Proceedings of
The 1996 ASME Design Engineering Technical Conference and
Computers in Engineering Conference
August 18-22, 1996, Irvine, California

96-DETC/DFM-1312

DESIGN OF MEMS VIA EFFICIENT SIMULATION OF FABRICATION

Ted J. Hubbard
Department of Mechanical Engineering
Technical University of Nova Scotia
Halifax, Novia Scotia, Canada B3J 2X4
hubbardt@tuns.ca

Erik K. Antonsson[1]
Engineering Design Research Laboratory
California Institute of Technology
Pasadena, California 91125
erik@design.caltech.edu

ABSTRACT

A novel model is introduced to simulate the fabrication of MEMS (Micro Electro Mechanical Systems). This approach provides computationally efficient and geometrically accurate simulation results by hybridizing a geometrical and cellular approach. Existing simulation methods, including numerical and analytical models, are examined. Factors influencing simulation results, such as fabrication conditions and etching regimes, are introduced and discussed. Simulation examples are presented and non-ideal effects are considered.

INTRODUCTION

The development of new sensors and actuators to be fabricated by silicon micromachining techniques requires many prototypes and much experimentation. At this time, the methods to transform a (desired) 3-D etched shape into a required mask-layout are largely *ad hoc*, particularly for high aspect-ratio devices. Brysek, Petersen, and McCulley recently observed (1994):

> "Little simulation of micromachining processes is available." (Bryzek, Petersen, and McCulley, 1994, Page 25)

Since the customary and desirable etchants are highly anisotropic, the transformation from mask to shape is geometrically complex.

> "In-depth knowledge of the [fabrication] process is needed because in micromachining 'what you see' is often NOT 'what you get'." (Bryzek et al., 1994, Page 25)

The primary method of creating a MEMS (Micro Electro Mechanical Systems) mask-layout is trial-and-error, guided by experience. This process produces many prototypes, is not robust to variations in the fabrication process, and provides only informal guidance on the design of the next generations of MEMS.

Prior work has shown that simulations can be developed for

- common MEMS fabrication processes (Buser and de Rooij, 1991; DeLapierre, 1992; Frank and Ives, 1960; Fruhauf, Trautman, Wittig, and Zeilke, 1993; Hubbard and Antonsson, 1994b; Sequin, 1991),
- finite element analysis (Buser, Crary, and Juma, 1992; Crary and Zhang, 1990; Maseeh, Harris, and Senturia, 1990; Pourahmadi and Twerdok, 1990; Schwarzenbach, Korvink, Roos, Sartoris, and Anderheggen, 1993; Senturia, Harris, Johnson, Kim, Shulman, and White, 1992; Zhang, Crary, and Wise, 1990),
- corner compensation (Abu-Zeid, 1984; Offeriens, Kuhl, and Sandmaier, 1991; Puers and Sansen, 1990),
- and design (Kota, Ananthasuresh, Crary, and Wise, 1994; Trimmer and Gabriel, 1987; Wise, 1991; Wise and Najafi, 1991),

and can reduce the amount of experimentation and the number of prototypes required. Previous fabrication simulation research includes:

[1] Corresponding author

- ASEP (Anisotropic Silicon Etching Program) which can predict the output shape based on traveling planes (Buser et al., 1992; Buser and de Rooij, 1991),
- the Slowness Method (Sequin, 1991),
- the E-Shape method (Hubbard and Antonsson, 1994b),
- and a Cellular Automata method (Than and Buttgenbach, 1994; Hubbard and Antonsson, 1994a).

These methods each have significant limitations, which will be discussed below. ASEP, Slowness and E-Shape all utilize an analytical vector approach, based on planes or lines moving in response to etch rates. These approaches provide excellent spatial fidelity (they model complex shapes well). The Cellular Automata (CA) method subdivides the volume of substrate into many identical cells, introduces simple rules into each cell that govern its rate of removal when it is on a surface exposed to an etchant. This approach has limited spatial fidelity (non-orthogonal lines and planes are "jagged"), and consumes significant computation for even a modest discretization of a volume.

For the purposes of simulation, two different types of interactions exist: local and global. Both of these types must be able to be identified and located by a simulation for it to operate successfully. Local intersections arise as planes appear and disappear (and hence, intersect) at etched corners. Global interactions arise when two originally separate parts of a mask shape grow during etching and intersect (for example, two holes grow large enough to intersect).

Local interactions depend only on a line segment and its adjacent neighbors. For example, in Figure 1, segment ab has local interactions only. If an etched shape is represented by N line segments, then $2N$ calculations are necessary.

Global interactions on the other hand can in general involve any two (or more) segments and not just adjacent segments. For example, in Figure 1, region c has global interactions between two shapes. If an etched shape is represented by N line segments, then in the general case $N(N-1)$ intersection calculations must be done. For large N global interactions scale as N^2.

As the size and complexity of shapes and devices increase, the number of segments needed to model them also increases, and the global interactions become more important. Furthermore, many of the more interesting phenomena in MEMS (e.g. corner compensation, etc.) involve global intersections.

Analytical (vector) methods, such as ASEP, Slowness, and E-Shape are able to handle local intersections quite well, but incur significant computational costs to identify and locate global intersections (since they must test every polygonal line segment against every other segment). CA methods handle both local and global intersections without difficulty, but incur the computational costs mentioned above.

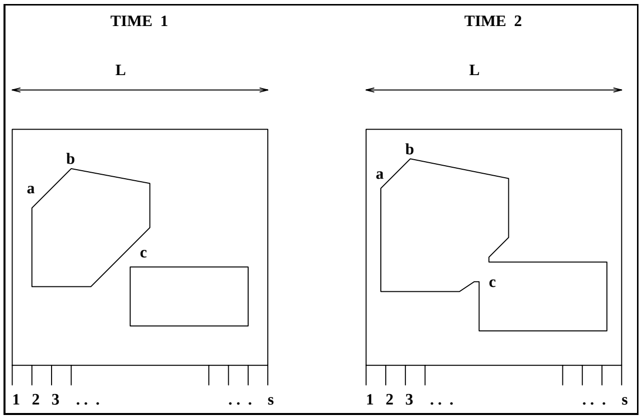

Figure 1. Shape evolution regimes.

Since the need for spatial accuracy may dictate a small cell size, and hence a large number of cells (perhaps approaching the number of atoms in the substrate), the CA methods in general will be computationally expensive and slow. Vector methods can be comparatively inexpensive but only if global intersections are identified and located manually.

Directed Segments Model In all models, the evolving etched shape is represented by a number of primitives: cells in the CA (Than and Buttgenbach, 1994; Hubbard and Antonsson, 1994a); traveling planes in ASEP (Buser et al., 1992; Buser and de Rooij, 1991); tangents in the Eshape (Hubbard and Antonsson, 1994b); vectors in the Slowness method (Sequin, 1991). The number of primitives varies with each model (high for CA, low for Eshape) as does the complexity of interactions (low for CA, high for Eshape). The level of representation used will determine both the accuracy and efficiency of any particular model. In order to incorporate the robustness and flexibility of an atomistic approach like the Cellular Automata, while retaining the speed and efficiency of models like the Eshape, the authors have developed a hybrid model, called the Directed Segments (or "Segs") method.

The "Segs" approach seeks to optimize the trade off between accuracy and efficiency. The model represents the shapes as a large number of small segments or facets (as in the CA), but they also retain geometrical information as in the Eshape method. This is analogous to finite element analysis which customarily uses elements that conform to the local shape of the object being modeled. The basic approach is to start with the polygonal boundary of a vector method, then subdivide each straight line segment into many smaller segments. This implements a spatially accurate discretization of the shape, but with many fewer elements than a CA method. Additionally, a rectangular grid of cells is superimposed on top of the line segments, to identify and locate global intersections. Because there are now a large number of primitives (segments plus geometry) the method is very robust and non-linear effects can be modeled. Since the primitives also include geometry information, many fewer primitives are needed to accurately model a shape. This greatly decreases the computational cost.

Approach

The Segs method proceeds by first computing local intersections, then proceeding to global.

The local, analytical calculations involve the two nearest neighbors and check the relative validity of adjacent line segments. A segment is valid if it lies in the still unetched half planes of its two neighbors. In terms of the local tangents and normal:

$$(n_i \cdot t_{i-1})(n_i \cdot \delta_{i,i-1}) \geq 0 \tag{1}$$

where n is the segment normal, t is the segment tangent, and δ is the segment position delta vector. The invalid segments are eliminated and the test is repeated for all the segments in the shapes.

The global, cellular calculations involve an overlaid grid. Each line segment location is written to the grid. Global interactions are easily detected when a grid cell that has been previously written into is overwritten.

Hybridizing the two methods decouples local and global interaction calculations and permits each to be optimized individually.

Algorithm for Segs simulator

The rate data is read in from a file as three-dimensional vectors each with an etch rate magnitude. The three dimensional rates are decomposed into a two dimensional etch rate in the plane of the surface of the wafer r_{2d}, and a lateral etch rate due to depth r_z. r_{2d} is the rate at which a plane moves in the mask plane. r_z is the lateral distance between the top or mask layer edge of a etch facet and the bottom edge of a etch facet.

The mask data is read in from a file as a set of polygons. The mask plane is divided into a grid or cell array of N by N cells. If the total length of the mask plane is L then each cell is L/N across. The mask polygons are then converted into lists of short segments of length L/N (generally shorter than the shortest side of the mask polygon(s)). Each element of the list contains an x, y location, the local slope θ (direction of the line in the plane of the surface of the wafer), the calculated local normal and tangent vectors. The elements are called directed line segments.

Two dimensional simulation: Local interactions: At each time step the new x, y position of each segment in the list is updated by adding a velocity vector in the direction of the local normal equal to the etch rate r_{2d} multiplied by the time step. Each segment i is compared to the neighboring segment $i+1$. Using dot products of the local normal, local tangent and the vector from segment i to segment $i+1$, the relative location of the segments is determined. If segment $i+1$ lies above the half plane defined by the local tangent of segment i then it is unetched, if it is below then it is etched and thus removed from the list. This process is repeated for all segments in the polygon list. Next the comparison is done in the opposite winding direction from segment i to segment $i-1$. This winding in both directions is repeated until the length of the polygon list ceases to decrease. If the length drops to zero, then the polygon is etched away.

Global interactions: At each time step, for each segment, the x, y position is converted into a cell location in the N by N grid. The polygon index and list index are then written to the cell array at this location. A global intersection occurs when two segments are written to the same cell. When this occurs, the intersecting polygon lists are cut at the intersections and rejoined into a new longer list. The parsed listed are then used as the input for the next time step.

Three dimensional simulation: The two dimensional simulation provides the top or mask layer edge of the etch facets. The bottom edge is found by using the parsed list at the end of each time step and updating the list with r_z instead of r_{2d}. For every unit time step, the etch increases by a depth d, with a corresponding lateral distance between the top and bottom edges of r_z. Thus from x, y positions at the top edge we subtract a velocity vector in the direction of the local normal equal to the etch rate r_z multiplied by the time step. Note that the bottom edge lies within the shape and thus the lateral etch rate is subtracted. The new list generated is then checked for local and global intersections as before. Multiple depth contours can be generated by repeating the above mentioned process multiple times.

Optimization

What is the optimal subdivision size for modeling cells? Large subdivisions result in a coarser approximation, while too fine a subdivision results in an inefficient algorithm. If the etching area is a square L units by L units and the etched shapes are modeled by N line segments with an average length l_{av}, the optimal subdivision size can be estimated. The use of an analytic expression to calculate the trajectories of line segments will have a computational cost. Let the cost in machine cycles be given by C_1, then the net cost of calculating global interactions analytically is N^2C_1.

Now consider the case for a large number of subdivisions s. If the cost of performing a logical intersection determination is given by C_2 and a length L requires s calculations, then the net cost for global interaction processing of etched shapes is $sC_2(Nl_{av}/L)$ where Nl_{av} is the total perimeter of the shapes. If s is too small, then false intersections will occur since two non-intersecting segments may fall within a very large cell. But too large an s will be too expensive. The transition s where the two approaches have equal cost is given by:

$$s_t = NC_{12}L/l_{av} \tag{2}$$

where C_{12} is the ratio of analytical computational costs to logical computational costs. The ratio C_{12} varies with implementation but is larger than 1, because logical calculations involve AND's, OR's, etc., and possibly additions, while analytic calculations involve multiplications, divisions, and possibly transcendentals.

Thus when the number of line segments is large and the segments are relatively short, a subdivision, or cellular approach is advantageous. However, the relative cost of each approach will vary with the shapes being etched. The Segs method is a flexible and variable model which permits hybridization of the two approaches.

Simulation Results

The hybrid simulator outlined above has been implemented in a portable computer program. Both the length of segments and the cell subdivision size are variable, so that resolution and efficiency can be traded-off. In most cases the subdivision length is several hundred. Simulation times depend on the complexity of the shapes, with typical simulation times on a SUN Microsystems Sparc workstation of tens of seconds. Shapes are input via CIF, GDSII, or a public domain drawing program (xfig), process (etch rate) information is input via text files, and output is available in several formats including PostScript, and IGES for subsequent 3-D solid modeling and finite element analysis. The integration of MEMS design tools and FEA packages (Senturia et al., 1992) is essential to modeling complex systems. Figure 2 shows an example Segs method simulation result. This figure shows the time evolution (each contour is a different etch time) of the 2-D intersection between the (growing) etched hole and the unetched surface of the wafer.

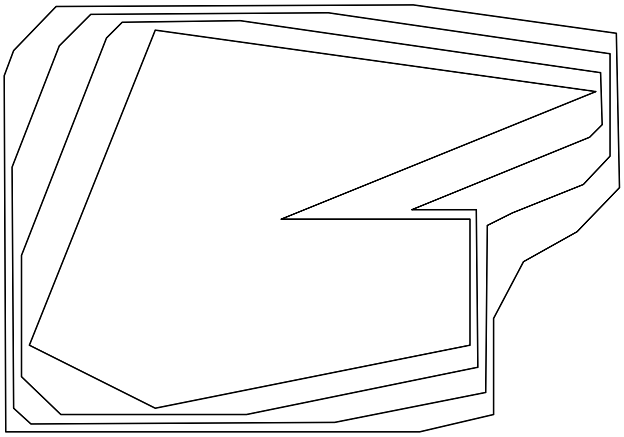

Figure 2. Two-Dimensional Example.

Non-Ideal Effects

Hybrid methods are well suited to modeling non-ideal effects, since the two levels of representation provide a choice as to modeling approaches. This is especially true because some non-ideal effects are small scale, while others are large scale. The smaller the scale at which the non-ideal effects occur, the more expensive the modeling since resolution must go up, and the maximum allowable grouping decreases. The optimum parameterization will depend on the particular phenomena being modeled. The following is a list of some non-ideal effects and possible modeling approaches.

- Multilayer/multi-process simulations are performed by concatenating several one-layer/one-process simulations. Such simulations tend to be large scale, that is no extra resolution is needed, rather, the extra cost comes from inter-layer interactions. The etching of upper layers propagates down to lower layers.
- Mask erosion is a special case of Multi-Layer. As well as etching the substrate, etchants also etch the mask, although both the rate and anisotropy may be different for the two materials. Because mask erosion is a large scale phenomena, high level modeling can be used. An additional mask layer with different etch characteristics is modeled prior to the substrate layer and any newly exposed regions are considered in the underlying substrate etch.
- Etchant aging is a special case of multiprocess with only time variation, and no additional inter-layer interactions. It is large scale so no extra resolution is needed.
- In etchant loading, the local concentration may vary, due to local consumption of etchant. Etching effectiveness decreases in small channels and confined spaces.

- In non-homogeneous etching, the local concentration of etchant may vary due to improper mixing. Both of the last two processes are small scale so resolution increase is necessary. Furthermore, modeling will be difficult to parameterize since the status of neighboring region and local fluid flows may have to be considered.

Accurate modeling of non-ideal effects is necessary to estimate fabrication yield and increase yields by designing robust, variation tolerant shapes.

Implementation in 3-D

The presentation above has described the approach utilized by the Segs method in two dimensions. The extension to three dimensions is straightforward, and also has been implemented computationally. While the relative scaling of the numerical and analytical approaches does not change since both increase by the same factor, the absolute cost does increase by a multiplicative factor of N.

Three dimensional shapes are modeled by using many layers each consisting of a two dimensional simulation. The topmost layer represents the mask. In the first etching step, the first layer is etched. In the second step, the first layer is etched again and then is used as a mask for the second layer. By propagating the changes from layer to lower layer, three dimensional shapes are formed as shown in Figures 3, 4, and 5. Figure 3 shows a time history of simulated etching, with each contour corresponding to a depth. In this Figure, both the 2-D shape of the etched hole at the surface of the wafer can be seen, along with the evolving 3-D shape of the hole. Figure 4 shows four time steps in the evolution of the etching of an initially round hole with a smaller masked round peg in the center (the left-hand element of the Figure). The next time-step is shown in the top element of the Figure, followed by the bottom element, and finally the right-hand element. As can be seen, the initially round mask hole evolves into a nearly square shape at the surface, with sloping sides. The initially round peg is considerably reduced in size, and has become an 8-faceted truncated pyramid. Figure 5 shows four time-steps in the evolution of a 4-pointed "star" into a rectangular peg with sloping sides. This figure illustrates the value of a 3-D simulation method, such as the Segs method reported here, to predict the required "pre-distortion" of a mask-layout to obtain a final desired 3-D shape. In this case, the four points of the "star" "protect" the four corners of the desired rectangular final shape. The size and shape of the protecting or compensating structures is highly dependent on etchant, etch time, desired shape, other nearby shapes, etc.

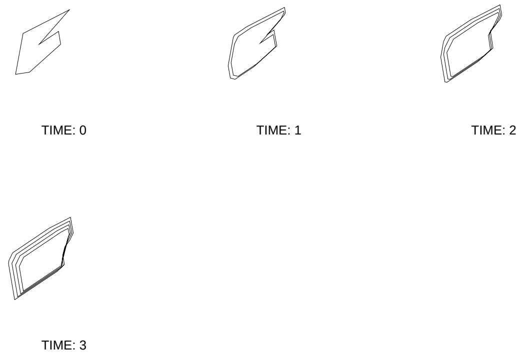

Figure 3. Three-Dimensional Example: Time Evolution.

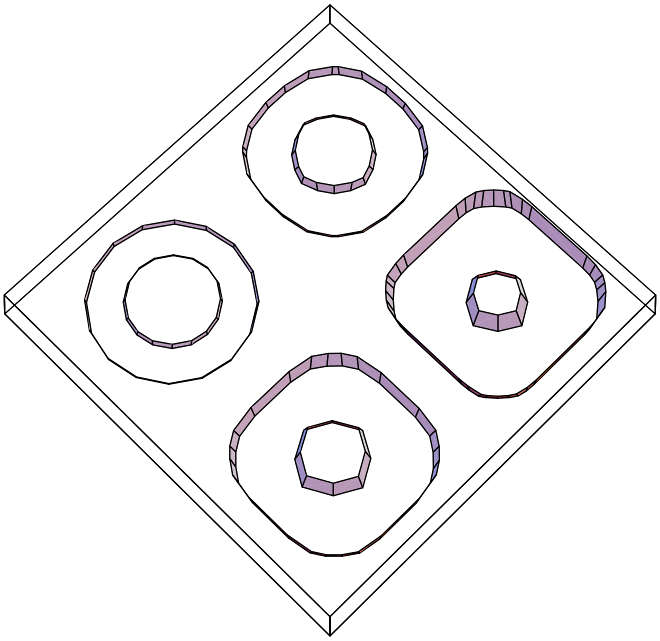

Figure 4. Three-Dimensional Example: Holes.

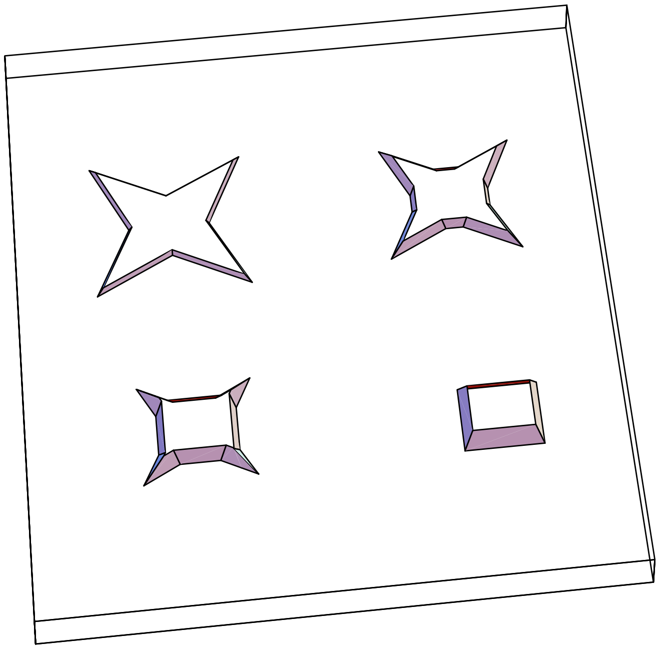

Figure 5. Three-Dimensional Example: Corner Compensation.

CONCLUSION

It is crucial to have accurate and efficient simulators of MEMS fabrication processes to reduce the design iteration cycle and thus reduce the number of required prototypes and ultimately the cost of MEMS development. Furthermore, complex systems will require design and simulation tools in order to accurately model both fabrication and performance. The Segs method for MEMS fabrication simulation described above provides such a tool.

Acknowledgments

The authors gratefully acknowledge Professor Y.C. Tai for the use of his facilities and many helpful discussions. This material is based upon work supported, in part, by the National Science Foundation under Grant No. ECS-9023646. Any opinions, findings, conclusions or recommendations expressed in this publication are those of the authors and do not necessarily reflect the views of the sponsor.

REFERENCES

Abu-Zeid, M., 1984, "Corner Undercutting in Anisotropically Etched Isolation Contours," *J. Electrochem. Soc.*, Vol. 131, pp. 2138–2142.

Bryzek, J., Petersen, K. E., and McCulley, W., 1994, "Micromachines Made of Silicon," *IEEE Spectrum*, Vol. 31, (No. 5), pp. 20–31.

Buser, R. A., Crary, S. B., and Juma, O. S., 1992, "Integration of the anisotropic-silicon-etching program ASEP within the CAEMEMS CAD/CAE framework," In *MEMS 92*, pp. 133–138 Institute of Electrical Engineers.

Buser, R. A., and de Rooij, N. F., 1991, "ASEP: A CAD program for silicon anisotropic etching," *Sensors and Actuators*, Vol. 28, pp. 71–78.

Crary, S. B., and Zhang, Y., 1990, "CAEMEMS: An intergrated computer-aided engineering workbench for micro-electro mechanical systems," In *MEMS 90*, pp. 113–115 Institute of Electrical Engineers.

DeLapierre, G., 1992, "Anisotropic crystal etching: A simulation program," *Sensors and Actuators*, Vol. 31, pp. 267–274.

Frank, F. C., and Ives, M. B., 1960, "Orientation-Dependent Dissolution of Germanium," *J. Appl. Phys.*, Vol. 31, (No. 11), pp. 1996–1999.

Fruhauf, J., Trautman, K., Wittig, J., and Zeilke, D., 1993, "A simulation tool for orientation dependent etching," *J. of Micromechanics and Microengineering*, Vol. 3, pp. 113–115.

Hubbard, T. J., and Antonsson, E. K., 1994a, "Cellular Automata in MEMS Design," Engineering design research laboratory report EDRL-TR 94d, California Institute of Technology.

Hubbard, T. J., and Antonsson, E. K., 1994b, "Emergent Faces in Crystal Etching," *Journal of Microelectomechanical Systems*, Vol. 3, (No. 1), pp. 19–28.

Kota, S., Ananthasuresh, G. K., Crary, S. B., and Wise, K. D., 1994, "Design and Fabrication of Microelectromechanical Systems," *ASME Journal of Mechanical Design*, Vol. 116, (No. 4), pp. 1081–1088.

Maseeh, F., Harris, R., and Senturia, S., 1990, "A CAD architecture for MEMS," In *Transducers 90*, pp. 44–49 Institute of Electrical Engineers.

Offeriens, H. L., Kuhl, K., and Sandmaier, H., 1991, "Methods for the Fabrication of Convex Corners in Anisotropic Etching of (100) Silicon in Aqueous KOH," *Sensors and Actuators*, Vol. A25-27, pp. 9–13.

Pourahmadi, F., and Twerdok, J., 1990, "Modeling Micromachined Sensors with Finite Elements," *Machine Design*, pp. 44–60.

Puers, B., and Sansen, W., 1990, “Compensation Structures for Convex Corner Micromachining in Silicon,” *Sensors and Actuators*, Vol. A21-23, pp. 1036–1041.

Schwarzenbach, H. U., Korvink, J. G., Roos, M., Sartoris, G., and Anderheggen, E., 1993, “A micro electro mechanical CAD extension for SESES,” *J. of Micromechanics and Microengineering*, Vol. 3, pp. 118–122.

Senturia, S. D., Harris, R. M., Johnson, B. P., Kim, S., Shulman, M. A., and White, J. K., 1992, “A computer-aided design system for microelectromechanical systems (MEMCAD),” *Journal of Microelectomechanical Systems*, Vol. 1, pp. 3–13.

Sequin, C. H., 1991, “Computer simulation of anisotropic crystal etching,” In *Transducers ’91*, pp. 801–806 San Francisco, CA, USA.

Than, O., and Buttgenbach, S., 1994, “Simulation of anisotropic chemical etching of crytalline silicon using a cellular automata model,” *Sensors and Actuators*, Vol. A45, pp. 85–89.

Trimmer, W. S., and Gabriel, K. J., 1987, “Design Considerations for a Practical Electrostatic Micro-motor,” *Sensors and Actuators*, Vol. 11, pp. 189–206.

Wise, K. D., 1991, “Integrated microelectromechanical systems: A perspective on MEMS in the 90s,” In *MEMS ’91*, pp. 33–27 Institute of Electrical Engineers.

Wise, K. D., and Najafi, K., 1991, “Microfabrication techniques for integrated sensors and microsystems,” *Science*, Vol. 25, pp. 1335–1342.

Zhang, Y., Crary, S. B., and Wise, K. D., 1990, “Pressure sensor design and simulation using the CAEMEMS-D module,” In *Tech. Dig. IEEE Solid-State Sensor and Actuator Workshop*, pp. 32–35 Institute of Electrical Engineers.