UDK621.3:(53+54+621+66) ISSN0352-9045

SOFTWARE TOOL FOR THE SUPPORT OF ON-LINE THERMAL MONITORING OF MICROELECTRONIC SYSTEMS

Marko Lamot, Hermes Softlab, Ljubljana, Slovenia
Borut Žalik, Univerza v Mariboru, Fakulteta za elektrotehniko, računalništvo in informatiko, Maribor, Slovenia

Key words: physics, electrotechnics, electronics, microelectronics, microelectronic systems, temperature monitoring, temperature sensors, measuring points, Delaunay triangulation, interpolation, simple algorithms, computer software, computer software tools

Abstract: An approach and the supporting software tool for the construction of isotherms for thermal monitoring of microelectronic systems is described. The approach is based on the triangulation of a set of measuring points corresponding to the positions of temperature sensors within the monitored system. Our implementation easily handles examples with thousands of points which drastically exceeds the number of temperature sensors in currently reported case studies. Construction of iso-lines and their interpolation to isotherms are algorithmically simple, fast and non-demanding solutions appropriate for on-line thermal monitoring implementations.

Programsko orodje za podporo sprotnemu nadzoru temperature v mikroelektronskih sistemih

Ključne besede: fizika, elektrotehnika, elektronika, mikroelektronika, sistemi mikroelektronski, nadzor temperature, senzorji temperature, točke merilne, Delaunay triangulacija, interpolacija, algoritmi enostavni, oprema programska računalniška, orodja programska racunalniška

Povzetek: V članku opisujemo postopek, podprt z izdelanim programskim orodjem, za konstrukcijo izoterm za nadzor temperature v mikroelektronskih sistemih. Postopek je osnovan na triangulaciji množice merilnih točk, ki ustrezajo pozicijam senzorjev v nadzorovanem sistemu. Razvita programska oprema z lahkoto obravnava primere z več tisoč točkami, kar sicer drastično presega število temperaturnih senzorjev v sedanjih eksperimentalnih študijah. Za konstrukcijo izoterm obstaja algoritmično enostavna, hitra in računsko nezahtevna rešitev primerna tudi za izvedbo sprotnega nadzora temperature v namenskih sistemih.

1 Introduction

With rapidly increasing complexity and operation speed of modern VLSI circuits and systems, temperature issues are becoming an important factor that must be considered for fail safe operation. On-line temperature monitoring by means of built-in temperature sensors aims at detecting potential thermal problems. If the temperature of the monitored part of the circuit or system increases and gets close to the given limit, precautions like decreasing the operating frequency, changing the execution of tasks or temporally switching off some parts of the system can be taken.

The common types temperature sensors are thermoresistors, thermocouples and active devices such as diodes and transistors. They can be produced with the same technologies as integrated circuits which makes it possible to integrate a temperature sensor in an IC. Sensors for thermal monitoring of microelectronic structures should have low power consumption (typically in the order 1mW or less), temperature range of 0-150 °C, and they should encompass a reasonably small area. In addition, linearity, accuracy and long-term stability are required. Different innovative approaches to the design of temperature sensors for on-line thermal monitoring have recently been reported /1-4/.

On-line temperature monitoring of a system requires continuous acquisition and processing of temperature measurements. The resulting data flow might slowdown the overall system operation hence some practical solution is needed to overcome the problem. Szekely *et al.* /2/, /4/, proposed the integration of temperature sensors within the bundary-scan (BS) infrastructure /5/. The idea is to integrate within the BS circuitry a frequency-output temperature sensor together with an internal counter and register to store the measurement result and to use the serial BS test data input/output line for the transfer of temperature measurement data. By minor modification of standard BS architecture and the addition of a few BS instructions, a temperature sensor with 2ms sampling time and 0.06 °C resolution has been implemented. The salient feature of this solution is that the existing BS test data input/output line can be employed to transmit temperature measurement data among the parts of system. BS approach, originally defined for digital systems has been extended to mixed-signal circuits /6/, /7/ which further generalises the above solution.

The management and processing of thermal measurement data via BS architecture is further elaborated in /4/. For a given set of subsystem elements (ICs or packages) with built-in temperature sensors a dedicated Thermal Monitoring Master (TMM) chip is proposed for the acquisition of temperature measurements within the subsystems. The reported case study was limited to a small number of sensors. However, if the number of sensors in a system is higher (i.e., 20-30 or even more), some strategy of selective readings of sensors need to be employed to process the data efficiently. Furthermore, problems of calibration need to be considered as well as diagnostic solutions in the case when individual sensors give unrealistic temperature values. The ultimate goal is, of course, to have correct and sufficiently detailed presentation of the surface temperature distribution

The tool described in this paper offers an effective solution for the construction of isotherms. The approach is based on the triangulation of a set of measuring points, which can be performed off-line once the sensor positions are established. The next steps: construction of iso-lines and their interpolation to isotherms are algorithmically simple, fast and non-demanding solutions appropriate for on-line thermal monitoring implementations.

In the following we give a brief introduction to triangulation, construction of iso-lines and isotherms. We present the basic principles and comment the feasibility of the proposed solutions in practice. At the end of each section we describe the main features of our implementation. Finally, in the conclusion, we sketch the idea of the application of the implemented tool in on-line thermal monitoring.

2 Triangulating a Set of Measuring Points

2.1 Background

For a set *S* of 2D points, let us determine the maximal set of edges connecting these points such that the edges do not intersect. The problem is known as a maximal plane subdivision, but in engineering, it is usually referred to as *triangulation* /8/. One can perform, of course, many different triangulations upon the same set of points. Obviously, not all are good for engineering applications and we need a criterion for selecting the most appropriate one. Two criteria are widely accepted today:

- the first seeks for a triangulation whose total length of edges forming the triangulation is minimal (so-called minimal weight triangulation) /8/, and
- the second selects those triangulation whose smallest inner angle of all created triangles is maximal /9/.

There is a strong feeling that a determination of a minimal weight triangulation is NP-hard problem, although also simple approximations in a polynomial time are known /8/. However, these approximations may produce a lot of very thin triangles, which are usually not appropriate for further use in engineering. On the other hand, triangulation which optimises the smallest inner angle in the constructed triangulation creates much "better" triangles and is nowadays a preferred approach. Delaunay (a Russian mathematician) showed that there is a very simple rule - called empty circle property – which ensures that the triangulation is optimal regarding the smallest inner angle of all generated triangles /10/. He proved that every circumference circle determined by any triangle from that triangulation does not contain any other point from the input set of point *S*. Such a triangulation is referred to as Delaunay triangulation. Figure 1 shows an example of Delunay triangulation. For one of the triangles its circumference circle is plotted dashed. We can see that no any other input point falls into the circle.

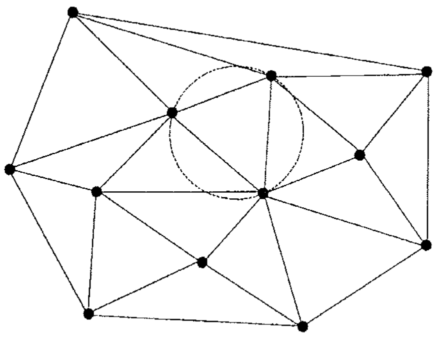


Figure 1 Delaunay triangulation

Different algorithms have been suggested for constructing Delaunay triangulation. One of the most popular today is the so-called *randomised incremental algorithm* described in /10/. The algorithm is fast and works in expected time complexity *O(n log n)*, but it is also memory demanding which often limits its applicability. On the other hand, Fang and Piegl introduced a triangulation algorithm, which is less memory demanding, but still efficient enough to handle huge set of points /11/. For acceleration, the algorithm uses a uniform plane subdivision which is very popular in applications of computational geometry /12/. We followed the Fang-Piegel's algorithm and a detailed description of our implementation can be found in /13/. Figure 2 shows a screen-dump of a 100 points being triangulated by our algorithm. To obtain better measured results and to show that the triangulation algorithm works stable, we have triangulated much larger sets of points. The CPU time measured on PC Pentium 2/266 Mhz, shows Table 1.

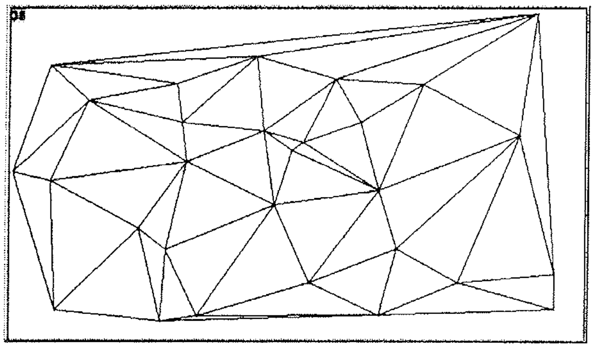


Figure 2 A screen-dump of 30 triangulated measurement points

Table 1 Spent CPU time of the triangulation algorithm

No. of points	Spent CPU (s)
1000	0.2
2000	0.6
5000	2.0
10000	4.4
15000	7.2
20000	10.8

3 Iso-line construction

Having triangulation and triangles arranged in the proper data structures, the next step is to construct the iso-lines (isotherms in our case). To speed up this part of the algorithm, each edge of the triangle has to have an information which triangles it borders (this information is obtained during the triangulation construction). Each edge is shared by at most two triangles (edges belonging to the convex hull border only one triangle). This information can be efficiently stored in the data structure using only two pointers.

The construction of the iso-lines consists of two steps:

- the set of polyline roughly interpolating the desired iso-line is constructed and
- the obtained set of polylines is interpolated by a curve.

3.1 Conctruction of iso-polylines

Let us consider an example shown in Figure 3a. The vertices of the triangles store the values of a scalar field (temperature in our case). Let us, for example, determine the isotherm displaying temperature of 100°C. This isotherm is roughly interpolated by two polylines in Figure 3a. One of them is closed and the second is not.

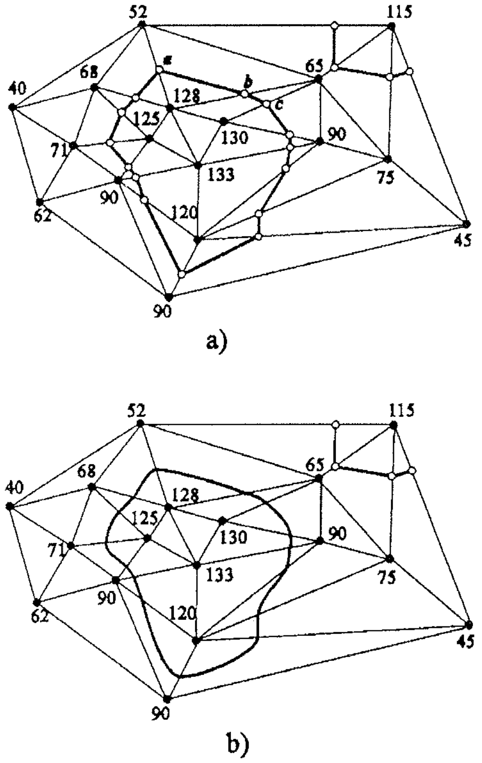
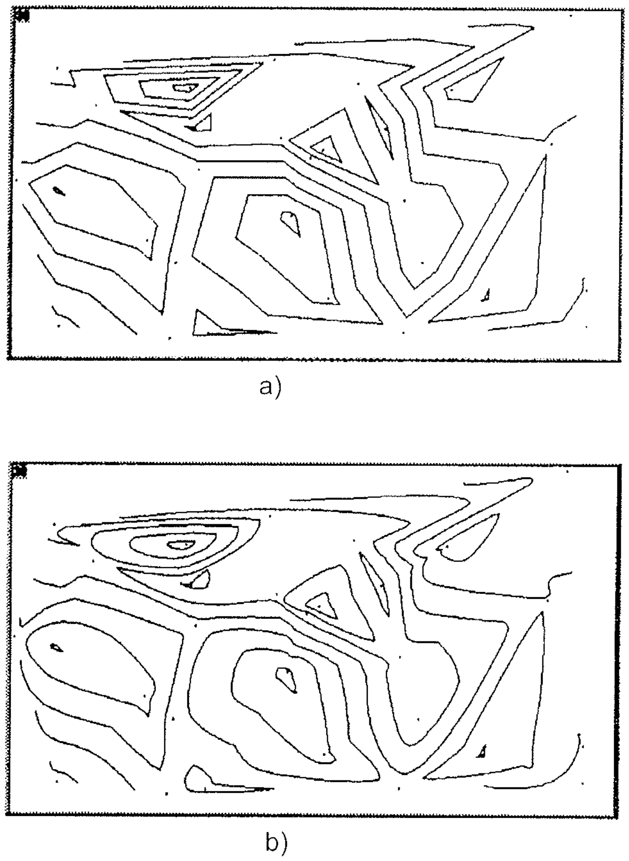


Figure 3 Construction of iso-polylines (a) Isotherms (b)

The algorithm searches at first through the set of triangles until it finds the first triangle whose vertices' temperature values enclose the value (i.e., 100°C). Suppose that it has found the triangle with the temperature value at the vertices (52°, 65°, 128°). The isotherm crosses the edges connecting vertices pairs (52°,128°) and (65°, 128°). The crossing corresponding to 100°C is determined using linear interpolation. When both points on the first triangle are determined (points ***a*** and ***b*** in Figure 3a), we mark the triangle as already used and move to the neighbouring triangle using the corresponding pointer where the next point is determined. The algorithm then continues until the starting triangle is met, or until an edge belonging to the convex hull of the set of point is determined. Such an edge has the pointer to the neighbouring triangle set to *NULL*. In this case, we must start to travel in the opposite direction from the starting triangle until the second boundary edge is met. The algorithm continues until all triangles are examined. Once the iso-polylines are obtained they are interpolated to get the isotherms shown in Figure 3b.

3.2 Interpolating the iso-polylines

There are different methods how to interpolate a given set of points and not all of them are appropriate for our purpose. Namely, the following properties are required: the curve should not oscillate, it should be smooth, the multiply values has to be easy to represented, algorithmically it must be simple and it should be plotted fast. The well-known Lagrange method, for example, does not fulfil these criteria. It oscillates and does not allow multiply values. In computer aided geometric design much better methods have been developed. They are known as piece-wise methods, because the final curve consists from segments smoothly fitting together. The most popular methods are for example: cubic Hermite interpolation, Catmull-Rom spline, Overhauser spline, Akima methods end others /14/. A comprehensive col-



b)

Figure 4 Isothermes showed by polylines (a) Isotherms interpolated by Hermit curves (b)

lection of methods is given in /15/. For our application we have used a variation of cubic C^1 Hermite interpolation. The method uses only polygons of third order therefore the calculations are fast. The result of our program is depicted in Figure 4. At the top (Figure 4a), iso-polylines are shown, while at the bottom, Hermit interpolation was used for the construction of the corresponding isotherms.

4 Conclusion

We have described an approach for the construction of isotherms based on the triangulation that can be applied in thermal monitoring of microelectronic systems. The first step, triangulation, is normally performed off-line due to the fact that sensor positions are defined once and known in advance for the given application. Our implementation easily handles examples with thousands of points /13/ which drastically exceeds the number of temperature sensors in currently reported case studies. After triangulation, the construction of isotherms is algorithmically simple and non-demanding which makes it suitable for embedded system implementations (i.e., on-line thermal monitoring). Linear interpolation that was used in the example in Section 3.1 for determining the crossing corresponding to 100°C serves just as a rough approximation, it can be replaced by other techniques describing more precisely temperature propagation on the employed materials and their geometry in each particular case.

Acknowledgement

The authors are grateful to dr. Franc Novak from Institut "Jožef Stefan", Ljubljana, Slovenia, for his help in preparation of the paper.

Literature

/1/ V.Szekely, Cs. Marta, Zs. Kohari, M.Rencz, "CMOS sensors for on-line thermal monitoring of VLSI circuits", IEEE trans. on VLSI systems, Vol.5, No.3, 1997, pp.270-276.

/2/ V. Szekely, M. Rencz, J.M. Karam, M. Lubaszewsky, B. Courtois, "Thermal Monitoring of self-checking systems", Journal of Electronic Testing: Theory and Applications, Vol.12, No.1/2, 1998, pp. 81-92.

/3/ K.Arabi, B.Kaminska, "Integrated temperature sensors for on-line thermal monitoring of microelectronic structures", Journal of Electronic Testing: Theory and Applications, Vol.12, No.1/2, 1998, pp. 93-99.

/4/ V.Szekely, M.Rencz, A.Pahi, B.Courtois, "Thermal monitoting and testing of electronic systems", IEEE trans. on Components and Packaging technology, Vol.22, No.2, 1999, pp. 231-237.

/5/ IEEE Standard Board, IEEE Standard 1149.1-1990, Standard test access port and boundary scan architecture, 1990.

/6/ F.Novak, A.Biasizzo, M.Santo Zarnik, "Considerations of IEEE 1149.4 Standard in analog design", Informacije MIDEM, Journal of Microelectronics, Electronic Components and Materials, Vol. 29, No.2, 1999, pp 85-88.

/7/ U.Kač, F.Novak, S.Maček, M.Santo Zarnik, "Alternative test methods using IEEE 1149.4", Proceedings of Design, Automation and Test in Europe Conference DATE2000, IEEE Computer Society, Paris 27-30 March, 2000, pp. 463-467.

/8/ F. P. Preparata and M. I. Shamos, "Computational geometry - An Introduction", Springer Verlag, New York, 1985.

/9/ P. Su and R. Drysdale, "A comparison of sequential triangulation algorithms", Computational Geometry - Theory and Applications, Vol. 7, 1997, pp. 361-386.

/10/ M. de Berg, M. van Kreveld, M. Overmars, O. Schwarzkopf, "Computational Geometry - Algorithms and Applications", Springer-Verlag, Berlin, 1997.

/11/ T.-P. Fang and L. A. Piegl, "Delaunay Triangulation Using a Uniform Grid", IEEE Comput. & Graphics, Vol. 13, No. 3, 1993, pp. 36 – 47.

/12/ B. Žalik, "A Topology Construction from Line Drawings Using a Uniform Plane Subdivision Technique", Comput.-Aided Design, No. 5, Vol. 31, 1999, pp. 335-348.

/13/ M. Lamot, F. Novak, in B. Žalik, "Algoritem triangulacije z uniformno delitvijo ravnine" Technical Report No. 8226, Institut Jožef Stefan, Ljubljana, april 2000.

/14/ G. Farin, "Trends and Surfaces for Computer Aided geometric Design – A Practical Guide", Academic Press, Boston, 1988.

/15/ N. Guid, B. Žalik, A. Vesel, and T. Klojčnik, "Curvmod: Algorithm Functional Description", Computer Graphics Report Series, University of Maribor, Vol. 1, No. 2, 1994.

univ. dipl. inž. Marko Lamot,
Hermes SoftLab, Litijska 51, SI-1000 Ljubljana,
(Maribor office), Slovenia.
e-mail: marko.lamot@hermes.si
tel: +386 61 1865 815
fax: +386 61 1865 816

dr. Borut Žalik,
University of Maribor
Faculty for Electrical Engineering and
Computer Sciences,
Smetanova 17, SI-2000 Maribor
e-mail:zalik@uni-mb.si
tel: ++ 386 62 220 7471
fax: ++ 386 62 211 173

Prispelo (Arrived): 12.5.00 *Sprejeto (Accepted): 30.8.00*