Thermal simulation tools for microsystem elements

V. Székely*, M. Rencz✦, A. Páhi*, A. Poppe*, Sz. Hajder*

*Technical University of Budapest
Department of Electron Devices
1521 Budapest, Hungary
e-mail: szekely@eet.bme.hu

✦MicReD
Microelectronics Res & Dev Ltd.
Fax: (36 1) 246 17 70
e-mail: rencz@computer.org

Abstract

Two different field solver tools have been developed in order to facilitate fast thermal and electro-static simulation of microsystem elements. The μS-THERMANAL program is capable for the fast steady-state and dynamic simulation of suspended multilayered microsystem structures. The 2D-SUNRED program is the first version of a general field solver based on an original method, the successive node reduction. SUNRED offers a very fast and accurate substitute of FEM programs for the solution of the Poisson equation.

1. Introduction

Thermal and electro-static effects play fundamental role in the operation of many microsystem elements, as e.g. infrared sensors, thermal rms meters, capacitive displacement sensors, actuators based on the electrostatic force, etc. All these problems require fast and reliable field solver programs capable to solve the following equation:

$$p(x,y,z)+c\frac{\partial T}{\partial t}=\frac{\partial}{\partial x}\left(\lambda\frac{\partial T}{\partial x}\right)+\frac{\partial}{\partial y}\left(\lambda\frac{\partial T}{\partial y}\right)+\frac{\partial}{\partial z}\left(\lambda\frac{\partial T}{\partial z}\right) \qquad (1)$$

where λ is the heat conductivity, c is the heat capacitance per unit volume, $T(x,y,z)$ is the thermal field and $p(x,y,z)$ is the incoming heat flux density. FEM methods can always be used for this purpose, but the drawbacks are well known: first of all the huge amount of time required to define the problem and then the limitations in resolution determined by the mesh.

Two entirely different programs have been developed by the authors for the solution of Eq. (1) aimed at microsystem applications [1],[2]. These two programs are presented in this paper.

2. The μS-THERMANAL program

The μS-THERMANAL program has been developed for the thermal simulation of suspended microsystem elements as cantilevers, membranes, bridges, together with the investigation of the conventional IC chips [3].

The ancestor of this program, the THERMANAL program was an early realization of the well-known algorithm of Kokkas [4], with the extension for unlimited number of layers [5]. In the model of Kokkas the structure consists of equally shaped rectangular layers stacked on an ideal heat sink. The dissipating elements are on the surface of the uppermost layer only, and the heat is removed via the bottom surface, the sidewalls are adiabatic. Heat transfer is assumed only by conduction. This model is an appropriate one for conventional ICs, this explains that several realizations have been reported even recently.

In the algorithm of [4] the solution of the differential equation is constructed in the form of two-dimensional Fourier-cosine series. In our program this series is calculated by Fast Fourier Transformation (FFT) method – resulting in a quick solution both in steady-state and in the frequency-domain.

The Fourier algorithm relatively easily can be extended to calculate the temperature distribution on multi-layer structured membranes as well. In this case the boundary conditions are different from the usual IC structure: the top and the bottom surfaces of a membrane are adiabatic, while the sidewalls are isothermal, considering the bulk silicon nearly ideal heat sink.

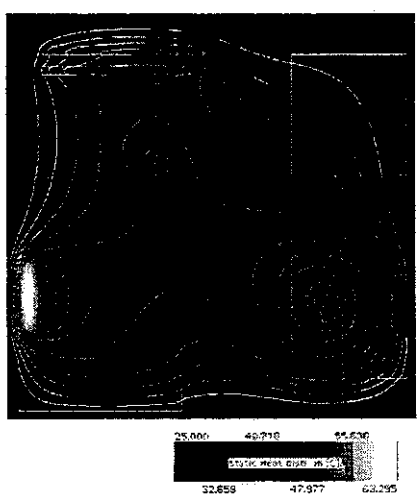

Fig.1. Steady-state temperature distribution on a membrane surface. The rectangular shapes are the dissipating elements.

These boundary conditions are fulfilled by using Fourier-*sine* expansion instead of Fourier-cosine [3]. Simulation results of a membrane structure are presented in Fig.1. Rectangular bridge and cantilever structures require the consideration of other, special boundary conditions. The Fourier method can be matched to these conditions as well, see [3].

We have extended this model by the consideration of free convection cooling, taking into account the heat transfer by convection on the top and bottom surface of the membrane. Moreover we modified the original algorithm also towards considering dissipators inside the

layered structure. The detailed description of these algorithms can be found in [3].

From heat transfer point of view a special class of microsystem elements are the parts suspended on thin and narrow strips. In these structures the leads act as heat sinking elements. In the μS-THERMANAL program we consider them as pseudo-dissipating elements with negative dissipation. The algorithm for the exact calculation of these dissipation values is given in [6].

3. SUNRED: a 2D thermal and electrostatic simulator

During our work in the thermal simulation of different microstructures we encountered the problem, that the accuracy of the applied FEM simulation tool had to be limited down to an unacceptable value if we wanted to have the simulation results in a reasonable time. To overcome this problem we have developed a new dedicated field solver program SUNRED, which works currently in two dimensions, but it is expandable to 3D. The applied method is a finite difference method. The algorithm is the *SUccessive Node REDuction*, leading to the acronym SUNRED.

This tool is designed especially for the fast calculation of the thermal behavior of arbitrary shape integrated microstructures. A special requirement was to calculate on a grid that is fine enough to obtain not only the temperature distribution but the accurate streamlines of the heat-flow as well.

An interesting feature of this simulator is that characteristic methods of three distinct disciplines are combined in it. These fields are

- the electromagnetic field theory,
- the linear network theory and
- the image processing methods.

Combination of these methods resulted in an original and very useful tool.

3.1. The model

The current 2D version of the program treats the linear heat conduction problems in two dimensions. The equation being solved is in the steady-state case

$$p(x,y) = \frac{\partial}{\partial x}\left(\lambda\frac{\partial T}{\partial x}\right) + \frac{\partial}{\partial y}\left(\lambda\frac{\partial T}{\partial y}\right) \qquad (2)$$

This is the 2D form of the well-known Poisson equation. Electrical current-stream fields and electrostatic potential fields can be described by the same equation, which means that the program is capable to solve such problems as well. Thus, investigation of capacitive sensor and actuator structures can be also performed with the help of this program.

The investigated area is a rectangle. A dense equidistant grid is spawned to this area defining a cell matrix. The suggested grid size is e.g. 256×256. A material type is assigned to each cell. This assignment is performed by constructing an image – in the sense of the digital image handling methods. Each pixel of this digital image corresponds to a grid-cell whereas the material type constituting the cell is coded by the color of the pixel. Thus, in order to enter a problem two files have to be prepared:

- the "problem-image" in any usual image format (the suggested format is the BMP),
- the "material-table" assigning different material parameters to each color.

This method of problem definition provides a very easy and fast input of complex geometrical arrangements (using any general picture editing tools). Almost arbitrarily shaped structures can be investigated. Real images, e.g. a microscopic image may also be used as geometry input.

On the edges of the investigated rectangular area either forced temperature or zero heat-flow can be prescribed. Excitations can be defined in the interior of the investigated area as well, forcing a given temperature or a given heat-flux to any cell. Obviously a new "color" should be introduced for each excitation value in the problem image.

The solution of Eq.(2) is accomplished using the method of *finite differences*, and applying a network model for the thermal field. The cells of the field are described by an electrical model. The cells are squares (or rectangles), with a node in their center (Fig.2a.). Heat flux can be forced into them – this corresponds to the current flowing in this node. Forced temperature means the forced value of the cell node.

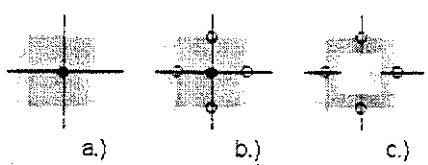

Fig.2. Cell, center node and terminal nodes

The boundary between different materials is lying always on the cell edges. In other words: each cell is "filled" by a single material. Each cell has four terminals in the direction of its four neighbors (Fig.2b.). On the terminals each cell can be described by their admittance matrix. This way the center node is hidden, but knowing the terminal temperatures the temperature of the center node can be back-calculated. Fig.2c. presents that the cell shows four terminals to the outside and the inner node is hidden.

The steady-state model of the cell is shown in Fig.3. It contains four thermal conductances. The values of these conductances depend on the thermal conductivity of the material filling the cell and on the geometry. This basic cell can be described by an admittance matrix of 4×4 size.

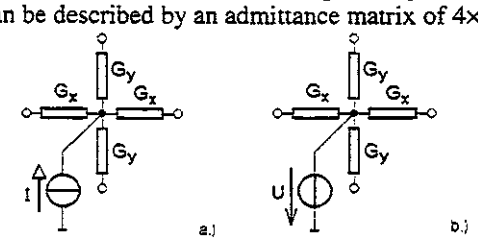

Fig.3. Steady-state circuit models of a single cell. a.) Current excitation, b.) forced voltage

3.2. The solution algorithm

The solution of the problem is done by the electrical solution of the whole model network. This raises serious problems because of the size of this network. Using a grid of 128×128 lines the model network consists of 32768 nodes. For a 256×256 grid arrangement this number is 131072. Although the corresponding circuit matrix is extremely sparse the solution of such a big network is a hard problem.

In order to avoid the troublesome "when to finish the iteration" problems we have not considered iterative solutions – only direct methods have been investigated. Similarly to the idea described in [7], a successive procedure has been developed for the network reduction. The essential features of this algorithm are briefly presented in this paragraph.

Four basic cells can be assembled to form a block or macrocell as shown in Fig.4a. In other words: a *1st order cell* has been built from four *zero-order cells*. The four interior connecting terminals of the cells can be eliminated; they will not appear in the outside-directed description.

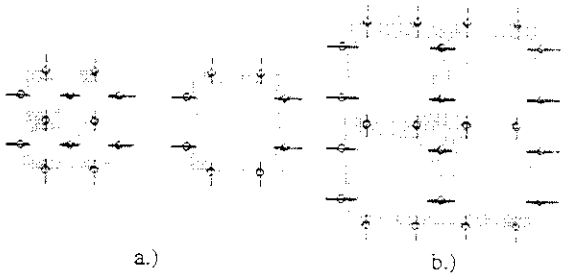

Fig 4. Network reduction. a.) Four basic cells will constitute a 1st level cell, b.) Building a 2nd level cell

Using four 1st level cells we can assemble a 2nd level cell as shown in Fig.4b. The inner terminals can be eliminated again.

Continuing this successive construction of higher and higher level cells we obtain finally the matrix of a single cell – the terminals of which are lying on the four edges of the investigated rectangular field. Matching with the boundary conditions means the solution of this matrix for the U or I constraints, given individually for the terminals lying on the boundaries of the investigated field. The voltages of all the inside nodes can then be calculated by a successive back-substitution.

Let us present the procedure in terms of the data flow and arithmetic operations used. The cells are described by their admittance matrices **Y** relating to the boundary nodes and by their inhomogeneous vectors **J** representing the excitations on all the inside nodes but reduced to the boundary nodes. For the 0th level cells shown in Fig.3. **Y** and **J** can be generated by using elementary calculations. The connection of two cells as shown in Fig.5 is equivalent to the addition of their **Y** matrices and **J** vectors where the areas of the connected nodes overlap as shown in Fig.5. as well.

The next step is to eliminate the inside (≡connected) nodes. The required calculations are commonly used in the network theory. These calculations are presented in details in a former paper [2]. For the reasons of page limitation we omit them here.

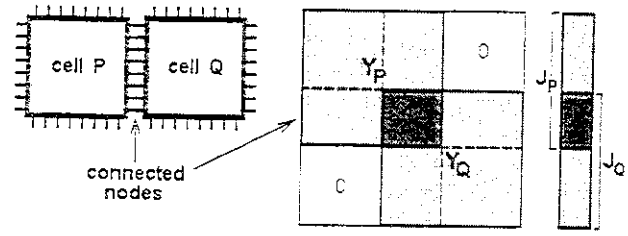

Fig.5. Connection of two cells and the resulting Y matrix and J vector

The description of all cells by their **Y** matrices represents a huge amount of data. As the processing is essentially serial, it is advantageous to store these data streams in files. Thus the organization of the program is mainly *pipelined*: the program segments read one or more streams from files and writes the results into further files. This way a quite large number of nodes can be handled on computers having only limited amount of memory.

The short description of the main program segments demonstrates clearly the pipelined process:

1st step: network reduction. The segment reads the queue of the **Y** matrices of nth level cells, reduces them in fours and writes the resulting, n+1th level **Y** matrices into a new file. Some additional matrices needed for the **2nd** and **4th** steps are calculated and stored into a further file as well. The number of runs of this segment is $\log_2(K)$, where K is the number of the pixels in one edge of the problem-image.

2nd step: forward substitution. This segment reads the queue of the **J** inhomogeneous vectors of nth level cells, reduces them in fours by using the additional matrices written in the 1st step and writes the resulted, n+1th level **J** vectors into a new file. The number of required runs is $\log_2(K)$ again.

3rd step: solution. This segment uses the uppermost level **Y** matrix and **J** vector and solves the corresponding system of linear equations, taking into account the actual boundary parameters.

4th step: backward substitution. This segment calculates the voltages on the internal nodes in a hierarchical top-down order.

The hierarchical network reduction requires $\log_2(128)=7$ successive steps for the 128×128 grid, 8 steps for the grid-size of 256 and so on. The detailed analysis of the N total number of operations on floating values gives

$$N \cong 127\, P^{3/2} \qquad (3)$$

where P is the node number for the whole model network.

In the literature of the matrix algebra or among the functions in the computer libraries we can find a number of different algorithms and routines dedicated to solve sparse matrices very effectively. In some cases Ordo($P^{1.1}$) or even better results have been reported for electrical networks. This fact raises the question: is the algorithm of successive node reduction superior in time saving or not?

We proceeded some comparative studies in order to answer this question. We have investigated three direct sparse algorithms:

CH the algorithm described in a classical work of network theory [8],

S1.3 SPARSE 1.3 sparse solver of the Netlib library (Univ. of Berkeley)

ILU sparse solver of [9] that uses incomplete LU factorization.

The first important result is that the Ordo($P^{1.1}$) or similar results can be achieved only in the case if the network is mostly one-dimensional as it is typical for the electrical circuits where the network is arranged along the line or the tree structure of the signal propagation. If the network constitutes a two-dimensional net with equal sizes in both directions we could not reach better result than Ordo($P^{1.5}$) with the investigated three methods. The number of the required floating point operations were generally less, however, than in the case of SUNRED.

The second important conclusion is that the considered sparse techniques suffer from the huge overhead of the preparation and organization time. This fact is obvious if the total computer time is investigated. Fig.6 shows the run-times on a SUN Enterprise 2170 200 MHz computer in the function of the node number. In this comparison a fourth, iterative solution is considered as well: the sparse Gauss-Seidel algorithm (**GS**). It is obvious that for large model networks (>15000 nodes) the SUNRED algorithm is superior in computer time. It can be further concluded that the SUNRED algorithm can be considered as a member of the sparse algorithm family, with the special feature of exploiting the highly regular structure of the network to be solved and saving a great part of the time devoted to organizing/ordering the calculations.

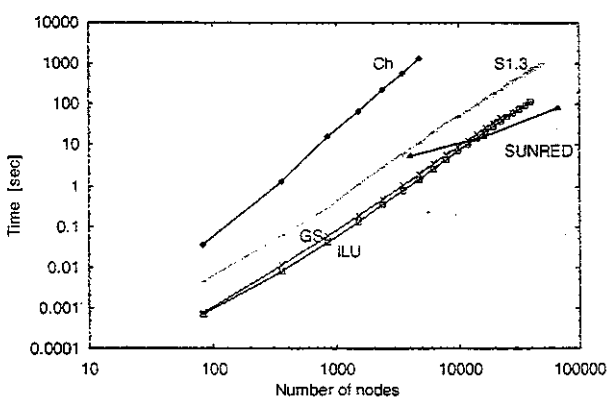

Fig.6. Total computer times of the different sparse solutions

The SUNRED program provides both the steady-state analysis and the transient (time-domain) simulation. In the latter case the reverse-Euler integration scheme is used. This method reduces the problem of the transient analysis to a d.c. solution for each time-step.

A useful feature of the program is that 3D structures having cylindrical symmetry can be calculated with the same 2D algorithm. The basic idea of the corresponding algorithm is a transformation of the heat-conduction equation. Details of this transformation are presented in [2].

3.3. Presentation of the results

The results of the simulation are treated as images again. The temperature (or potential) fields, which are essentially 2D scalar functions, can be considered as black-and-white images. The brightness of the image points is proportional to the temperature (or potential) of each point. The program provides the results in the form of digital images, in the standard BMP image format. Such a potential image is shown in Fig.7.

Although this image gives a good qualitative view of the potential field, the potential values can not be read from this picture. A basic procedure of the image processing can help to overcome this problem. This procedure is the intensity transformation: an arbitrary $b_{tr}=f(b)$ function can be used to map the original b pixel intensities into the b_{tr} brightness values. A set of appropriately chosen functions offer a rich variety of presentations for the same temperature (potential) field, and provide the good quantitative evaluation at the same time. Fig.8. is an example for this: it is the same image as in Fig.7. but after a suitable intensity transformation.

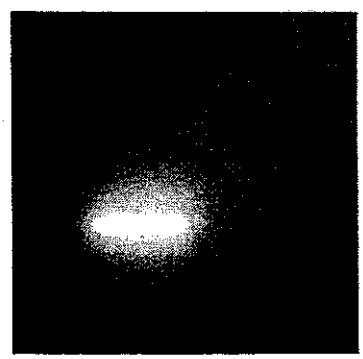

Fig.7. Grayscale image of a potential field

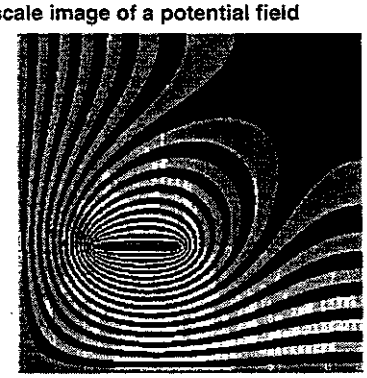

Fig.8. Potential or temperature field using a telegraph signal like intensity mapping.

The temperature field or potential image is generally not enough to visualize thermal, electrostatic or streaming fields. Tracing of the heat-stream lines (or electrical field lines) is an often encountered requirement. Especially in the case of temperature fields the streamlines provide an easy way to "discover" how and where the heat flux is streaming.

To obtain streamline pictures further image processing steps are required. The simulator provides the J_x, J_y components of the J current density vector in form of two intensity images. These vectors have to be turned by 90°. This can be made simply by interchanging the two images and negating one of them:

$$G_x = J_y \qquad G_y = -J_x \tag{4}$$

It can be easily proven that the $P(x,y)$ potential function of this G field is suitable to trace the streamlines.

The equipotential lines of this *P pseudopotential* are the streamlines of the original field. Visualization of these lines can be proceeded in the same way as in case of the real potential field.

This pseudopotential can be constructed if and only if there is no divergence in the $J(x,y)$ vector-field. This means that the procedure can be applied only on the divergence-free regions of the field. A combined streamline and potential picture generated by using this pseudopotential is shown e.g. in Fig.14.

3.4. Examples

The first example demonstrates how a microphotograph can be used as the input for problem statement. The electron-microscopic image of a cantilever made by front-side micromachining is shown in Fig.9. Image processing methods (edge detection and smoothing) were used to extract the contours of the structure and to color them appropriately. The resulting image, shown in Fig.10, was used as the input of the simulation. A small size dissipating element was defined on the broadened part of the cantilever. This dissipator appears as a lighter rectangle in the Fig. 10.

Fig.9. Electron-microscopic image of a cantilever

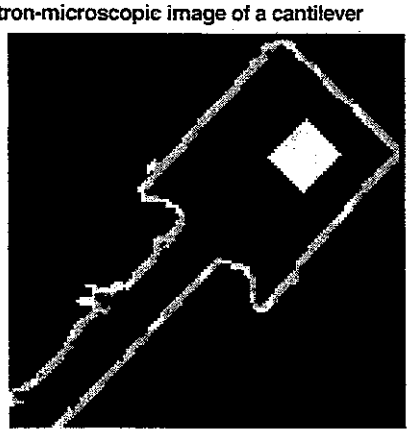

Fig.10. The same photograph after image processing steps

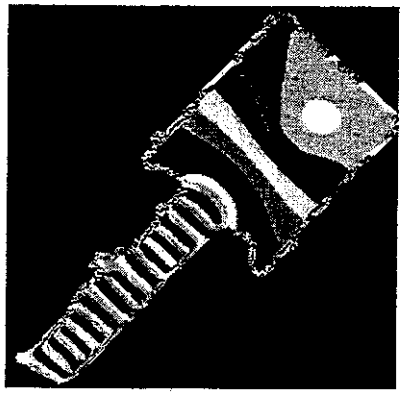

Fig.11. Results of the simulation

The results of the steady-state simulation are shown in Fig. 11. The different shadows correspond to regions of different temperatures.

The second example is an electro-static problem. A cantilever is investigated again. The cross-section of the structure is shown in Fig.12a. 10 V potential is applied to the upper conductive layer of the cantilever while the bulk is grounded. The equipotential lines are shown in Fig.12b. A useful feature of the program is the calculation of the divergence of the resulting field. This divergence is proportional to the charges. Fig 12c shows the divergence image of the problem. The black lines show the positive charges while the light ones correspond to the negative charges. Knowing the magnitudes of these charges the electro-static forces can be calculated easily. Incorporation of this feature is subject of the recent upgrade of the program.

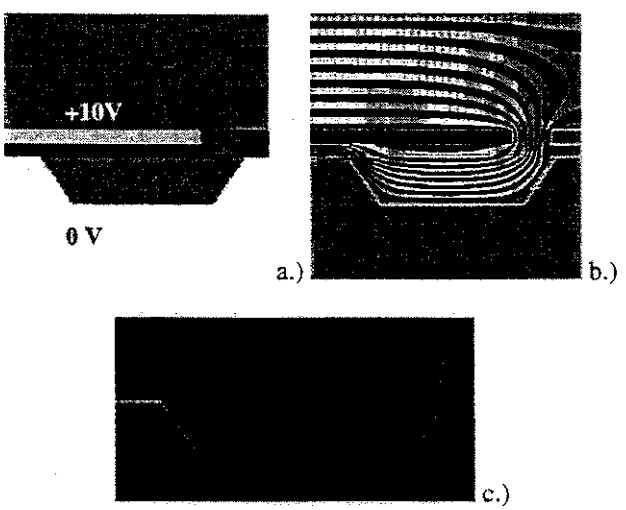

Fig.12. Electro-static simulation of a cantilever

In the next example the simulation of an H-shaped membrane is presented. The membrane is suspended by four handles. A single dissipating component is considered on the surface of the membrane as it its shown in Fig.13.

The results of the steady-state simulation are presented in Fig.14. Both the isotherms and the heat-flow lines are visualized.

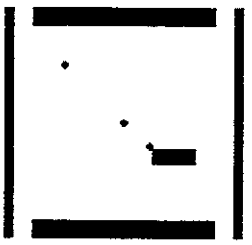

Fig.13. The H-shaped membrane. The gray rectangle is the dissipating element

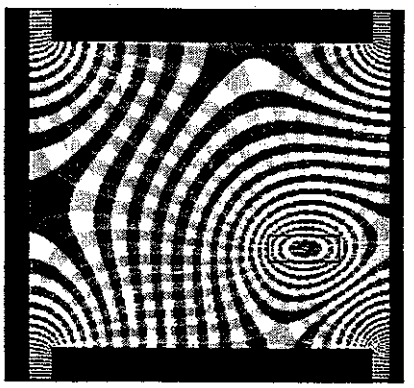

Fig.14. Steady-state simulation results for a H-shaped membrane

The transient analysis of the problem generates a frame sequence that can be played as a movie. Four frames of such a sequence are shown in Fig. 15. These frames are related to the transient process that occurs if the dissipation sets-in in the t=0 instant. Both time and spatial functions can be extracted from the frame sequence. In Fig.16 the time functions of the temperature are visualized referring to the three locations marked by dots in Fig.13.

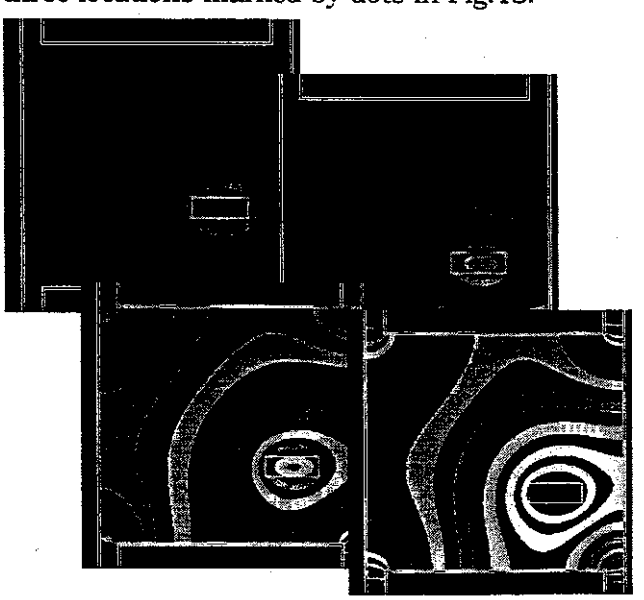

Fig. 15. Four frames of a time-sequence resulted by the transient simulation

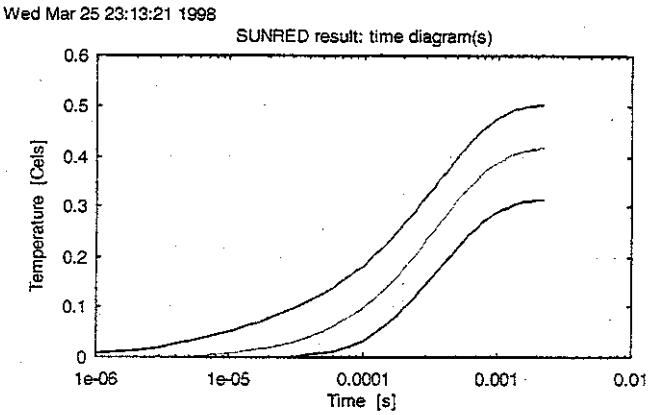

Fig.16. Time diagrams for the problem of Fig.13.

4. Conclusions

Two new, fast and easy-to-use thermal simulation tools have been developed for the accurate thermal simulation of dedicated microsystem elements and special 3D structures. With the help of the novel 2D-SUNRED program arbitrarily shaped structures can be thermally analyzed in minutes, currently in two dimensions. The development of the 3D version of the program is in progress. Using the μS-THERMANAL simulator, both the steady-state and the frequency-domain behavior and even the accurate dynamic thermal model of microsystem elements can be obtained.

References

[1] **V. Székely, M. Rencz:** Fast field solver programs for thermal and electrostatic analysis of microsystem elements, ICCAD'97 Int. Conf. on Computer-Aided Design, Nov. 9-13 1997, San Jose, Ca, USA, Proc. pp.684-689

[2] **V. Székely, M. Rencz:** Fast Field Solvers for Thermal and Electrostatic Analysis, DATE'98 Conference, Febr. 23-26. 1998, Paris, pp. 518-523.

[3] **V. Székely, A. Csendes, M. Rencz:** μS-THERMANAL: An efficient thermal simulation tool for microsystem elements and MCM's. SPIE'96 Symposium on Micro-machining and Microfabrication, 14-15 October 1996, Austin, Texas, USA, SPIE Proc. Vol. 2880, pp. 64-75.

[4] **A.G. Kokkas:** Thermal analysis of multiple-layer structures, IEEE Trans. on El. Dev., V. ED-21, No. 11, pp. 674-681 (1974)

[5] **V. Székely, P. Baji, M. Rencz:** Graphical computer methods in the design of integrated circuits, Periodica Polytechnica, V. 23, No. 3-4, pp. 331-338 (1979)

[6] **V. Székely, A. Poppe:** Novel tools for thermal and electrical analysis of circuits, Electrosoft, V. 1. No. 4. pp. 234-252 (1990)

[7] **T. A. Johnson, R. W. Knepper, V. Marcello and W. Wang:** Chip substrate resistance modeling technique for integrated circuit design, IEEE Trans. on Computer-Aided Design, Vol.CAD-3, No.2, pp. 126-134 (1984)

[8] **L. O. Chua, Pen-Min Lin:** Computer-aided analysis of electronic circuits, Prentice-Hall, Englewood Cliffs, 1975

[9] **Barrett et.al.:** Templates for the solution of linear systems: Building blocks for iterative methods, DARPA & ARO supported project DAA L03 - 91 - C - 0047