.S.E.E. and M.S.E.E. degrees from Purdue University, West LaFayette, IN, in 1984 and 1985, respectively. Since 1989, he has been in the IBM Resident Study Program working toward the Ph.D. degree at the Massachusetts Institute of Technology, Cambridge.

He joined the IBM Corporation, Lexington, KY, in 1985. He is currently a Staff Engineer in the Power Products Group, Endicott, NY, in the Systems Technology Division. His current work at MIT is in the electrical and mechanical modeling of microsensors and microactuators. His research interests are in the design, packaging, and electrical/mechanical simulation of energy conversion electronics.

**Songmin Kim** was born in Seoul, Korea, on April 19, 1963. He received the S.B. degree in mathematics and the S.B. and S.M. degrees in electrical engineering from the Massachusetts Institute of Technology, Cambridge, in 1986 and 1990.

Since 1990, he has been a Research Assistant in the Department of Electrical Engineering at MIT. His research interests are in the areas of electromagnetic theory and numerical methods applied to the calculation of three-dimensional capacitances and inductances.

Mr. Kim is a member of Phi Beta Kappa, Tau Beta Pi, and Sigma Xi.

**Keith Nabors** received the B.E.S. degree from the Johns Hopkins University, Baltimore, MD, and the S.M. degree from the Massachusetts Institute of Technology, Cambridge. He is currently working toward the Ph.D. degree at the Massachusetts Institute of Technology.

His current interests are in numerical methods for circuit parameter extraction.

**Matthew A. Shulman** received the B.A. degree in physics and chemistry from Harvard University, Cambridge, MA, in 1991. He is currently working as a software design engineer at Microsoft Corporation, Redmond, WA.

**Jacob White** (S'80-M'83) received the B.S. degree in electrical engineering and computer science from the Massachusetts Institute of Technology, Cambridge, and the S.M. and Ph.D. degree in electrical engineering and computer science from the University of California, Berkeley.

He worked at the IBM T. J. Watson Research Center from 1985 to 1987, was the Analog Devices Career Development Assistant Professor at the Massachusetts Institute of Technology from 1987 to 1989, and is a 1988 Presidential Young Investigator. He is currently an Associate Professor at MIT. His research focuses on theoretical and practical aspects of serial and parallel numerical methods for problems in circuit design and fabrication.