# The Intelligent Motion Surface: A hardware/software tool for the assembly of meso-scale devices[1]

Murilo G. Coutinho, Peter M. Will, IEEE Members,
and P. Selvan Viswanathan, IEEE Student Member
USC Information Sciences Institute
Marina del Rey, CA
(310)-822-1511
coutinho@isi.edu

**Abstract:** *The MicroElectroMechanical Systems (MEMS) field is proceeding at a rapid rate in designing and developing new ever smaller, ever higher performance sensor and actuator devices in silicon. These devices need to be assembled along with more conventional small devices into complete systems. Currently there are* ***no automation systems*** *capable of meeting the exacting requirements of handling these small and often fragile (until encapsulated) MEMS devices. This paper describes the latest results of our ongoing research on sensorless MEMS manipulator arrays. We use CMOS VLSI technology to construct automatic handling devices that scale with the technology curves of CMOS VLSI, and may offer a solution to the micro assembly problem. The Intelligent Motion Surface (IMS), itself a MEMS device, has been designed to meet this challenge. This paper also introduces the CILIA software, a high-level programmable simulation software available to the research community via the WWW. The CILIA software decouples the design of applications requiring the use of manipulator arrays from the actual devices used to implement them, therefore offering a level of abstraction for the application designer, who can implement a set of motion plans independent of the MEMS device actually being used. Also, the CILIA software can be used to simulate both macro and micro manipulator array devices.*

## 1. Introduction.

Microelectronics combined with micromechanics has enormous promise for delivering new functionality to future military and industrial products. These new microsystems will have sufficient novelty and computational power to allow many present-day bulky products to be miniaturized. These products span communications, portable devices, high precision equipment and medical monitoring among others. Many products with new functionality involve micro-mechanical devices as sensors and other active components.

The MicroElectroMechanical Systems (MEMS) field is proceeding at a rapid rate in designing and developing new, ever smaller, ever higher performance sensor and actuator devices in silicon. These devices need to be incorporated into packages (such as MCMs) and the packages into systems to be of use to the end user. The conventional die and package size has increased over the years but the new interest in portable devices is causing the issue of package size to be reconsidered: portable applications demand smaller packages designed to match the needs of small portable systems incorporating both small silicon chips and small MEMS devices. Currently there are **no automation systems** capable of meeting the exact requirements of handling these small and often fragile (until encapsulated) MEMS devices. The field is being termed *meso-scale robotics* to reflect its intermediate position between normal centimeter and micron sized devices.

In this paper, we describe the latest results of our ongoing research on sensorless MEMS manipulator arrays for the meso-scale automation problem. We use VLSI technology to construct automatic handling devices that scale with the technology curves of CMOS VLSI, and may offer a solution to the micro assembly problem. Our current chip designed in the MOSIS 2.0 micron scalable CMOS process was able to actually carry a part placed on its surface.

---

[1] *This work is sponsored by DARPA under Fort Huachuca Contract No. DABT-63-92-C-0052.*

The Intelligent Motion Surface (IMS) was originally introduced in [1], [2]. The IMS approach inverts the usual robotics device in that it puts all of the smarts in the feeder (the IMS) and reduces the conventional robot to a necessary, extremely fast, high accuracy, but relatively low dexterity device operating in conjunction with, but separated from, the IMS itself.

The use of micro manipulator arrays as programmable MEMS devices give the basis for a new class of precision positioning devices and precision feeding devices. The primary reference to moving parts on a surface is the patent application of Babinski from IBM [9], who managed to control semiconductor wafers on a surface pierced with appropriately designed holes through which air was pumped. Konishi and Fujita [6] also designed an air flow based conveyance micro motion device. Fujita el.al. [5] transported parts on a one dimensional array of polyamide "fingers". Other possible implementations of arrays of micro manipulators are also described in [4], [8]. Although several micro manipulator devices have been proposed, all but our current chip design presented in this paper, lack the ability to integrate both CMOS (for the control circuitry) and MEMS structures on the same chip (which is fundamental to implement the programmability property of an IMS).

Our IMS itself is built from an array of simple manipulators, each one capable of moving in one single direction. Different areas of the IMS are programmed to provide force and velocity vectors to coerce the carried part into performing the correct motions desired from the application viewpoint. That is, an appropriately chosen sequence of fields in the IMS (where in each field all the micro manipulators are coordinated to cause motion in a given direction), can produce desirable motions in the carried part. The fields may have arbitrary shape in the plane or may be null, i.e. have no field, or even have retracted robots. Suitable partitioning of the entire plane can cause the whole surface to behave in manners interesting for automation applications.

As an example, if we partition the plane into two regions, so that the left half moves parts to the right and the right part moves parts to the left, then a part placed anywhere on the surface in any orientation will be forced to move towards the centerline of the surface where it will align itself with the fields. The part will move until the forces impressed by the surface equalize in magnitude and are collinear. If collinearity is impossible, then the part will oscillate in a pattern defined by Newton's laws, friction and the magnitude and real shape of the IMS fields. Many field configurations have been explored, including fields for rotation, recognition, selection based on a part's shape, feeding one at a time, elementary mechanical logic including latching and pipelining, and the assembly of two parts into a sub-assembly [3].

This paper also introduces the CILIA software, a high-level programmable simulator that decouples the design of applications requiring the use of manipulator arrays from the actual devices used to implement them, and thus is a tool for the wide community of researchers in this sub-field of automation. The CILIA software offers a tool for the application designer, who can implement a set of motion plans independent of the MEMS device actually being used.

A demonstration-level software was also developed in [1], [2], showing the potential capabilities of the IMS to perform both simple (translating an object) and complex (sorting parts) motion plans. However, the CILIA software is far more advanced and robust, providing capabilities impossible to be achieved using the prototype software, such as being able to run on either standalone or client/server mode.

For movies of a real chip and simulations of an IMS prototype, see the project Web page at **http://www.isi.edu/mass.**

## 2. The IMS Hardware.

### 2.1. The First Generation IMS Prototypes.

The first generation IMS prototypes were made using magnetic flaps. The elementary micro manipulator is shown in Figure 1. This is a 400 micron permalloy coated flap made in Caltech as a collaborative effort with Prof. Tai of Caltech and Prof. Ho of UCLA. Please, refer to [3] for a comprehensive description of this prototype.

During the testing phase, this micro manipulator was able to lift 7 MOSIS chips, and is thus suitable for carrying silicon devices. The elementary device when used in synchronism with nearby robots allows the smooth flow of parts placed on the surface. For movies of a real chip and simulations of this first generation IMS prototype, see **http://www.isi.edu/mass**.

However, a viable IMS implementation requires both CMOS and MEMS structures integrated on the same chip. The permalloy-coated flap approach used on the first generation of prototypes produces acceptable forces, but because of the particular micromachining process, we cannot integrate the all-important CMOS structures required to control each single cell of the array of micro manipulators of the IMS from a user defined motion plan. This observation led us to develop the second generation of IMS prototypes based currently on a MOSIS scalable CMOS VLSI process using the two layers of metal followed by a XeF2 release etchant.

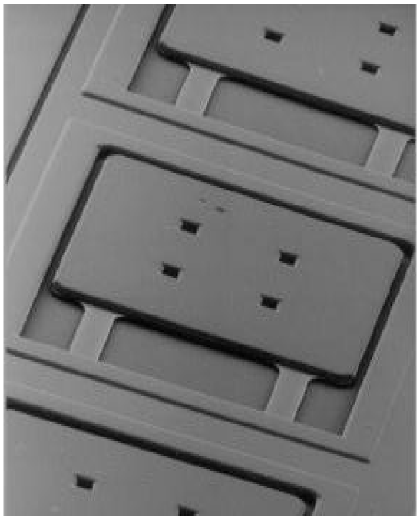


**Figure 1:** Single cell of permalloy magnetic actuator.

### 2.2. The Second Generation IMS Prototypes.

We have attempted to simplify the requirements of the elementary micro manipulators to the highest possible degree. The motivation was that extracting forces from MEMS actuators is a very difficult problem, and getting useful forces from motion off the surface (out of the plane) is extremely difficult.

In our research, therefore, we have sought to develop algorithms and software from which we could achieve complex motion behavior at the IMS level from the collective action of individual micro manipulators, each capable of only a simple, one degree-of-freedom motion.

The individual micro manipulators can be arranged in groups of four, each moving along one cardinal direction. This group of micro manipulators together with its associated CMOS control circuitry unit is defined in this paper as a MOXEL (motion control navigation cell). A schematic model of a MOXEL is shown in Figure 2.

The MOXEL can be replicated over the entire chip surface in typical VLSI fashion. Each micro manipulator of the MOXEL has a different motion direction selectable by the CMOS control unit in the cell. This control unit is capable of producing the desired motion direction from X-Y coordinates addressing within the cell array. The IMS can be covered with identical cells addressed the same way as in memory cells. A finite state machine (see Figure 3) controls the memory loading and the direction and phase (gait) in each cell to provide the programmed force field.

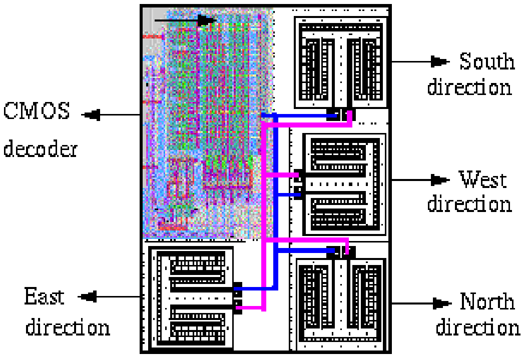
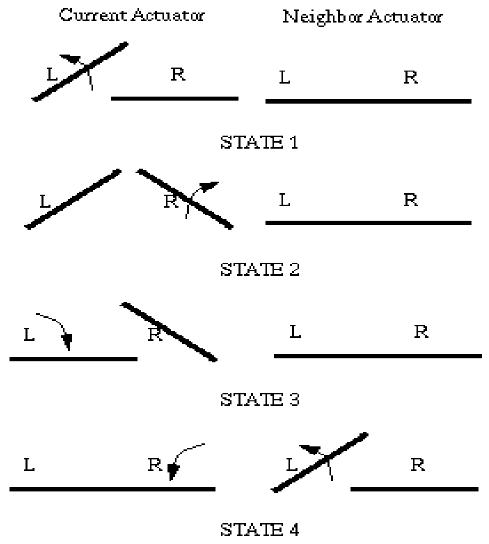


**Figure 2**: MOXEL with CMOS decoder and four unit robots, each capable of simple movements in just one cardinal direction.



| | STATE 1 | STATE 2 | STATE 3 | STATE 4 |
|---|---|---|---|---|
| L | UP | HOLD | DOWN | DOWN |
| R | DOWN | UP | HOLD | DOWN |

**Figure 3:** Simple finite-state machine used to coordinate the micro manipulators movement.

Our current chip designed in the MOSIS 2.0 micron scalable CMOS process is based on a simple Lorentz force actuation (see Figure 4) of the cantilevered structures.

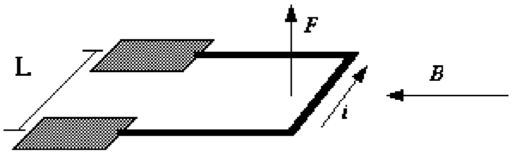


**Figure 4:** Principle of Lorentz force.

The Lorentz force actuation for each manipulator is given by:

$$F = B\,i\,L \tag{1}$$

where:

B = magnetic field being applied;
i = current being applied;
L = length of the tip of the actuator;

Figure 5 shows an SEM picture of our current experimental prototype fabricated. The individual micro manipulators showed a large amount of residual stress which curved the structures upwards. The devices were tested using thermal actuation as well as Lorentz forces.

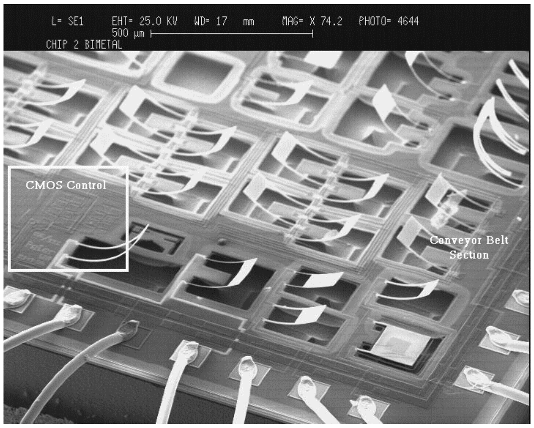


**Figure 5**: SEM picture of the experimental Lorentz force chip. The pairs of actuators ranging from top-left to bottom-right form a bi-directional conveyor belt. The CMOS control unit can be partially seen on the left side of the picture. Each actuator is 200 by 100 microns.

Our current chip actually managed to lift and carry a part on its conveyor belt, but due to the conveyor belt's limited width, the part was falling to one of its side after some steps. Better chips have been submitted to the MOSIS fabrication service.

## 3. The CILIA Software Tool.

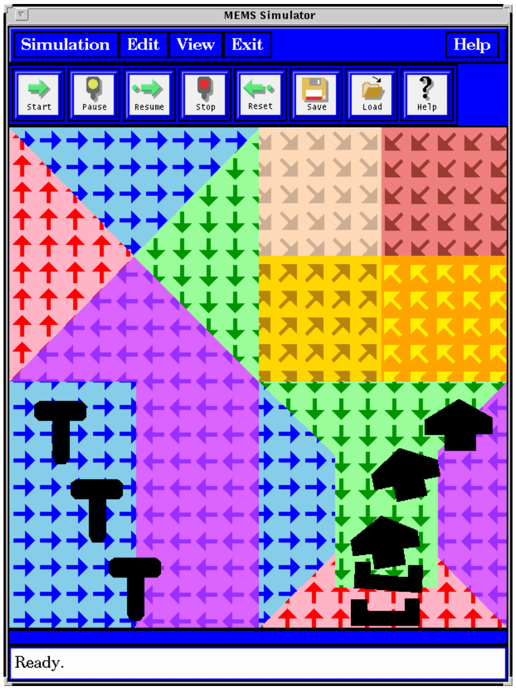
CILIA is a high-level programmable simulation software that decouples the design of applications requiring the use of manipulator arrays from the actual devices used to implement them. Therefore, it offers a level of abstraction for the application designer, who can implement a set of motion plans independent of the MEMS device actually being used. The CILIA software consists of:

- A Force Field Editor to construct force fields of general shapes;
- An Object Editor to construct objects of general complexity;
- A built-in library of geometric Boolean operations used by both editors;
- Unlimited Undo/Redo functionality;
- Flexible Save/Load operations: the user can save force fields only, objects only, or a complete configuration including both;

- A Simulation Engine that implements a rigid body dynamics model to simulate the motion plans involving the interaction of the objects placed on the IMS surface with the force fields defined. The graphical output of the simulation is shown on CILIA's Main Window (see Figure 6);
- An impulse-based collision model;
- User modifiable Simulation Parameters, including: static/dynamic friction coefficients, coefficient of restitution, simulation time step, among others;

The CILIA software can be executed in two different modes: standalone or client/server. The standalone mode was implemented for the SUN platform running the Sun OS operating system. It corresponds to the situation where the entire program is executed on a single machine. The client/server is used over the network. The server program was also implemented for the SUN OS platform, whereas the client program has two versions: a Motif version for the SUN OS platform (machine dependent, used mainly for development and testing purposes) and a Java applet version (machine independent, accessible from within any Java enabled Web browser). The Java applet has the same functionality of the Motif version, but because of the Java language security requirements, some of its functionalities have had to be disabled. For instance, save/load capabilities are disabled in the client Java applet.

CILIA is now available for the entire research community. The reader is invited to visit our Web page at **http://www.isi.edu/mass** and use CILIA the fullest. We plan to release improved versions on a timely basis, based on both our own research developments and user feedback comments and suggestions.



Ready.

**Figure 6**: Screen dump of the standalone version of the simulator software. Simultaneously: (upper-left) A rotating static force field; (upper-right) A centering static force field; (down-left) A part being aligned in the center with two static force fields; (down-right) The assembly of two parts into a sub-assembly, with careful choice of the static force field configuration and field strength.

### 3.1. Implementation Details.

Basically, the machine dependent version of the CILIA software is written in the C language and runs to about 100,000 lines of code at this stage. The GUI was implemented in two flavors: using Motif (machine dependent version), and using the Java AWT library (machine independent version). We had to re-write the entire GUI in Java for the machine independent version. The core functionality was kept on the server program.

CILIA implements its client/server using the thin client model. This model assumes the client machine to be simple and with almost none resources (easy maintainability). On the other hand, the model assumes a powerful host running the server program and a fast network connection. This model is most suitable for Intranets.

### 3.2. Basic Functionality.

CILIA comes with a system library that implements several complex geometric manipulation tasks. On both standalone and client/server version, CILIA includes both Force/Velocity Field editor and the Object editor. The user is able to draw, group, ungroup, cut, copy, paste, intersect, add, subtract, half plane cut, rotate, translate, change force field strength, classify the fields according to their priority levels, change priority levels, change the display status (making the force field windows visible or invisible to the user), save, load and edit, among the default set of editing functionality that is provided. All geometric Boolean operations implemented can handle general polygons with an unlimited number of holes inside.

CILIA also provides other more advanced features, such as unlimited undo/redo for both Field and Object editors, functions to change the simulation sampling time, functions to change the estimate of both static and dynamic friction coefficients of the IMS chip surface, and functions to change the density of the parts being simulated in order to allow the design of parts with the same size and shape but made of different materials. Multiple parts can be tested at the same time, and force field windows can be dynamically created, modified or destroyed in order to achieve complex tasks, such as moving a part among several moving obstacles. The latter will be a very common situation in the real IMS assembly line, where we want to optimally minimize the power consumption and at the same time maximize the use of the IMS surface by as many parts as possible (or needed).

The CILIA's simulator engine implements the simulation dynamics using a motion model based on rigid body and fluid dynamics [10], [12], [13]. The current model implements a macro-motion description that does not consider the micro manipulator arms motion details individually. That is, the simulations presented in this paper assume that the dimensions of the objects being manipulated are orders of magnitude greater than the dimensions of each micro manipulator.

The simulation of the dynamic interaction among rigid bodies being moved by the IMS takes into account various physical characteristics such as elasticity, friction, mass and moment of inertia. Collisions among objects moving on the surface are modeled using impulse-based techniques [11].

## 4. Conclusion.

In this paper we described the latest results on our ongoing research work on the development and implementation of an IMS. Our current chip actually managed to lift and carry a part on its conveyor belt, but due to the conveyor belt's limited width, the part was falling to one of its side after some steps.

This paper also introduced the CILIA software, a high-level programmable simulation software that offers a level of abstraction for the applications designer, who can now implement a set of motion plans independent of the MEMS device actually being used. CILIA is available to the research community via our Web page **http://www.isi.edu/mass.** The CILIA software can be used to simulate and design both micro and macro manipulator array devices, from an application viewpoint.

The first experimental results of the current chip are encouraging, and more tests concerning the reliability and endurance of the chips must be performed. These tests will appear in future publications.

## 5. Acknowledgments.

We would like to thank Mr. C. Liu and Mr. M. Pottenger for their work during the development and testing of the first-generation IMS prototypes. We also would like to thanks the reviewers for their useful comments.

## 6. References.

[1] P. Will and W. Liu, *Parts Manipulation on a MEMS Intelligent Motion Surface*, ISI Research Report - ISI/RR-94-391, Marina del Rey, CA, May 1994.
[2] W. Liu and P. Will, *Parts Manipulation on an Intelligent Motion Surface*, Human Robot Interaction and Cooperative Robots, Vol. 3, Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 399-404, Pittsburgh, PA, August 1995.
[3] W. Liu, P. Will, and M. Pottenger, *Modeling and Simulation of an Intelligent Motion Surface in MEMS*, Simulation and Design of Microsys-

tems and Microstructures, Proc. of the First International Conf. on Simulation and Design of Microsystems and Microstructures, pp. 311-320, Southampton, England, September 1995.

[4] C. Liu, T. Tsao, P. Will, Y. Tai, and W. Liu, *A micro-machined magnetic actuator array for microrobotics assembly systems*, in Transducers - Digest International Conf. on Solid-State Sensors and Actuators, Stockholm, Sweden, 1995.

[5] N. Takeshima and H. Fujita, *Design and Control of Systems with Microactuator Arrays*, Proc. IEEE Work. in Advanced Motion Control, pp.219-232, Yokohama, Japan, March 1990.

[6] S. Konishi, H. Fujita, *A Conveyance System using Air Flow bases on the Concept of Distributed micro motion systems*, Transducers - Digest International Conf. on Solid-State Sensors and Actuators, Yokohama, Japan, June 1995.

[7] K. Pister, R. Fearing and R. Howe, *A planar levitated electrostatic actuator system*, Proc. IEEE Microelectromechanical Systems Conf., Napa Valley, CA, pp. 67-71, Feb. 1990.

[8] K-F. Böhringer, B.R. Donald and N.C. MacDonald, *Single-Crystal Silicon Actuator arrays for micro manipulation tasks*, IEEE Proceedings of the Ninth Annual International Workshop on MEMS, San Diego, CA, pp 7-12, Feb. 1996.

[9] Babinski, JP et.al., US PATENT 3976330, August 24, 1976.

[10] J. K. Hahn, *Realistic Animation of Rigid Bodies*, Computer Graphics, pp.299-308, vol.22, number 4, August 1988.

[11] Brian V. Mirtich, *Impulse-based Dynamic Simulation of Rigid Body Systems*, Ph.D. thesis, Univ. of California at Berkeley, Fall 1996.

[12] Norman I. Badler et.al, *Making them Move: Mechanics, Control, and Animation of Articulated Figures*, Morgan Kaufmann Publishers, 1991.

[13] W.V. Tipping, *An Introduction to Mechanical Assembly*, Business Books Ltd., London, 1969.