Thermal Simulation and Design of a GaAs HBT Sample and Hold Circuit

Ken Poulton, Knud L. Knudsen and John J. Corcoran
Hewlett-Packard Laboratories
1651 Page Mill Road, Palo Alto, CA 94304

Keh-Chung Wang, Richard L. Pierson, Randy B. Nubling and Mau-Chung F. Chang
Rockwell International Science Center
1049 Camino Dos Rios, Thousand Oaks, CA 91360

Abstract

GaAs Heterojunction Bipolar Transistors (HBTs) offer high speed and good device matching characteristics that are attractive for many high-speed circuits. However, maintaining safe operating temperatures is more difficult in GaAs circuits than in silicon because of the lower thermal conductivity of GaAs. In addition, the speed and gain of GaAs HBTs is best at high current densities and with relatively short and wide emitters, both of which increase the peak device junction temperatures.

This paper will describe methods used to predict and measure device temperatures within an IC, and their application to the design of an HBT sample and hold circuit (S/H). In particular, we will describe a new thermal simulation tool called ThCalc. ThCalc calculates the temperature profile of an IC and runs fast enough to allow calculations on a whole chip.

Introduction

For all integrated circuits, keeping device temperatures within reasonable bounds is critical to long-term reliability. Most IC failure mechanisms are accelerated at higher operating temperatures; it is common to find that an increase of 10 °C leads to a particular failure occurring in half the time. For this reason, it is usual for IC process specifications to include a maximum device junction temperature between 100 and 150 °C.

In addition, controlling device temperatures can be key to meeting performance requirements. Higher operating temperatures usually degrade device performance, for example, by reducing f_T and increasing leakage currents. Another important effect is that self-heating can cause offset and "memory" effects in bipolar transistors because V_{BE} varies at around −1 mV/°C. Even a temperature gradient across a chip may degrade performance by upsetting symmetry between otherwise matched devices.

In general, GaAs ICs suffer more from all thermal effects than silicon ICs because the thermal conductivity of GaAs is only one third that of silicon. In addition, the speed and gain of GaAs HBTs is best at high current densities and with relatively short and wide emitters, both of which increase the peak device junction temperatures. In the 50 GHz HBT process we used [1], emitter widths are 1.4 μm and current densities of up to 0.8 mA/μm^2 are allowed. This gives a linear current density of 1.1 mA/μm, which is 2 to 5 times higher than found in most GaAs FET processes.

The S/H Circuit — First Layout

We designed and laid out an HBT sample and hold circuit (S/H), attempting to keep the device junction temperatures below 150 °C. Assuming a heatsink temperature of up to 80 °C, this allows an on-chip temperature rise of up to 70 °C. We estimated the thermal resistance of individual HBT devices using the expressions found in [2] and designed the circuit to keep the self-heating rise of each device below 40 °C. We were not able to calculate the proximity heating (how much each device heats up its neighbors) and the circuit was laid out as densely as allowed by the design rules. The power in the core area (600 × 400 μm) is about 2 W.

Initial testing of the fabricated circuit indicated very high operating temperatures and short device lifetimes. By comparison to deliberately overheated single transistors and resistors, we estimated temperatures in the S/H were in excess of 300 °C. This stimulated the development of a software tool for thermal calculations which we call ThCalc.

Inside ThCalc

Physics

In general, static temperatures obey the Poisson equation [2]:

$$\nabla(k\nabla T) = -\dot{q} \tag{1}$$

where k is the thermal conductivity of the medium, $T(x,y,z)$ is the temperature and $\dot{q}(x,y,z)$ is the power generation per unit volume in the medium.

There are various known methods for solving this equation, both analytical and numerical, but most assume that k is independent of T. However, this is not true for semiconductors. The thermal conductivity of GaAs is roughly

$$k(T) = k(T_0)\left(\frac{T}{T_0}\right)^n \tag{2}$$

where $k(300K) = 0.44$ W/cm-K and n is about −1.25 [2], [3]. This means that the thermal conductivity of GaAs decreases by 30% between 25 and 125 °C. Silicon's thermal conductivity is about three times higher, but has a similar temperature dependence.

This would seem to invalidate the known methods of solving the Poisson equation. It can be shown, however, [4] that we can reduce the above two equations to the standard (constant-k) Poisson equation with a change of variables:

$$\tau = T_0 + k^{-1}(T_0)\int_{T_0}^{T} k(T')dT' \tag{3}$$

The pseudo-temperature τ now satisfies the constant-k form of the Poisson equation and the usual solution methods apply. This allows us to solve for temperatures assuming a constant k and to use superposition to account for multiple heat sources. We may then transform back to the the real temperature rise (above the chip backside temperature T_0) via

$$\Delta T = T_0\left[\left\{\frac{(n+1)(\Delta\tau + T_0)}{T_0} - n\right\}^{\frac{1}{n+1}} - 1\right] \tag{4}$$

Standard Solution Methods

Since Poisson's equation also describes electrostatic fields, there is a lot of literature on its solutions. The well-known methods of solving Poisson's equation include

- analytic solutions
- numerical series-summation methods
- numerical finite-element/finite-difference solutions

Analytic (exact) solutions are obviously the most attractive, but the only relevant result seems to be the solution for a point (or hemispherical) source on the surface of a semi-infinite medium:

$$T(r)=\frac{P}{2\pi kr} \qquad (5)$$

where P is the source power and r is the distance from the source to the point in question.

When we add the boundary condition of constant temperature at the backside of a chip, however, it seems that there are no closed-form solutions. This brings us to series-summation solutions. Two such programs [5] [6] were tried, but the series convergence, even for sophisticated summation algorithms, is very slow. These programs may require hours to calculate temperature profiles consisting of only one source and a few dozen calculation locations (using a 14 MIPS workstation).

As with the series-summation programs, finite-element and finite-difference simulators [7] [8] can provide important results for simple cases, but are also far too slow for full-chip profiles. In addition, they require large amounts of memory (16 to 48 MBytes for a single-device simulation), making simulations of even SSI chips impractical.

ThCalc's Method

ThCalc provides a computationally-efficient solution by taking advantage of the relatively simple set of geometries needed to represent heat flow in integrated circuits. It may be thought of as a series of simple steps:

1) start with the analytic solution for a point source
2) integrate in 2 dimensions to model a flat rectangular source
3) add empirically-determined corrections to account for finite substrate thickness, die attach layers, surface layers, chip edges, etc.
4) add up the contributions of multiple heat sources (superposition)
5) transform from pseudo-temperatures to real temperatures to account for the semiconductor's temperature-dependent thermal conductivity
6) simplify where possible to speed up solutions without sacrificing much accuracy.

We start with the analytic solution for a point source, and integrate in x and y (the two dimensions in the plane of the chip's surface). This gives a complicated expression for the temperature distribution on the surface of a semi-infinite medium due to a rectangular ($w \times l$) source [2]. (In the far-field ($r>\sqrt{w^2+l^2}$) we revert to the simpler point-source solution for greater speed.)

Next, we account for the effects of a finite substrate thickness. It was found that for substrate thicknesses larger than the source dimensions, the infinite-substrate surface temperature distribution should be multiplied by $e^{-\frac{r}{h}}$ where h is the substrate thickness and r is now the horizontal distance from the source to the calculation point ($r=\sqrt{x^2+y^2}$). This provides a very close match to results from the finite-difference and series-summation programs. For this and most of the other empirical expressions, the finite-difference electrostatics simulator fcap3 [8] was used as the main standard of comparison. Pamin [6] was the series-summation program used. (Note that ThCalc only calculates temperatures near the semiconductor surface; the above correction factor does not hold near the backside of the chip.)

For a source with dimensions comparable to the substrate dimensions, ThCalc breaks the source down into smaller pieces and treats them as separate sources. As a check, we note that in the limit of sources much wider than the substrate thickness, ThCalc does converge to the parallel-plate limit, albeit with a great deal more computation.

Next, we wish to account for extra layers beneath the chip, e.g., an epoxy die-attach layer. This contribution was derived exactly from the point-source, finite-substrate expression described above for the case of the second layer having the same thermal conductivity as the first. Empirical correction factors for different values of thermal conductivity were then added in. The resulting expression is

$$\Delta\tau_2(r)=\frac{P}{2\pi k_2 r}\left[e^{-\frac{r}{\left(\frac{k_1}{k_2}\right)^{1/4} h_1+h_2}}\right]\left[1-e^{-\frac{r h_2}{\left(\frac{k_1}{k_2}\right)^{1/2} h_1(h_1+h_2)}}\right]$$

where k_1 and h_1 refer to the semiconductor substrate and k_2 and h_2 refer to the die-attach layer. This is found to match well with finite-difference simulations for all "reasonable" cases of die-attach layers (to be explicit, when $h_2/h_1 < 25k_2/k_1$).

The effect of chip boundaries may also be included in the simulation. These effects are included by the method of images: each source in the chip is mirrored about the four chip edges to create an array of 9 sources (the original and 8 images). For all practical cases (chip thickness less than the horizontal dimensions) 8 images is completely sufficient. The drawback to using this feature is that it increases compute time.

For verification purposes, it is also useful to predict temperatures at the top of the dielectric and encapsulation layers of the finished IC. These predictions can be compared directly to infrared or liquid crystal surface temperature measurements. For this purpose, we modify r, the horizontal distance from the source to the calculation point to allow small values of z:

$$r=\sqrt{x^2+y^2+z^2}$$

In practice, the value of z used is chosen to best fit the temperatures seen in finite-difference simulations of device, interconnect and intermetal dielectric layers. In fact, even for junction temperature calculations, the value of z is typically in the range of 0.5 to 1.5 um to account for the effects of non-flat device geometries and the cooling effects of interconnect metal.

It is important to note that although several empirical expressions are used in the program, the value of z is the only "fudge factor" in the model. All other parameters given by the user are simply physical descriptions of the chip and layout geometries.

Correcting for Temperature-dependent Thermal Conductivity

The above calculations produce the pseudo-temperature τ for one source. After adding together the contributions from all the heat sources, we then convert these calculated pseudo-temperatures to real temperature rises using equation (4) above.

The contributions of non-semiconductor layers are handled separately: the thermal conductivity of these materials is generally fairly independent of temperature, so their contri-

butions are left unchanged in this step.

Performance Tricks

As mentioned earlier, the complicated expression for a rectangular source is used only in the vicinity of the source. Once the calculation point is far enough from the source, we use the simpler point-source expression.

The other major performance feature of ThCalc is the grouping of sources by physical location. The number of groups varies between 30 and 200 depending on the number of sources. ThCalc finds the weighted center of all the power sources in each group, and uses that as a single source to represent all the sources in the group when the distance from the group edge to the calculation point is more than 1.5 group widths. This simplification leads to about a 7× speedup in ThCalc for a 2400-device (silicon) chip. It also reduces the runtime penalty for using the method of images from 9× to 2 to 5×. The loss in accuracy is only about 1%.

Comparison to Other Tools

Figure 1 shows the thermal profile of a 2.2 × 2.2 μm source on a 75 μm thick substrate, attached with 50 μm of epoxy, as calculated by Fcap3, Pamin, and ThCalc (these dimensions were chosen to exercise the tools rather than represent real devices). All three simulations agree quite well. ThCalc, however, runs over 100,000 times faster than either of the more general simulators and takes little memory.

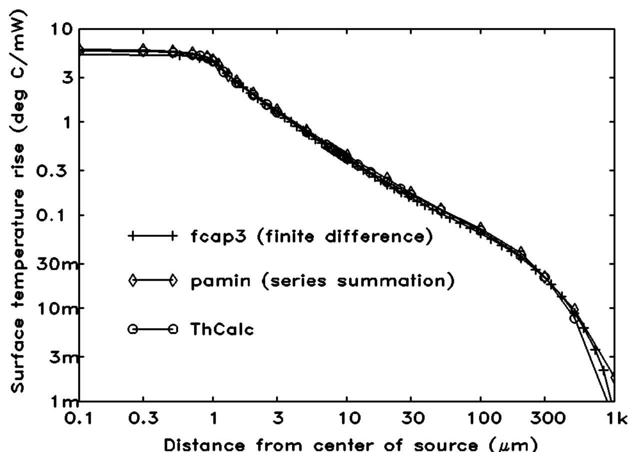

Figure 1: Fcap3, Pamin and ThCalc: 2.2 × 2.2 μm source, h_{semi} = 75 μm, h_{epoxy} = 50 μm

Using ThCalc

Characterizing a New Process for ThCalc

To fully characterize a new process for ThCalc, we need to choose a value of *z* for each type of power-dissipating device (transistors, diodes, resistors). Detailed fcap3 models of individual devices are built and run for several different dimensions of each device. The ThCalc *z* parameter is then adjusted for each device to give the best fit to the fcap3 results. The values obtained usually are 1 to 3 times the physical z-dimension of the devices' power-dissipating region, but also depend on whether the power region is buried or on the surface of the semiconductor, and what interconnect metal geometries are typically used.

This characterization process can be skipped, of course, and guesses or even a value of 0 can be used for *z*. The value of 0 is guaranteed to be conservative, since this corresponds to the highest power density case, a flat 2-dimensional power region. This will exaggerate the temperature rise of each device due to self heating, but not change the thermal influence of one device on another.

Comparisons to Measurement

Detailed 3-dimensional descriptions of two HBT devices were entered into fcap3 and simulated. The predictions of fcap3 are compared to measured [9] temperature rises for those transistors in Figure 2. The temperature-dependent thermal conductivity of the semiconductor is demonstrated by the upwards bend of the lines (the fcap3 results have been post-processed to include this effect, too). The agreement is within 3% for the small device and low by 17% for the large device. This is not as close as we would like, but is adequate for most purposes.

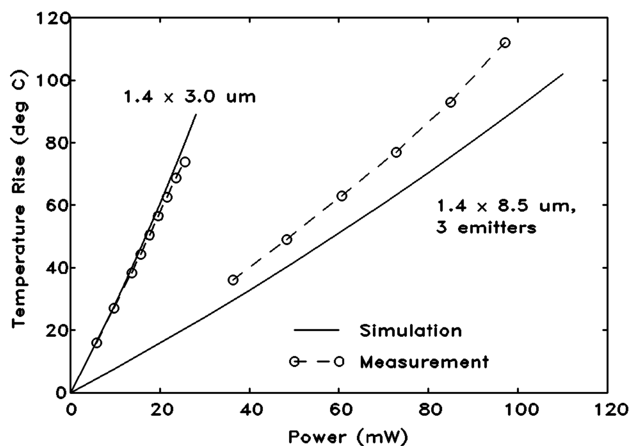

Figure 2: Measurement and simulation of temperature rise in two HBT devices.

Using ThCalc for Design

ThCalc's input is a text file listing the location, size, power and type of each device in the chip. For early stages of a design, this list may be constructed by hand, at least for small parts of a chip. During or after the layout phase, it can be extracted from the chip database. We wrote one tool which extracts the device geometries from our artwork database and another which finds the power in each device using a SPICE simulation. These are merged together to form the input file for ThCalc.

Two useful forms of output from ThCalc are a text listing of peak temperature in each device, and contour plots of the temperatures in the chip. The contour plots give the designer an intuitive feel for the locations of the major concentrations of power. The text listings are sorted to show the devices with the highest temperatures. ThCalc thus provides both qualitative and quantitative feedback on circuit temperatures. This allows the designer to concentrate on reducing temperatures only in the hot spots.

Running ThCalc to generate the device listing for a 2400-device chip took about 4 minutes on a 14 MIPS workstation; the contour plot took about an hour.

To reduce the peak temperatures in bipolar ICs, several strategies are available (in order of increasing disruption to

the circuit design):

- use device layouts with narrower emitters (with the same area) and/or wider spacing between emitters
- spread devices farther apart from each other
- decrease device current density (increase emitter or resistor area)
- reduce the power dissipated in the circuit.

Application to the S/H Circuit

We used ThCalc to calculate the temperatures in the first layout of the HBT S/H chip. The thermal contour plot is shown in Figure 3.

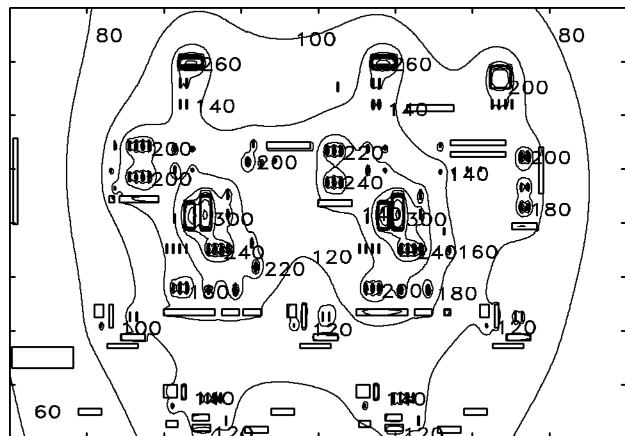

Figure 3: Simulated temperature profile of HBT S/H

ThCalc predicted that the highest device temperature rise on the S/H chip will be about 300 °C on a pair of resistors near the center. This agreed with the location of visible thermal damage to the polyimide layer over the chip, which is expected to begin at around 250 °C. Destructive testing of some S/H chips and single devices indicated that the predictions of ThCalc agree with measurement, though the uncertainties of these measurements were about 25%.

Also on the same IC, a diode added to the circuit as a temperature probe showed a rise of about 110 °C (by V_{BE} measurements). ThCalc predicted 104 °C.

We then used ThCalc to calculate the temperature reductions possible by backlapping the substrate from 25 down to 3 mils and using a solder die attach instead of epoxy. ThCalc predicted that temperatures would reduced by over 100 °C (see Table I), but this is still far from the goal of temperature rises less than 70 °C.

Using ThCalc as a guide, the hottest areas of the S/H circuit layout were then modified using the first three strategies listed above. For the new layout, ThCalc calculates temperature rises of less than 75 °C. This represents reductions in temperature rises of 1.6× to 2.7×. These reductions were achieved with only a 25% increase in the area of the S/H core and negligible change to the simulated electrical performance. This demonstrates that using ThCalc, large reductions in operating temperatures can be made with quite modest area increases.

Chips have been fabricated and tested using the second layout and no evidence of overheating was observed. Two diodes were placed near the edge and center of the S/H to act as temperature monitors; V_{BE} measurements showed rises of 12 and 28 °C at these two diodes for a 3 mil, solder-attached die. ThCalc calculated temperature rises of 12 and 24 °C for these two locations.

layout	substrate thickness (mils)	die attach	temperature rise hottest resistor	temperature rise hottest HBT
first	25	epoxy	305 °C	255 °C
first	3	solder	200 °C	112 °C
second	25	epoxy	150 °C	154 °C
second	3	solder	74 °C	71 °C

Table I: Simulated junction temperature rises (above heatsink temperature) before and after use of ThCalc.

Summary

Thermal design is important for many kinds of GaAs circuits, but it is a key factor for GaAs HBT circuits. We wrote a new simulation tool called ThCalc to efficiently calculate integrated circuit temperature profiles from design data; its results show good agreement with the limited experimental data available. It is very fast compared to more general simulators and uses only one non-physical input parameter.

We used ThCalc to guide the re-layout of an HBT S/H chip to ensure that safe operating temperatures were maintained. Large reductions of peak junction temperatures were achieved with only a modest increase in die area.

Acknowledgments

Thanks for discussions and use of their thermal calculation programs go to Ken Lee, Tom Knotts, Bob Yeats and Greg Pfeiffer of HP and Prof. Chin Lee and Jay Min of UC Irvine.

This work was supported in part by Wright Laboratory, Wright-Patterson Air Force Base under contract F33615-88-C-1737.

References

[1] R.B. Nubling, J. Yu, K.C. Wang, P.M. Asbeck, N.H. Sheng, M.F. Chang, R.L. Pierson, G.J. Sullivan, M.A. McDonald, A.J. Price, and D.M. Chen, "High Speed 8:1 Multiplexer and 1:8 Demultiplexer Implemented with AlGaAs/GaAs HBTs", *Technical Digest 1990 GaAs IC Symposium,* pp. 53-56

[2] David Smith and John O'Neil, "Measurement and Prediction of Operating Temperatures for GaAs ICs" *IEEE SEMITHERM 86 proceedings,* paper A.5

[3] J.S. Blakemore, "Semiconducting and other Major Properties of Gallium Arsenide", *J. Appl. Physics,* 53(10), p. R123-R181, October 1982

[4] Whemple, et al, "Thermal Design of Power GaAs FETs", in the book *GaAs FET Principles and Technology,* Eds: Dilorenzo et al, p. 313-347, Artech House, 1982

[5] Greg Pfeiffer (HP), program "ictherm", unpublished.

[6] Chin C. Lee, "Thermal Analysis of Integrated Circuit Devices and Packages" *IEEE Trans. Compon., Hyb. and Mfg. Techn.* Vol 12, No 4, p. 701-709, Dec 1989

[7] Tom Knotts (HP), program "fe", unpublished

[8] K.M. Cham, K. Lee, et al *Computer-Aided Design and VLSI Device Development,* Kluwer Academic Publishers, Boston, MA, 1988, pp.136-140

[9] James Waldrop, K.C. Wang, and Peter Asbeck, "Determination of Junction Temperature In AlGaAs/GaAs HBTs by Electrical Measurement", submitted for publication