Tesi di Dottorato

Università degli Studi di Napoli "Federico II"

Dipartimento di Ingegneria Biomedica
Elettronica e delle Telecomunicazioni

Dottorato di Ricerca in
Ingegneria Elettronica e delle Telecomunicazioni

Development of a new electro-thermal simulation tool for RF circuits

---

Jerzy Piotr Nowakowski

Il Coordinatore del Corso di Dottorato
Ch.mo Prof. Giovanni Poggi

Il Tutore del Dottorato
Ch.mo Prof. Niccolò Rinaldi

A. A. 2008

Dla mojej Rodziny.
For my family.

Preface

Thermal interactions set a tremendous challenge for electronic designers, and a device/system can not be considered anymore as purely electrical one, since exists a strong interaction between operating temperature and electrical operating point. As a consequence, the electronic industry has a huge necessity of *electro-thermal simulation tools*, which will be able to take both thermal and electrical interactions into account during a design process.
The Ph. D. research activity was pursued towards the development of electro-thermal simulation tool based on commercially available software by Agilent, called Advanced Design System (ADS). The ADS is currently the electronic industry standard tool for Radio-Frequency simulation. The task has been completed successfully, and the working electro-thermal code has been created.

In particular, the outline of the thesis is as follows:

Chapter 1 Describes in details the importance of the electro-thermal simulation and the limitations of current simulation tools.

Chapter 2 Describes the thermal models used in electro-thermal simulation and the Compact Thermal Model developed during the Ph. D. stage in LAAS-CNRS laboratory in Toulouse, France.

Chapter 3 Describes the development stages of the electro-thermal simulation tool based on ADS environment for RF circuits.

Appendix Describes the optimisation algorithm for automated thermal impedance network identification.

Contents

List of Figures

List of Tables

CHAPTER 1

The relevance of the electro-thermal phenomena

Thermal problems in electronics have become one of the highest importance due to several technological trends, which contribute to consider thermal aspects in a reliable, modern electronic design.
Following the well-known Moore's law[1] [15], the enhanced device integration pursued in order to favour high current capability and higher operation speed, has led to a steady increase (Fig. 1.1) in power density [1, 16], which in consequence contributes

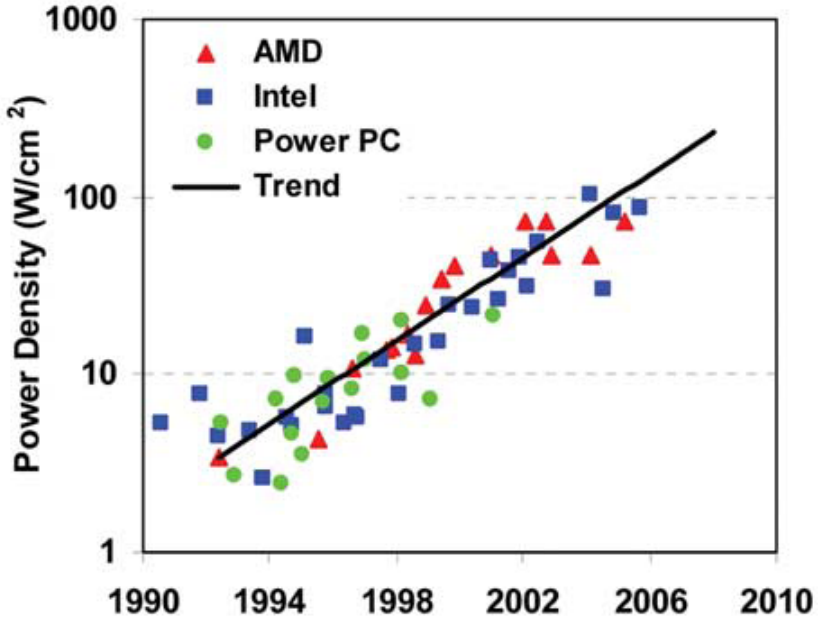


Figure 1.1: Power density on-chip trend. Source: [1].

to even higher device/system operational temperature [17]. As an example, the scaling

[1] the number of transistors per chip double every 24 months

in the CMOS technology, used for building e.g. microprocessors (Fig. 1.2), has been observed for recent 40 years. The first Intel 4004 microprocessor presented in 1971 had 2250 transistors on a 12mm^2 Silicon chip surface. The dissipated power in the device was negligible. The recent Intel products (Fig. 1.2) consist of around 2 billions

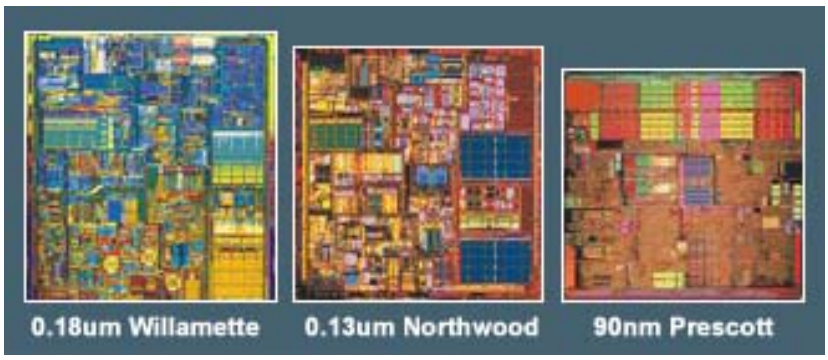
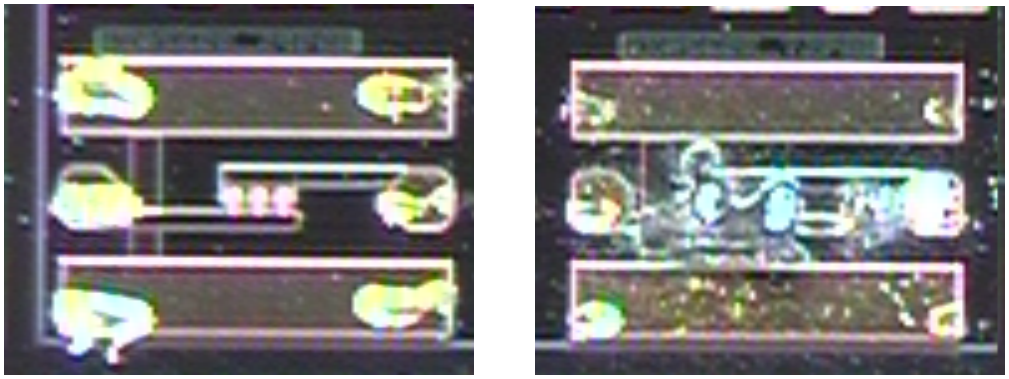


Figure 1.2: The comparison of Intel Pentium 4 family core. Increasing the amount of transistors on an even smaller surface causes serious problems with temperature.

of transistors, and dissipate up to 150 W of power on a comparable or even smaller Silicon surface. It simply means that the system starts to behave like a "light bulb", and without constant cooling - immediately burns down.
However, not only the system level designs encounter temperature related reliability problems. On the device level, the thermal phenomena must be taken into account as well. As shown in Fig. 1.3, the three-finger Gallium-Arsenide HBT has been destroyed



(a) Correct device (b) Broken device

Figure 1.3: GaAs HBT (a) before, and (b) after thermally induced breakdown.

due to thermally induced breakdown.
In bipolar transistors, a device operating point is important when describing the thermal reliability problem. A transistor Safe Operating Area (SOA) (Fig. 1.4) is limited by various breakdown mechanisms, caused not only by thermal instability. In particular, the literature describes two types of breakdown [18]:

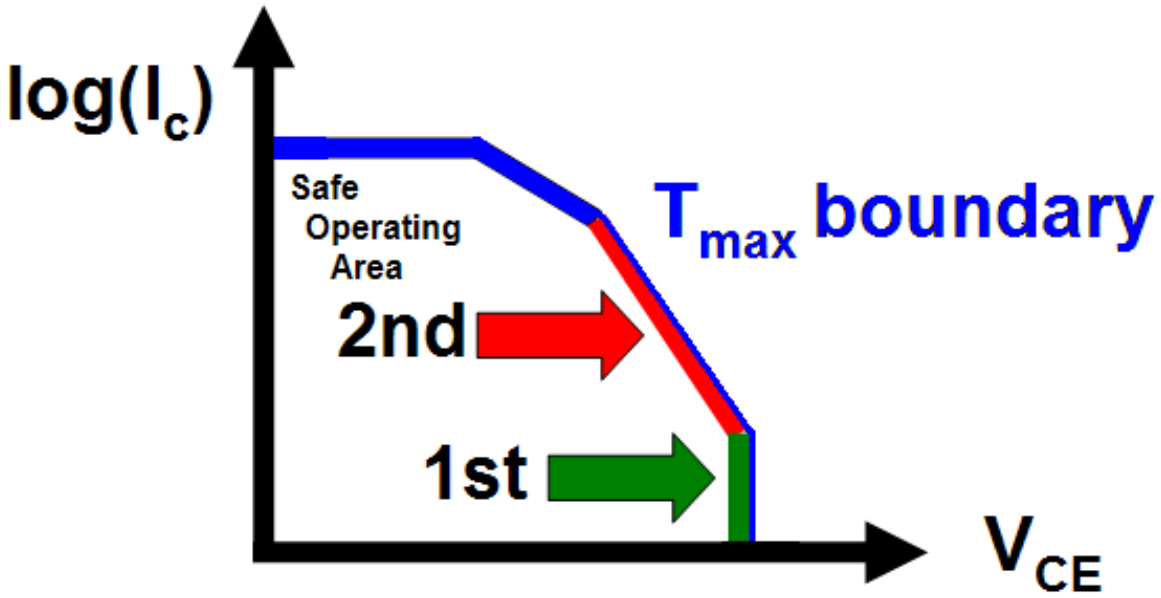


Figure 1.4: Transistor safe operating area is limited by first (1st) and second (2nd) breakdown.

1st breakdown is caused by avalanche multiplication, that is the multiplication of the carriers in the space-charge region.

2nd breakdown is caused by electro-thermal effects [19, 20] and/or impact ionization [21]. With respect to the first, avalanche breakdown, the second occurs for lower power levels.
The basic issues of breakdown mechanisms are roughly described in [22].

As shown in Fig. 1.4, the T_{max} boundary limits the SOA region. The SOA can be further minimised, due to device down-scaling. In that case, the SOA decreases while the current density increases (Fig. 1.5).
The increase in power density leads to higher operational temperature (Fig. 1.5), since the equation for the temperature increase above ambient depends on the thermal resistance R_{TH}, heat source size (for bipolar devices emitter size A_E), current density J_C and applied collector-emitter voltage V_{CE} (Eq. 1.1)

$$\Delta T = R_{TH} \cdot A_E \cdot (J_C \cdot V_{CE}) \tag{1.1}$$

With even higher current density related to the down-scaling issue, the necessity of even better electrical isolation performed on the device level is required. The technology of electrical isolation was evolving through several years, as described:

LOCOS (Local Oxidation of Silicon) used in many variations, like SILO (Sealed Interface Local Oxidation) [23], SWAMI (Side WAll Masked Isolation) [24], semi-recessed LOCOS, or poly-buffered LOCOS.

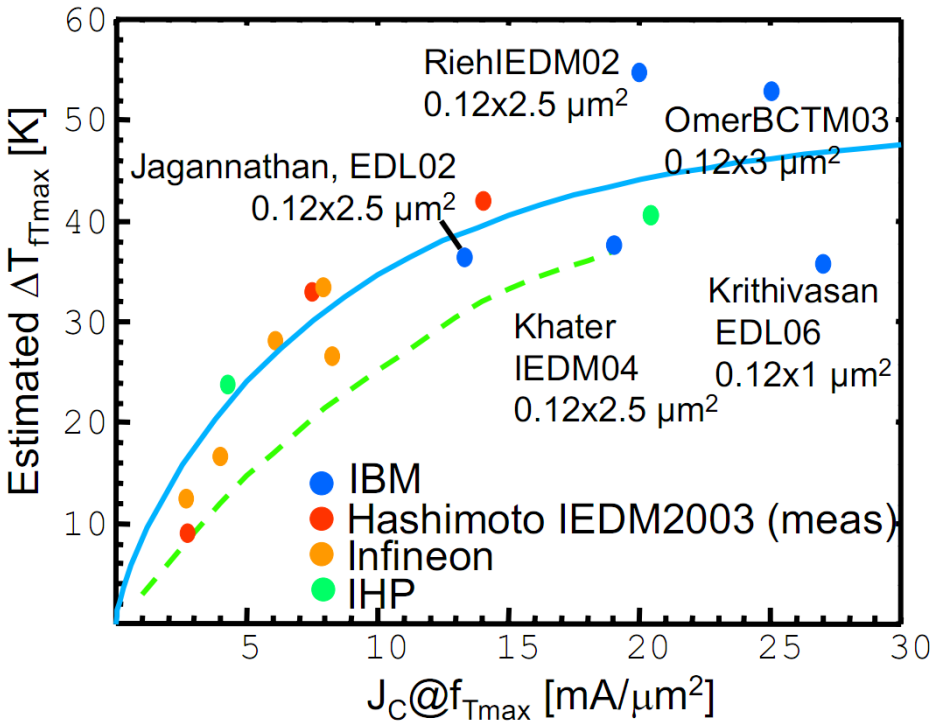


Figure 1.5: Estimated temperature increase above ambient for SiGe HBT devices. Source: [2, 3, 4, 5, 6, 5, 7, 8, 9, 10].

STI (Shallow Trench Isolation) performed for the devices built in technology nodes of 0.25 um and smaller.

DTI (Deep Trench Isolation) [25]

SOI (Silicon On Insulator) , being a modification of SOS (Silicon on Sapphire), with the developed recently SSOI (Strained SOI). SOI technology has become an industry standard for devices with channel width less than 180 nm from year 2000. [26, 27]

SOG (Silicon on Glass) [28] where thermal aspects are of the utmost importance [14, 29].

Abovementioned isolation techniques help to reduce the parasitics [30, 31, 32], leakage currents [33], and improve noise immunity for the cost of unwanted thermal isolation and even lower thermal conductivity [11] as shown in Fig. 1.6 With SOG technology the values of thermal resistance are even higher, (Tab. 1.1). Next to the technology down-scaling, the factor responsible for thermal problems is the use of new materials, which with respect to Silicon, represent better electrical properties for the of significantly smaller thermal conductivity value. As an example the application of Gallium-Arsenide (GaAs) in modern RF circuits, which thermal conductivity is around four times lower than that of Silicon (Tab 1.2).

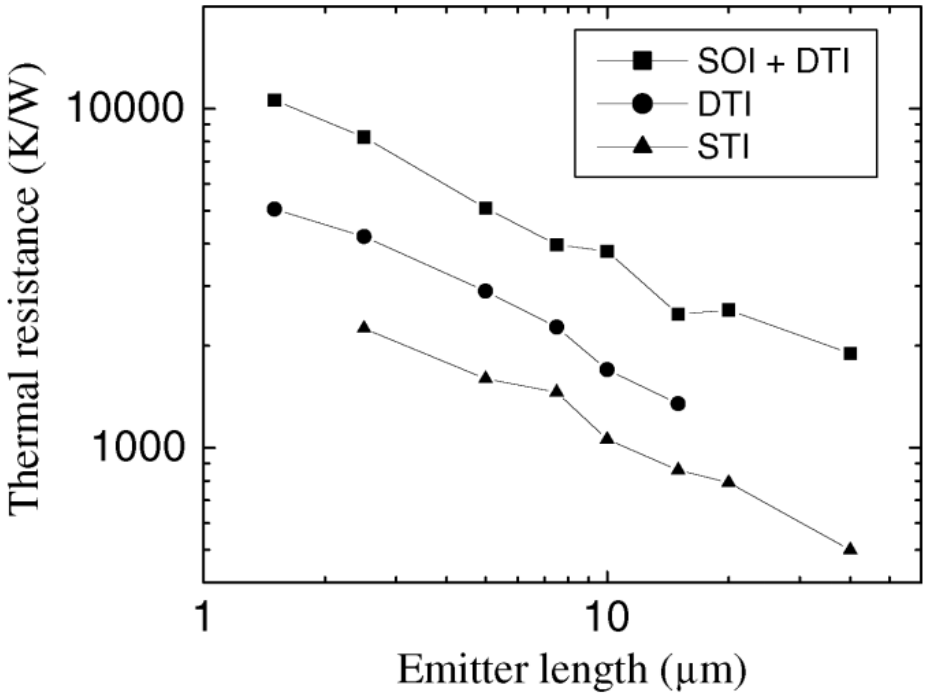


Figure 1.6: Measured thermal resistance as a function of emitter length with constant 0.25 μm emitter width. Source: [11]

Table 1.1: Comparison between bulk Si and SOG transistor. Source: [14]

NPN transistor process	bulk Si	SOG
Device area [um^2]	$21 \cdot 25$	$8 \cdot 21$
Emitter area [um^2]	$20 \cdot 1$	$20 \cdot 1$
h_{FE} (V_{BE}=0.8V)	100	100
V_A [V]	12	12
r_e [Ω]	3	3
r_b [Ω]	60	60
r_c [Ω]	20	3
BV_{CEO} [V]	5	5
$C_{bc}(V_{CB} = 0)$ V [fF]	75	30
$C_{eb}(V_{CB} = 0)$ V [fF]	88	88
R_{TH} [K/W]	300	10500

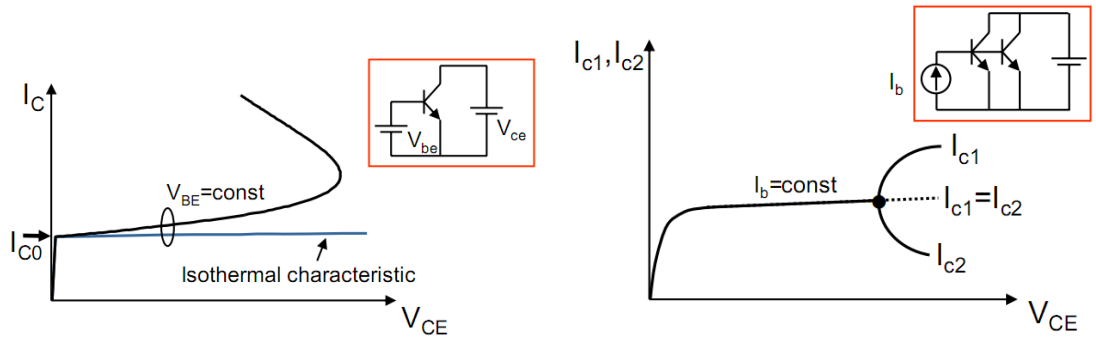
Table 1.2: Estimated thermal conductivity for various materials used/to be used in electronics.

Material	Thermal conductivity @300K [W/K · m]
GaAs	46
Si	148
Ge	60
Si_xGe_x	11-85
SiC	360-490
SiO_2	1.38
Si_3N_4	68
InP	68
InAs	27.3
GaP	110
GaN	130
GaSb	32
InAs	80
InSb	18
$Al_xGa_{1-x}As$	$0.55 - 2.12x + 2.48x^2$
AlN	285
BN	740
C (Diamond)	600-2000
$GaAs_{1-x}Sb_x$	-
InN	45-176

The temperature increase above ambient may cause electro-thermal effects, like self-heating), mutual-heating (thermal coupling), affecting the device operation modes like:

DC (Direct Current, steady-state) In bipolar devices, both BJTs and HBTs, the behaviour is modified with respect to the isothermal one, as shown in Fig. 1.7, For bipolar (both BJTs and HBTs) single-finger transistors, at a constant base-emitter voltage V_{be}, so called fly-back (snapback) is observed [19] (Fig. 1.7a). For multi-finger (or parallelled) devices, at a constant base current, the current bifurcation occurs [20] (Fig. 1.7b). Contrary to HBTs, the BJTs have positive temperature coefficient and the output current I_C rises as the collector-emitter voltage V_{CE} increases, as shown in Fig. 1.7b. For HBTs, the temperature coefficient is negative, and Fig. 1.7b would represent the decreasing collector current vs. the collector-emitter voltage V_{CE} until the bifurcation occurs. The hot-spots may occur [1].

AC (Alternating Current), small signal analysis Since usual designing technique requires the DC analysis in order to check the biasing points, which are distorted due to self-heating effects, both steady-state and the frequency dependence of small signal parameters are modified.



(a) Fly-back (snapback) behaviour (b) Current hogging (bifurcation)

Figure 1.7: Electrothermal effects in single (1.7a), and multi-finger bipolar devices (1.7b).

Transient (Dynamical) The time-domain analysis considers the *thermal impedance* idea, which is simply the variable thermal resistance in time. The behaviour of thermal impedance may be roughly approximated with an exponential curve, which at the end reaches the DC value. The case will be described in the Chapter 2.

1.1. The electro-thermal analysis idea

In solid-state devices, the current (and hence the dissipated power) is a function of device temperature, which in turn, is determined by the dissipated power. Therefore the determination of device current (i.e. power) and a temperature represents a coupled *electro-thermal* problem. The electro-thermal analysis can be performed in several

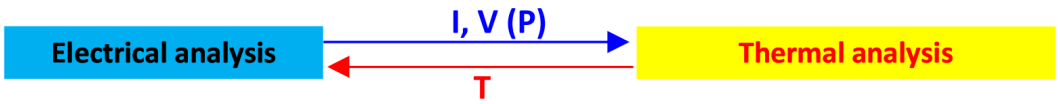


Figure 1.8: Electro-thermal simulation scheme.

ways, since both electrical and thermal problems are coupled, and must be solved continuously (Fig. 1.8). As described in the chapter 2, the thermal problem can be solved using several approaches, however solving the heat diffusion equation at each iteration point is time-consuming and it is difficult to automate such a process. The most efficient way is to evaluate a thermal matrix in advance, using one of the methods described in the Chapter 2. Like it was mentioned, the thermal nonlinearities can be further included using the Kirchoff transform [34]. The scheme of an electro-thermal simula-

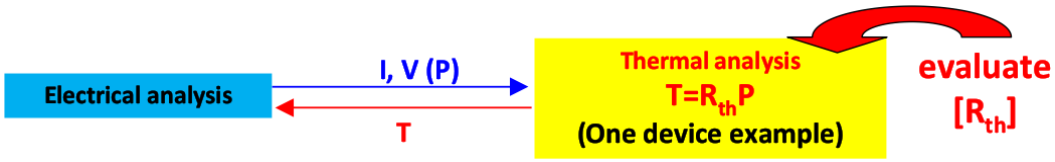


Figure 1.9: Electro-thermal simulation scheme based on thermal resistance.

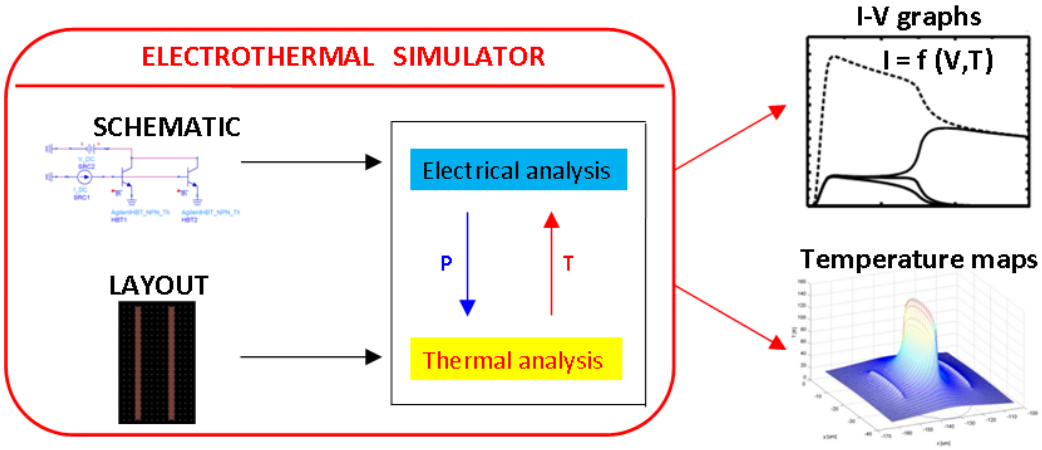


Figure 1.10: Electro-thermal simulation scheme performed by a circuit simulator.

tion performed by a circuit simulator is shown in Fig. 1.10. The thermal information is extracted from a layout, and the electrical data is saved in a schematic. After the electro-thermal simulation, the results are plotted in Current-Voltage graphs or temperature maps. Such a simulation can be performed using a modern circuit simulator like e.g. ADS. Some simulation tools are unable to perform this task, due to certain limitations within their structure. These limitations are briefly described in the Sec. 1.2.

1.2. The limitations of present simulation codes

Engineers involved with the design and development of solid-state devices and circuits have to consider thermal aspects as having the same importance as purely electrical ones. In principle, designers can resort to electro-thermal simulation tools in order to accomplish this complex task. Unfortunately, traditional programs such as the widely used simulator SPICE were realized for the analysis of integrated circuits, and are unsuitable for this purpose, since they do not account for self-heating (the temperature of any active device is assumed to be constant, that is, independent of dissipated power) and temperature gradients (the temperature of the circuit is specified by the user, and is

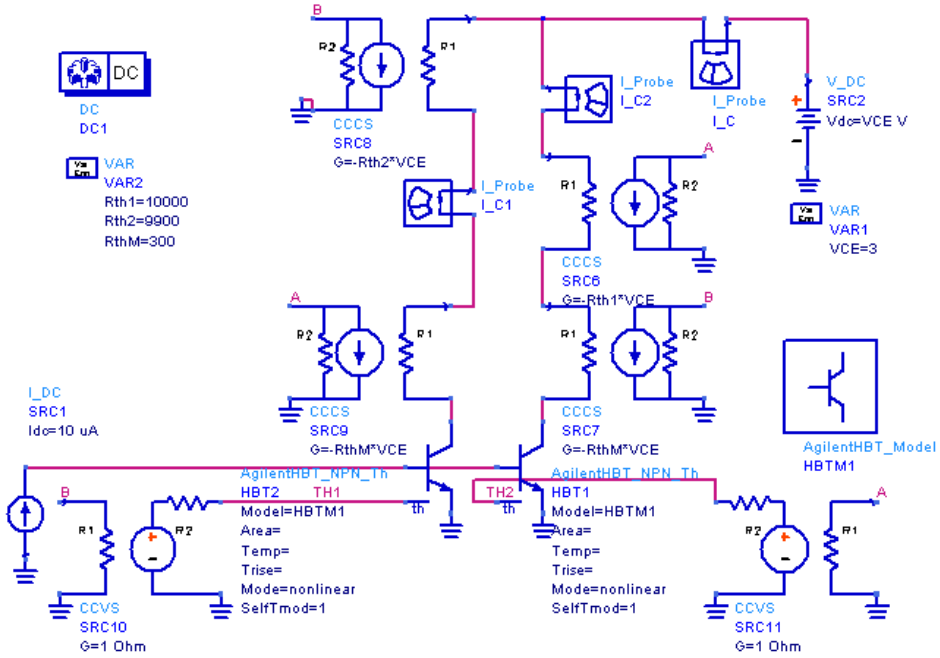


Figure 1.11: SPICE macro-modelling approach for two parallel transistors. The electro-thermal feedback is created via CCVS and VCVS.

uniform in the entire circuit under analysis). A commonly adopted approach to extend the SPICE capability toward the electro-thermal simulation of devices and circuits is the so-called macromodeling technique. The structural macromodeling method starts from the intrinsic SPICE model as a main core and describes some specific nonlinear device effects and the electro-thermal feedback by adding supplementary standard SPICE elements (i.e., resistors, capacitors, inductors, diodes, and transistors). The basic advantage of the structural macro-models is their manage-ability in all SPICE versions. However, they often require long analysis time, are associated to complicate parameter extraction techniques and it is difficult to model the hard nonlinearities. As an alternative, it is possible to employ the Analog Behavioural Modeling (ABM)-based macromodeling technique presented in Fig. 1.11, which makes almost entirely use of voltage-controlled voltage/current sources that enable the direct "in line" implementation of any kind of nonlinear expression. This allows tackling most of the problems arising with the structural macromodeling. Unfortunately, the computational time nonlinearly rises with the number of ABM sources; as a consequence, it is necessary to keep this number as small as possible; this might be solved by not accounting for certain effects that have no influence or no interest for a given simulation. Nevertheless, it is evident that such SPICE-like tools can be computationally viable only for circuits where the number of active devices is relatively low. To overcome all the above shortcomings a novel simulation code is hereby presented based on the commercially available Advanced System Design (ADS) software from Agilent. Contrarily to SPICE, tools like ADS incorporate recently developed bipolar transistor models that include the possibil-

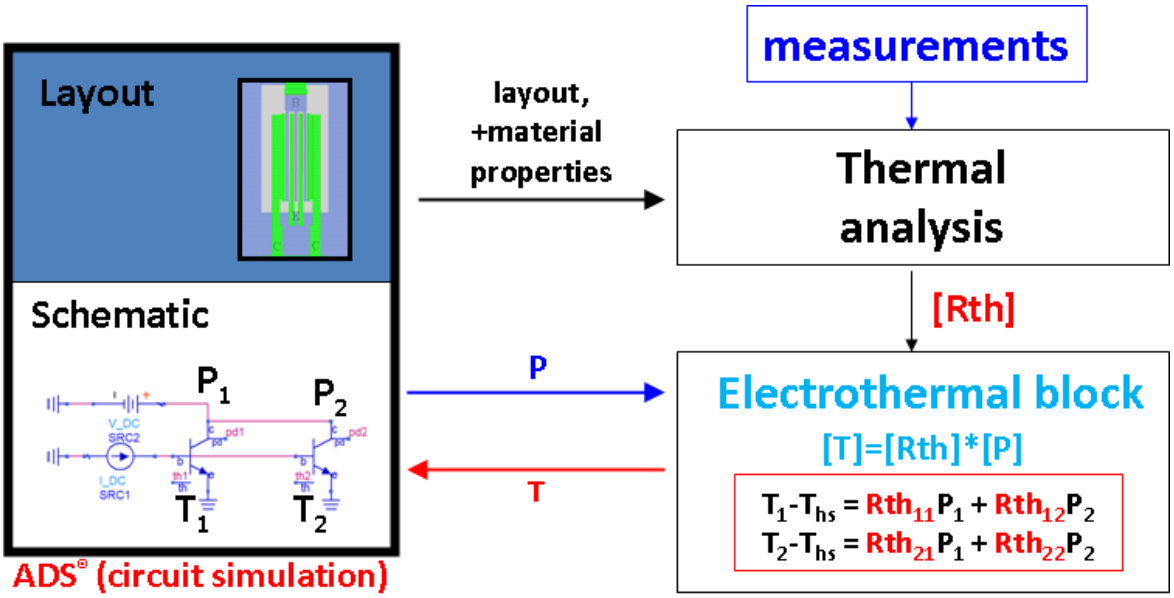


Figure 1.12: Electro-thermal simulation with ADS from Agilent. Both self and mutual interactions are included.

ity to activate the self-heating option (see e.g., the model MEXTRAM 504 for bipolar junction transistors). Such models are equipped with a supplementary terminal, namely a "thermal node", and include a default value for the thermal self-resistance. Hence, the temperature increase above ambient is evaluated from the dissipated power and considered, in turn, as a further input that modifies the thermally-sensitive parameters (electro-thermal feedback). However, thermal interactions between active devices integrated in the chip are not accounted for, which represents a considerable limitation for the electro-thermal simulation of high-density ICs.
The possible solution to the problem could be the development of new schematic components, so called Electro-thermal Feedback Blocks (ETFBs), as shown in Fig. 1.12. Correctly constructed ETFB could take into account both types of electro-thermal interactions that is for self and mutual ones.

CHAPTER 2

Thermal models for the electro-thermal simulation

A device thermal model is created solving the heat equation. It is not an easy task to accomplish, since the final solution depends on:

- Geometry.
- Boundary Conditions.
- Nonlinear phenomena

The heat-flow equation can be solved using:

Numerical Methods such as Finite Element Method (FEM), Boundary Element Method (BEM), and Finite Differences (FD), and Thermal Networks are able to deal with arbitrary complex geometries, including nonlinear thermal effects. Numerical methods require the highest computational effort for solving the heat flow equation with respect to other ones. Additionally it is not easy to iteratively automate these methods, since certain parameters may change from step to step. As a consequence, an intensive manual labour is required, even in commercial tools.

Analytical Methods . There are two fundamental methods to solve the steady-state heat equation:

1. *Separation of variables method.* In this approach the temperature solution is given in form of a double infinite series of trigonometric function. For this reason this method is also known as the *Fourier Series* approach. The computational efficiency of this method is related to the number of terms that have to be included in the series in order to achieve a given accuracy. This depends on the heat source-to-chop area ratio. Since this ratio is typically large in bulk devices, a high number of terms must be included in the summations.

2. *The image function method* (e.g. [35]). In this approach the temperature solution is first determined by neglecting the boundary conditions at the lateral and bottom boundaries . This implies that in this approximation one treats the heat source as if it were in a semi-infinite domain. It has been shown that it is possible to finish a simple closed-form solution for rectangular heat sources (see section 2.3.1 and 2.3.2). The second step is to incorporate the effect of boundary conditions. This is done by introducing an infinite number of fictitious heat sources. As a result, the temperature solution is expressed as an infinite sum of terms, each accounting for a fictitious image source. Note that if the heat source-to-chip area ratio is small, so the effect of boundary conditions will be negligible in practice and a small number of terms (or even just one one term) need to be included. Therefore this method is very efficient for small devices.

In section 2.2, the problem of solution of the heat-flow equation will be described from the mathematical point of view, introducing possible approximations. The idea of thermal resistance and thermal impedance will be demonstrated.

2.1. Assumptions for analytical thermal model

The steady-state heat-flow equation is described as follows:

$$\bigtriangledown [k(T) \bigtriangledown T] + g(x, y, z) = 0 \tag{2.1}$$

where $k(T)$ is the thermal conductivity, T is the temperature and $g(x, y, z)$ is the power density (which depends on a position) per unit volume [W/cm^3].

To solve the heat-flow equation (Eq. 2.2) one needs to specify:

- How the heat is exchanged with the environment (boundary conditions).
- The generation of the heat (power density g function).

Usually this problem is resolved by simplifying both:

Power Density. The active area where the heat is generated, is assumed of a simple geometry (parallelepiped or rectangle), with a power density g function, that is (1) constant and uniform in the heat source area (2) zero outside the heat source area.

Boundary Conditions (B.C.). The boundary conditions are simplified as follows:

1. on the top surface is assumed the adiabatic boundary condition (since the flow through passivation and metal contacts is neglected).
2. on the lateral faces is assumed a reasonable approximation
3. on the bottom an ideal heat sink is assumed, that is a constant temperature (isothermal boundary condition).

Nonlinear phenomena. Nonlinear phenomena is neglected, that is the thermal conductivity $k(T)$ becomes a constant parameter k.

2.2. Chip thermal model - mathematical considerations

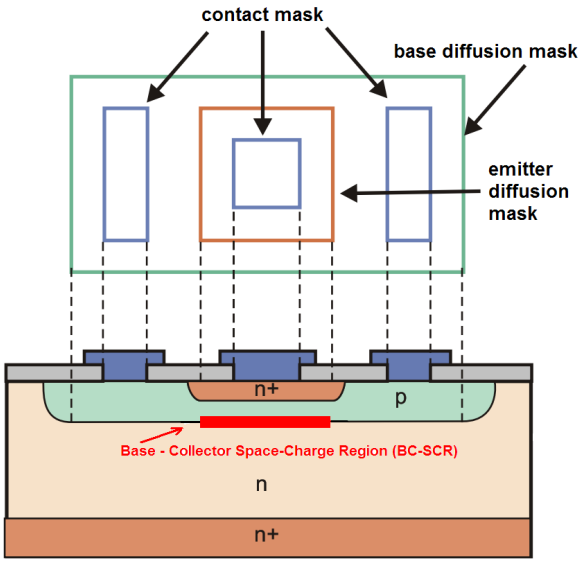


Figure 2.1: Base-Collector Space Charge Region in bipolar NPN device.

The steady-state heat-flow equation is described as follows:

$$\bigtriangledown [k(T) \bigtriangledown T] + g(x, y, z) = 0 \tag{2.2}$$

The nonlinear thermal conductivity dependence on temperature is neglected, so $k(T)$ becomes a constant parameter k. As a result, the steady-state heat-flow equation becomes linear partial-differential equation (PDE):

$$\bigtriangledown^2 T + \frac{g(x, y, z)}{k} = 0 \tag{2.3}$$

The linearity is assumed, however in a final solution, by applying the *Kirchhoff transform*, the nonlinear solution can be received.

For bipolar devices, the heat generation occurs in BC-SCR[1] region. (Fig. 2.1). A

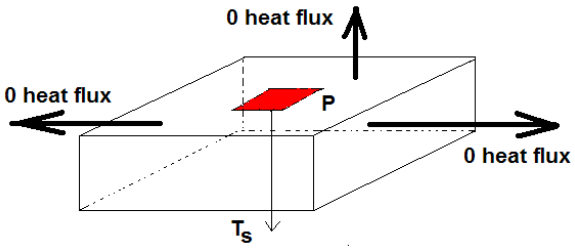
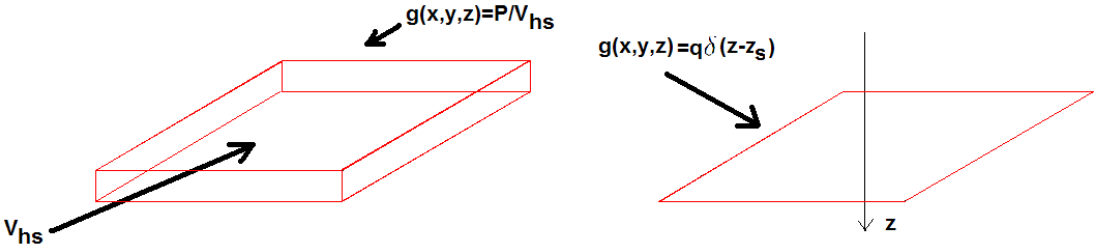


Figure 2.2: Assumptions for constructing the thermal domain for VHS and THS models.

heat is located underneath the emitter window (n region). As an immediate consequence, the heat source can be approximated as a rectangular parallelepiped volume centred at a depth z_s. In a more simple approximation, the heat source is treated as an infinetly thin rectangle. In both cases the power density is assumed to be uniform. Therefore, two cases will be considered:



(a) Volume Heat Source (VHS). (b) Thin Heat Source (THS).

Figure 2.3: BC-SCR region can be approximated by Volume or Thin Heat Sources.

- Volume Heat Source (VHS), where the power density g inside the parallelepiped is defined as:

$$g(x,y,z) = \frac{P}{V_{HS}} \quad [W/cm^3] \tag{2.4}$$

where P is a power density and $V_{HS} = H \cdot W \cdot L$. W is the BC-SCR width, L is the length and H thickness.

- Thin Heat Source (THS), where the power density g is defined by Eq. 2.5

$$g(x,y,z) = q \cdot \delta(z - z_s) \quad [W/cm^3] \tag{2.5}$$

where $q = P/WL$ represents the power density per unit area.

[1]Base-Collector Space-Charge Region

Both models are explained in detail in section 2.3. Once the geometry of the heat source has been specified we need to introduce suitable boundary conditions which specify how heat is exchanged with the environment:

➤ **Boundary Conditions.** Usually simplified B.C.[2] are assumed in the thermal model.

1. Adiabatic B.C. at the top and bottom surface (zero heat flux at these surfaces):
$$\frac{\delta T}{\delta n} = 0 \Rightarrow g(x) = 0, g(y) = 0 \tag{2.6}$$
where $n = x, y$)
2. Isothermal B.C. at the bottom:
$$T = T_s \tag{2.7}$$
This B.C. applies when the chip is in contact with an ideal heat sink.

Now, if the new variable θ is introduced, Eq. 2.2 becomes:

$$\theta = T - T_s = \Delta T \tag{2.8}$$

and heat-flow equation reads:

$$\bigtriangledown \theta + \frac{g}{k} = 0 \tag{2.9}$$

And Eqs. 2.7, 2.8 acquire the form of *homogeneous B.C.*:

$$\text{homogeneous B. C.} = \begin{cases} \delta\theta/\delta n = 0 \\ \theta = 0 \end{cases} \tag{2.10}$$

where n is the outward pointing normal.

It can be noted that, the problem given by Eq. 2.9 with homogeneous B.C. (Eq. 2.10) is linear and the temperature is proportional to the dissipated power.:

$$\theta(x, y, z) = \rho(x, y, z) \cdot P \tag{2.11}$$

where $\rho(x, y, z)$ is a function of position, heat source and chip geometry. The junction temperature increase above ambient θ_j is usually defined as an average of the temperature increase θ over the active volume V_A or active surface S_A:

$$\theta_j = \begin{cases} VHS \Rightarrow \frac{1}{V_A} \iiint_{V_A} \theta \, \mathrm{d}x \, \mathrm{d}y \, \mathrm{d}z \\ THS \Rightarrow \frac{1}{S_A} \iint_{S_A} \rho \, \mathrm{d}x \, \mathrm{d}y \end{cases} \tag{2.12}$$

[2]Boundary Conditions

For instance in FET devices S_A represents the channel region. In bipolar transistors the active region is represented by the base. Therefore S_A can be taken as a rectangular area $W \cdot L$ located at the emitter-base junction.

Finally, using Eq. 2.11 the junction temperature can be described by expressed as:

$$\theta_j = R_{th} \cdot P \tag{2.13}$$

where R_{th} is called the thermal resistance.

Concluding, the junction temperature T_j of a device, can be calculated from the product of thermal resistance and dissipated power:

$$T_j - T_s = R_{th} \cdot P \tag{2.14}$$

2.2.1. Case of multiple heat sources

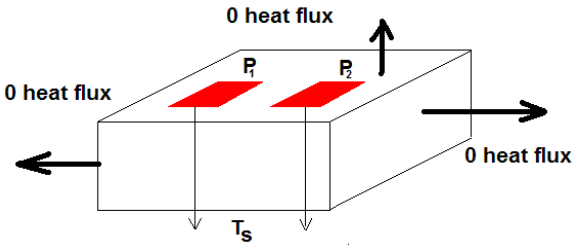


Figure 2.4: Assumptions for constructing the thermal domain for VHS and THS models in case of multiple heat sources.

For the case of multiple heat sources, let us consider the simplest problem, when only two heat sources are present (Fig. 2.4). The steady heat flow equation is described by:

$$\bigtriangledown \theta + \frac{g_1}{k} + \frac{g_2}{k} = 0 \tag{2.15}$$

where $g_1 = P_1/V_1$ and $g_2 = P_2/V_2$ represent the power densities which are assumed to be uniformly dissipated in the BC-SCR of the two devices subject to the homogeneous B.C. (Eq 2.10)

Since the temperature dependence of thermal conductivity can be neglected, the problem can be described using the superposition of effects, that is:

$$\begin{gathered} \theta = \theta_1 + \theta_2 \\ \bigtriangledown^2 \theta = \bigtriangledown^2 (\theta_1 + \theta_2) = \bigtriangledown^2 \theta_1 + \bigtriangledown^2 \theta_2 \end{gathered} \tag{2.16}$$

In particular, the problem for the 1st heat source (with dissipated power P_1 is described as follows:

$$\begin{cases} \bigtriangledown^2 \theta_1 + g_1/k = 0 \\ \text{Homogeneous B.C.} \end{cases} \tag{2.17}$$

and for the 2nd heat source (with dissipated power P_2):

$$\begin{cases} \bigtriangledown^2\theta_2 + g_2/k = 0 \\ \text{Homogeneous B.C.} \end{cases} \tag{2.18}$$

Applying the superposition method:

$$\bigtriangledown^2 \theta_1 + \bigtriangledown^2\theta_2 = -\frac{g_1}{k} - \frac{g_2}{k} \tag{2.19}$$

Finally a sum of linear terms is obtained:

$$\theta = \rho_1(x, y, z) \cdot P_1 + \rho_2(x, y, z) \cdot P_2 \tag{2.20}$$

As a result, the junction temperature for the two devices can be expressed as:

$$\begin{aligned} \theta_1 &= R_{th,11}P_1 + R_{th,12}P_2 \\ \theta_2 &= R_{th,21}P_1 + R_{th,22}P_2 \end{aligned} \tag{2.21}$$

where R_{th} is thermal resistance. Diagonal elements $R_{th,11}$ and $R_{th,22}$ represent the self-heating and non-diagonal $R_{th,12}$ and $R_{th,21}$ mutual one. P is dissipated power. As a consequence, Eq. 2.22 can be recast in matrix form as a product of thermal resistance matrix R_{th} and dissipated power vector P.

$$[\theta] = [R_{th}] \cdot [P] \tag{2.22}$$

2.2.2. Analysis of the time-dependent case

Considering the domain shown in Fig. 2.2, the time-dependent heat flow equation is as follows:

$$\frac{1}{\alpha} \quad \frac{\delta\theta}{\delta t} - \bigtriangledown\theta^2 = \frac{g(t, x, y, z)}{k} \tag{2.23}$$

where α represents the thermal diffusivity and k is the thermal conductivity. In Eq. 2.23 we assume that material parameters are constant and temperature independent. In addition to B.C. (Eq. 2.10) we must specify an initial condition:

$$\begin{cases} T(t = 0, x, y, z) = T_0(x, y, z) = T_s \\ \theta = T - T_s \end{cases} \Rightarrow \theta(t = 0, x, y, z) = 0 \tag{2.24}$$

By taking the Laplace transform we obtain:

$$\frac{s}{\alpha}\overline{\theta}(s, x, y, z) = \bigtriangledown^2\overline{\theta} = \frac{\overline{g}(s, x, y, z)}{k} \tag{2.25}$$

Let us now consider the temperature distribution for a constant (time-independent) dissipated power $g(t, x, y, z) = const.$ in the BC-SCR, then:

$$\overline{g} = \frac{g}{s} = \frac{P}{V_{HS}s} \tag{2.26}$$

The partial differential equation (PDE) in the Laplace domain can be written as:

$$\frac{s}{\alpha}\overline{\theta} - \bigtriangledown^2\overline{\theta} = \frac{g}{s \cdot k} \tag{2.27}$$

Since the above boundary value problem is linear, the solution can be expanded as:

$$\overline{\theta}(s, x, y, z) = \overline{\rho}(s, x, y, z)\frac{g}{s} \tag{2.28}$$

where the function $\overline{\rho}$ depends on the heat source and chip geometries. By defining the junction temperature by a suitable average value (see Eq. 2.22) we obtain:

$$\overline{\theta_j}(s) = Z_{th}(s) \cdot P \tag{2.29}$$

which in a time domain is as follows:

$$\theta_j(t) = Z_{th}(t) \cdot P \tag{2.30}$$

where Z_{th} it the thermal impedance and describes the temperature response to a step application of power $P(t) = P \cdot u(t)$ (see Fig. 2.5). . Similarly to the steady-state

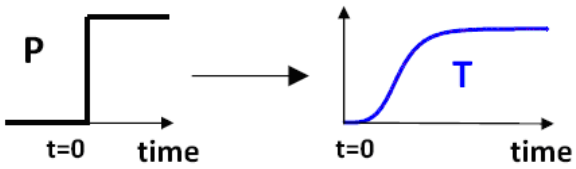


Figure 2.5: Thermal response of a circuit for a step application of power $P(t) = P \cdot u(t)$

case, the thermal impedance concept can be extended to the analysis of multiple heat sources. Thus in the case of two dissipating devices we have:

$$\begin{aligned} \theta_{j,1}(t) &= Z_{th,11}(t) \cdot P_1 + Z_{th,12}(t) \cdot P_2 \\ \theta_{j,2}(t) &= Z_{th,21}(t) \cdot P_2 + Z_{th,22}(t) \cdot P_2 \end{aligned} \tag{2.31}$$

Concluding, the time-dependent junction temperature $T_j(t)$ of a device, can be calculated from the product of thermal impedance $Z_{th}(t)$ and dissipated power:

$$T_j(t) - T_s(t) = Z_{th}(t) \cdot P \tag{2.32}$$

Summary In aforementioned derivations shown in section 2.2, has been clearly demonstrated, that a temperature junction T_j can be described using a product of thermal matrix R_{th}/Z_{th} and a dissipated power P. For a steady-state electro-thermal simulation a thermal resistance matrix R_{th} is used, and for transient case a thermal impedance matrix Z_{th}.

2.2.3. Thermal resistance

A thermal resistance R_{th}, necessary for a steady-state electro-thermal simulation, can be evaluated prior the thermal analysis in several ways:

Experimentally — using measurement techniques. The thermal matrix is obtained directly from the measurement results.

Numerically — solving the heat-flow equation. The thermal resistance matrix can be calculated using numerical programs for thermal simulation like e.g. ANSYS or COMSOL. In this case, a resulting thermal matrix can be precisely calculated (e.g. including complex boundary conditions and nonlinear phenomena), for the cost of high computational effort. Further it is hard to automate such a process by calling directly a numerical thermal program, from a circuit simulator.

Analytically — solving the heat-flow equation, taking into account a device geometry (circuit's layout) and other thermal parameters. Analytical solutions to a heat equation can be automated within a circuit simulator (like ADS), since the thermal matrix can be obtained based on few thermal parameters and circuit layout. In addition analytical calculation can be computationally effective with respect to numerical one.

As shown in previous section 2.2, thermal resistance can be represented in a matrix form if more than one heat source is taken into account. In that case, the diagonal of the matrix represents the self-heating effects, and other matrix elements are responsible for mutual interactions, as shown in Fig. 2.6. If the heat sources have equal thermal

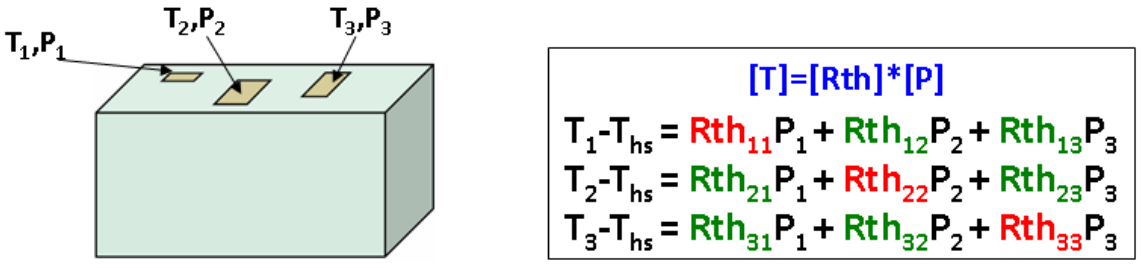


Figure 2.6: Case for three heat sources: Self (red) and mutual (green) thermal resistances.

parameters (e.g. size, material), the thermal resistance matrix becomes symmetric with respect to the diagonal.

2.2.4. Thermal impedance

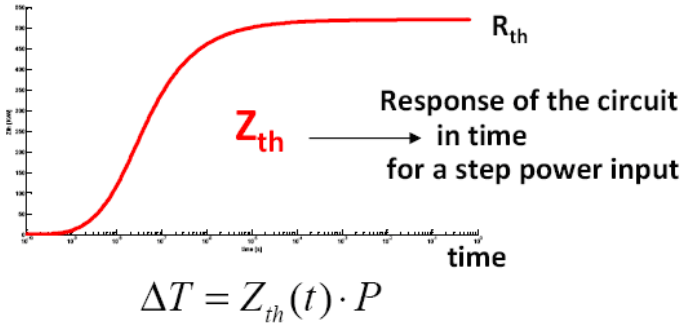


Figure 2.7: Thermal impedance as a response to a step application of power, logarithmic time scale.

From the definition, thermal impedance Z_{th} is a response to a step application of a power. Final value of thermal impedance always reaches steady state value R_{th}, as shown in Fig. 2.7.

Since an equivalence of electrical and thermal quantities exists, a popular way to approximate the thermal impedance is by means of electrical thermal impedance, that is by an infinite amount of RC pairs. In this way, by means of electrical equivalence,

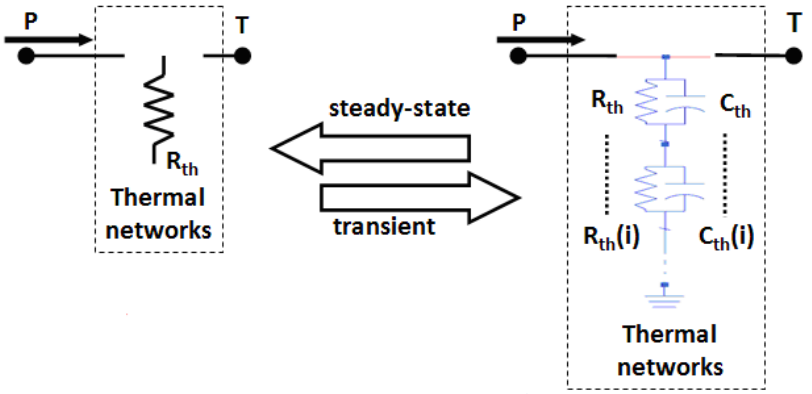
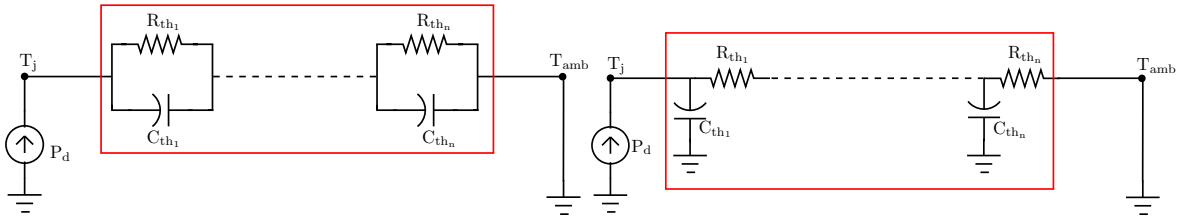


Figure 2.8: Schematic pass from steady state to transient case.

thermal phenomena can be described using circuit simulation environment. However, for computational reasons it is not possible to include infinite amount of RC pairs in a circuit simulator. The less amount of RC pairs is used, the better for a simulation time for a cost of accuracy. In these circumstances, a reasonable trade-off must be found.

A well known representation for thermal impedance is Foster (Fig. 2.9a) and Cauer (Fig. 2.9b) RC series. Both networks can be easily transformed between each other [13].



(a) Foster RC network (b) Cauer RC network

Figure 2.9: Common implementation of thermal circuit in commercial simulators. The dissipated power P_d is a current and the junction temperature T_j is a voltage. Ambient temperature T_{amb} is connected to ground.

Foster network

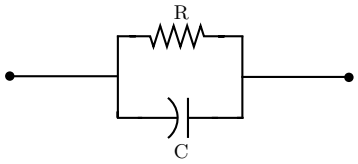


Figure 2.10: Simplest Foster RC network.

In Foster network, shown in Fig. 2.10, the basic cell is a parallel connection of one resistor and one capacitor. In the *Laplace domain*, the relationship for the impedance of such cell is the following:

$$Z_{th}(s) = R \parallel \frac{1}{sC} = \frac{R \cdot \frac{1}{sC}}{R + \frac{1}{sC}} = \frac{R}{1 + sRC} = \frac{\frac{1}{C}}{s + \frac{1}{RC}} \tag{2.33}$$

Performing the *Inverse Laplace Transform* of 2.33, the main advantage of Foster network becomes clearly visible:

$$Z_{th}(t) = R(1 - exp(-t/(R_{th}C_{th}))) \tag{2.34}$$

where C_{th} is a thermal capacitance. In the case of n RC pairs we obtain

$$Z_{th}(t) = \sum_{i=1}^{n} R_{th,i}(1 - exp(-t/(R_{th,i}C_{th,i}))) \tag{2.35}$$

Introducing the characteristic time constants $\tau_i = R_{th,i} \cdot C_{th,i}$ the thermal impedance is expressed as:

$$Z_{th}(t) = \sum_{i=1}^{n} R_{th,i}(1 - exp(-t/\tau_i)) \tag{2.36}$$

where thermal impedance Z_{th} is represented as :

$$Z_{th} = \sum_{i=1}^{n} R_{th_i}(1 - exp(-t/\tau_i)) \tag{2.37}$$

The time constants value τ_i are important parameters due to the fact, a thermal system can be represented as *distributed linear RC system*, with wide range of time-constant values [36]. According to [37], the range of possible time-constant values in thermal systems can be divided into following ranges:

- **10μs-100ms:** semiconductor chip / die attach.
- **10-50ms:** package structures beneath the chip.
- **50ms - 1s:** further structures of the package.
- **1 - 10s:** package body.
- **10-10000s:** cooling assemblies.

In the appendix A, the algorithm for automated identification of thermal impedance using Foster networks has been developed, based on desired amount of RC pairs N, which are the input parameter for a routine. The routine has been implemented in the electro-thermal simulator presented in chapter 3.

Cauer network

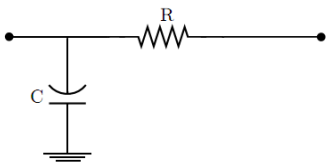


Figure 2.11: The simplest Cauer RC network.

The Cauer network is attractive from the thermal point of view, since provides more physical description with respect to the Foster one [13]. The problem with Cauer network is that it is easily defined in the Laplace domain using partial fraction expansion, however the time-domain equation is almost impossible to obtain using inverse Laplace transform. Concluding, the mathematical representation of Cauer networks is much more complicated than the Foster one.
For the simplest case, presented in Fig. 2.11, the impedance in the Laplace domain can be described as follows:

$$Z(s) = \frac{1}{sC + \frac{1}{R}} \tag{2.38}$$

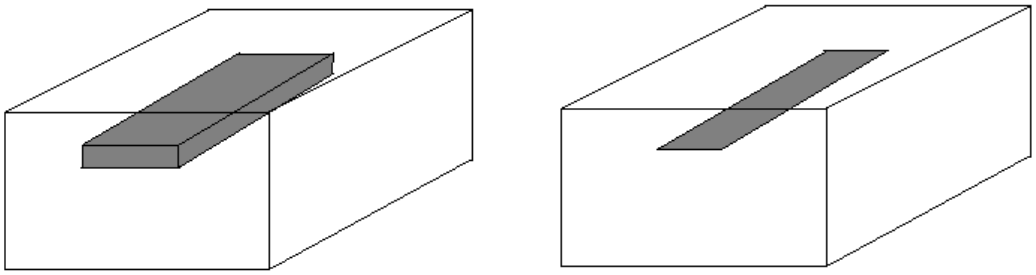
In case when N resistors and N capacitors are present, the network is described as:

$$Z_N(s) = \frac{1}{sC_1 + \frac{1}{R_1 + \frac{1}{sC_N + \frac{1}{R_N + \dots}}}} \tag{2.39}$$

As can be seen, Eq. 2.39 has a rather complex form, when the network is long. In practice it is not possible to perform the inverse Laplace transform in order to pass to the time domain, when the polynomial degree is higher than 4. An alternative to obtain the Cauer network could be to identify the curve, and translate it to a Foster network, using the method described in the appendix A. However during the translation process the resistances and capacitances of the Foster network will be different than for the Cauer one.

In the appendix A, an algorithm for automated identification of thermal impedance using Cauer networks has been developed, based on desired amount of RC pairs N, which are the input parameters for the routine. The routine has been implemented in the electro-thermal simulator presented in chapter 3.

2.3. Analytical thermal model



(a) VHS model. (b) THS model.

Figure 2.12: Analytical models: volume heat source (a) and thin heat source (b).

In bipolar devices working in forward active mode, the heating region is physically located in BC-SCR[3] (i.e. depletion) region, in which the scalar product between the vectors "current density" and "electric field" is positive. Therefore projection of the heat source on the top chip surface can be approximated geometrically with the emitter window. On the other hand, the heat source thickness is given by the BC-SCR width. Such a region can be thus approximated by the VHS (rectangular parallelepiped) model. Another approach is to use a THS located more or less on the metallurgical base-collector junction. As previously mentioned, in developing analytical thermal models, following approximations are made. In particular:

[3]Base-Collector Space-Charge Region

1. The region where power is dissipated is approximated by a simple geometry. Two models have been adopted:

   - **THS model** – an infinetly thin rectangle, Thin Heat Source model (Fig. 2.12b). This case is described in section 2.3.1.
   - **VHS model** – a parallelepiped with finite thickness, Volume Heat Source model (Fig. 2.12a). This case is described in section 2.3.2.

2. Power density is uniform within the heat source region and zero outside the region.

3. Boundary Conditions are neglected in all directions except the top surface, where adiabatic boundary condition is set. The image function method is used to solve the steady-state heat-flow equation.

4. The geometry of the heat source is independent on bias conditions.

2.3.1. Case of rectangular heat source (THS)

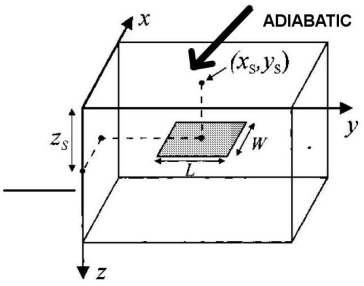


Figure 2.13: Thermal model of an integrated device. The chip is a right parallelepiped with a rectangular heat source of dimensions WL located at the depth z_s from the surface and centred around the point (x_s, y_s, z_s).

To develop a solution as a first step adiabatic boundary condition at the top surface is neglected. The temperature distribution in an infinite domain with a THS at a point P=(x,y,z) is given by [35] :

$$\theta(x,y,z) = \frac{q_s}{4\pi k}\left[g(\delta x_2,\delta y_2,\delta z) - g(\delta x_2,\delta y_1,\delta z) - g(\delta x_1,\delta y_2,\delta z) + g(\delta x_1,\delta y_1,\delta z)\right] \tag{2.40}$$

where g represents :

$$\begin{aligned} g(\delta x, \delta y, \delta z) = & -\delta z \; arctan \frac{\delta x \delta y}{\delta z \sqrt{\delta x^2 + \delta y^2 \delta z^2}} \\ & + \delta x \; log \left(\delta y + \sqrt{\delta x^2 + \delta y^2 \delta z^2} \right) \\ & + \delta y \; log \left(\delta x + \sqrt{\delta x^2 + \delta y^2 \delta z^2} \right) \end{aligned} \tag{2.41}$$

and $\delta x_1 = x - x_1$, $\delta x_2 = x - x_2$, $\delta y_1 = y - y_1$, $\delta y_2 = y - y_2$, $\delta z = z - z_s$. Here x_1, y_1, x_2, y_2, are coordinates of the rectangular heat source placed at a depth z_s.

Let us define the function ϕ:

$$\begin{aligned} \theta(x, y, z) &= \phi(\delta x_1, \delta x_2, \delta y_1, \delta y_2, \delta z) \\ &= \phi(x - x_1, x - x_2, y - y_1, y - y_2, z - z_s) \end{aligned} \tag{2.42}$$

To define the solution for the semi-infinite domain the adiabatic-condition at the top surface must be taken into account. By the image function method we add a fictitious image source placed at a symmetric position $-z_s$ with respect to the top surface. The temperature field generated by this image source is given by:

$$\phi(x - x_1, x - x_2, y - y_1, y - y_2, z + z_s) \tag{2.43}$$

By adding the contribution of the real and image source we obtain:

$$\theta_0 = \phi(x - x_1, x - x_2, y - y_1, y - y_2, z + z_s) + \phi(x - x_1, x - x_2, y - y_1, y - y_2, z - z_s) \tag{2.44}$$

which can be easily seen to satisfy the B.C. at the top surface.

2.3.2. Case of volume heat source (VHS)

The temperature distribution for an infinite domain with VHS is given by [35]:

$$\theta(x, y, z) = \frac{g}{4\pi k r} \left[\left[[l(\delta x, \delta y, \delta z)]_{\delta x_1}^{\delta x_2} \right]_{\delta y_1}^{\delta y_2} \right]_{\delta z_1}^{\delta z_2} \tag{2.45}$$

where g represents the power dissipated per unit volume and l represents the function:

$$\begin{aligned} l(\delta x, \delta y, \delta z) = & \frac{1}{2} \delta x^2 \; arctan \left(\frac{\delta y \delta z}{\delta x \sqrt{\delta x^2 + \delta y^2 \delta z^2}} \right) \\ & + \frac{1}{2} \delta y^2 \; arctan \left(\frac{\delta x \delta z}{\delta y \sqrt{\delta x^2 + \delta y^2 \delta z^2}} \right) \\ & + \frac{1}{2} \delta z^2 \; arctan \left(\frac{\delta x \delta y}{\delta z \sqrt{\delta x^2 + \delta y^2 \delta z^2}} \right) \\ & - \delta x \delta y \; log \left(\delta z + \sqrt{\delta x^2 + \delta y^2 \delta z^2} \right) \\ & - \delta x \delta z \; log \left(\delta y + \sqrt{\delta x^2 + \delta y^2 \delta z^2} \right) \\ & - \delta y \delta z \; log \left(\delta x + \sqrt{\delta x^2 + \delta y^2 \delta z^2} \right) \end{aligned} \tag{2.46}$$

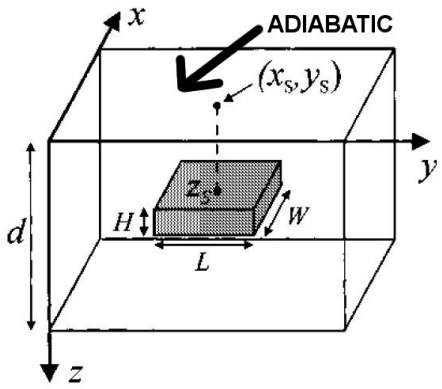


Figure 2.14: Thermal model of an integrated device with a volume heat source. The heat source is modelled as an embedded parallelepiped of dimensions WLH centred around the point (x_S, y_S, z_S).

As for the THS, let us define the function λ:

$$\theta(x, y, z) = \lambda(\delta x_1, \delta x_2, \delta y_1, \delta y_2, \delta z_1, \delta z_2) \tag{2.47}$$

and $\delta x_1 = x - x_1$, $\delta x_2 = x - x_2$, $\delta y_1 = y - y_1$,$\delta y_2 = y - y_2$, $\delta z_1 = z - z_1$, $\delta z_2 = z - z_2$ where x_1, y_1, x_2, y_2, z_1, z_2, are coordinates of the volume heat source.
The solution for semi-infinite domain is given by the superposition of the contributions of the real and image sources:

$$\begin{aligned}\theta_0(x, y, z) = \lambda(x - x_1, x - x_2, y - y_1, y - y_2, z + z_1, z + z_2) + \\ \lambda(x - x_1, x - x_2, y - y_1, y - y_2, z - z_1, z - z_2)\end{aligned} \tag{2.48}$$

2.4. Compact Thermal Model

An approach to the thermal analysis depends strongly on a kind of issue, which has to be solved. In the previous sections, the analytical thermal model has been described. The heat was exchanged through only one bottom boundary, with a constant temperature (isothermal boundary condition). Here, only one parameter (thermal resistance value) was necessary.
The thermal problem can be solved using more physically related quantity called *thermal resistance* R_{th} or *thermal impedance* Z_{th}. It is a simple approach, that can be easily implemented into electrical simulator. In this case, the heat flow is represented as a current, and a temperature is represented as a voltage. However, if the heat flow is

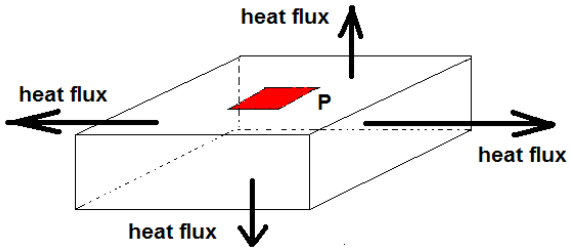


Figure 2.15: Compact Thermal Model can be created for complex domains.

more complex, and is described by n boundaries, one has to include $(n+1)^2$ parameters to solve the thermal problem [38]:

$$T_j = R_{th0}P + \sum_{i=1}^{n} a_i T_i \tag{2.49}$$

where a_i are the model parameters at *i-th* boundary, and T_i is the *i-th* boundary temperature. This mathematical formulation derived for description of thermal problem is called the *Compact Thermal Model (CTM)*.

It is important to note, that the CTMs do not have to be implemented only in a specialised thermal software like COMSOL or ANSYS, but also inside wide-used electrical simulators like SPICE. The example of equivalent thermal model, implemented within the electrical simulator is presented in Fig. 2.16.

2.4.1. General considerations

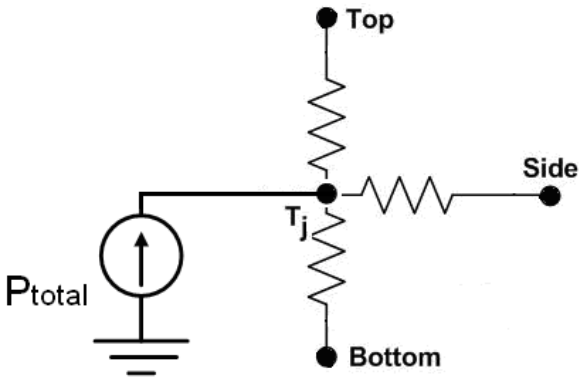


Figure 2.16: Device/System equivalent thermal model implemented within the electrical simulator by a Compact Thermal Model.

If a large scale system is considered with several boundaries: at top, bottom and lateral ones (e.g. a power MOSFET transistor together with a package, Fig. 2.15),

the creation of a 3D thermal model, based on the detailed physical structure is a cumbersome task.
Ideal CTM should be (1) accurate, (2) able to manage multiple heat sources and cooling surfaces, (3) taking nonlinear material properties into account, and (4) Boundary Condition Independent (BCI). Nevertheless, CTM structure has to remain simple, and easy to extract. Existing methodologies usually focus on a few of abovementioned qualities; otherwise, the resulting CTM would become too complex. Several methods have been already developed to provide accurate CTMs. One of them is the Delphi method [43, 44], which is able to give accurate static BCI CTMs for single-chip electronic devices and systems. An important drawback of Delphi, is the amount of 3D simulations [41]. In addition, dynamic models are difficult to extract. Delphi CTMs with more than two heat sources result in a very dense and excessively complex resistor network, where the number of 3D simulations or measurements becomes exorbitant. On the other hand, [45, 46] describe "traditional" CTM generation methods, with the way to make the thermal coupling between heat sources in static and dynamic modes. However in these methodologies, the CTMs are BCI only for systems with one cooling surface.
Beside that, the device manufacturers may not wish to publish the confidential data about the dimensions, materials and technology. However from the customer point of view, the knowledge of a thermal behaviour might be necessary. This problem has been noticed by the semiconductor industry [39, 40, 41, 42]. As the result, the standard equivalent thermal networks libraries have been developed, which are in use with standard CAD software. However, the drawback of this approach is the amount of 3D numerical simulations to be performed, in order to produce the CTM parameters.

A new CTM methodology will be demonstrated, which offers reduced amount of 3D simulations. The model has been under investigation during the PhD stage in LAAS-CNRS laboratory, Toulouse, France. Based on existing methods, the approach gives innovative solutions, in order to improve known CTM generation procedures. Firstly, a BCI CTM extraction method for systems with multiple cooling surfaces will be demonstrated. The CTM is an evolution of the star thermal network. Secondly, a new method conceived especially for multi-chip devices, i.e., multiple coupled heat sources [47, 48] will be presented. The thermal coupling is based on a definition of an Optimal Thermal Coupling Point (OTCP) between heat sources [49].

2.4.2. Methodology for Multi-Cooling Surface Structures

The star thermal network is a simple representation, which allows dealing easily as well with static as dynamic multiple cooling surface problems. In typical configuration the model (Fig. 2.17) has three cooling surfaces. Each one is represented with thermal resistance: R_{th_top}, R_{th_bottom} and R_{th_side}. The main drawback of the star model is with boundary conditions (R_{h_top}, R_{h_bottom} and R_{h_side}) change, the model becomes not valid. . The presented methodology is an extension of the star model representation, modified in order to deal with changing boundary conditions (BCI). The detailed

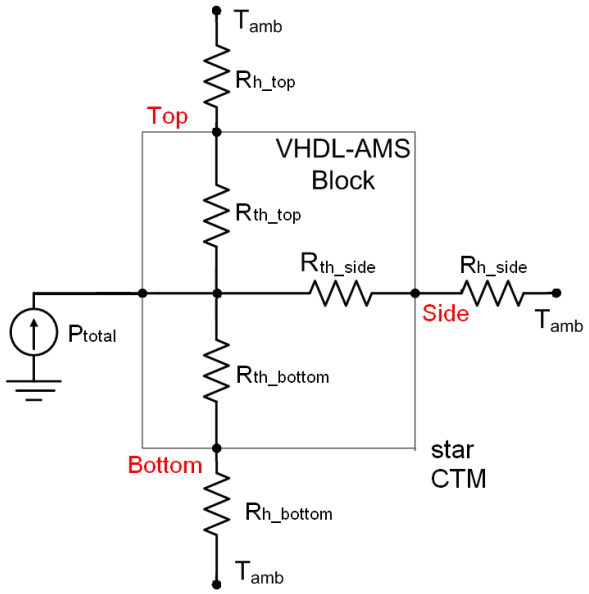


Figure 2.17: VHDL-AMS block presenting a star CTM and the boundary conditions applied (R_{h_top}, R_{h_bottom}, and R_{h_side})

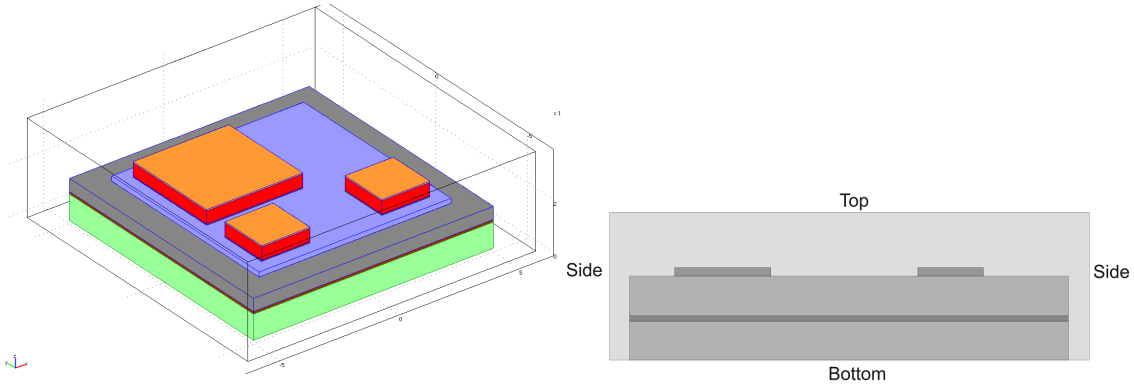
description of the star model can be found in [44], together with the Delphi model comparison. The thermal resistance between two isothermal surfaces is defined as in Eq. 2.50:

$$R_{th-x} = \frac{T_j - T_x}{P_x} \quad \left[\frac{K}{W}\right] \tag{2.50}$$

Being T_j and T_x the temperature of a junction and a cooling surface respectively, and P_x the heat power passing through surfaces. It is found that a thermal resistance between a junction and each cooling surface presents its minimum value when isothermal boundary condition is applied, while the rest of cooling surfaces are adiabatic. Moreover, all thermal resistances present their maximum value when isothermal condition is applied on every cooling surface. It is necessary to define thermal resistors, varying their values interactively with the applied boundary conditions on each cooling surface.

CTM extraction process

The CTM extraction methodology is illustrated on a device (Fig. 2.18) simulated with COMSOL. Only one of the three devices is dissipating heat. The dimensions are as follows: 12mm x 12mm x 3.8mm. The structure is composed by several layers of Si, Cu and Al_2O_3, enclosed in an epoxy package (Fig. 2.18a). The cooling surfaces are defined as Top, Bottom, and Side (Fig. 3b), being the last the sum of all side surfaces. Six thermal simulations with extreme cooling conditions are performed. A heat transfer coefficient h=20000 [W/m^2K] is applied to the following surfaces: (1) Top, (2) Bottom, (3) Side, (4) Top & Bottom, (5) Top & Side and (6) Bottom & Side, while (7) adiabatic



(a) Device intersection.

(b) Boundary conditions are applied to the packaged device.

Figure 2.18: The studied structure in COMSOL.

condition is applied to the rest of the structure. The following results show junction temperature, and heat power passing through cooling surfaces (Tab. 2.1). At this point,

Table 2.1: Results of 3D detailed simulation.

	T_j [°C]	P_{Top} [W]	P_{Bot} [W]	P_{Side} [W]
Cool. Top	146.4	100	0	0
Cool. Bot.	157.35	0	100	0
Cool. Side	177.32	0	0	100
Cool. Top&Bot.	109.33	51.75	48.16	0
Cool. Top&Side	122.54	57.13	04	2.53
Cool. Bot&Side	132.87	0	56.6	42.94

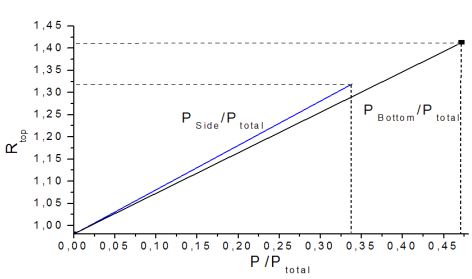
the thermal resistances corresponding to the three cooling surfaces Rth_{top}, Rth_{bottom}, Rth_{side} can be plotted versus the heat flux passing through the surfaces Side & Bottom, Side & Top and Bottom & Top respectively (Fig. 2.19). This is the key point to obtain a control equation for the values of the three resistors. Assuming thermal resistance values evolve linearly (Fig. 2.19), and the boundary conditions are applied as in Fig. 2.18, the variable thermal resistances are defined as follows:

$$R_{th\_top} = R_{top\_min} + \frac{P_{side}}{P_{total}}\alpha_{T-TS} + \frac{P_{bottom}}{P_{total}}\alpha_{T-TB} \tag{2.51a}$$

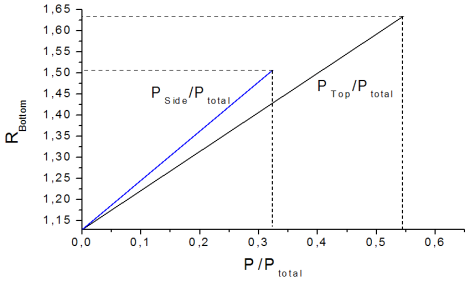
$$R_{th\_bottom} = R_{bottom\_min} + \frac{P_{side}}{P_{total}}\alpha_{B-BS} + \frac{P_{top}}{P_{total}}\alpha_{B-TB} \tag{2.51b}$$

$$R_{th\_side} = R_{side\_min} + \frac{P_{top}}{P_{total}}\alpha_{S-TS} + \frac{P_{side}}{P_{total}}\alpha_{S-SB} \tag{2.51c}$$

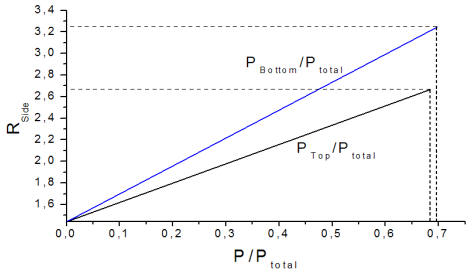
The R_{top_min}, R_{bottom_min} and R_{side_min} are the minimum values of the thermal resistance, i.e., while applying extreme cooling condition on the surfaces: Top, Bottom and



(a) R_{th_top} vs. (P_{Bottom}/P_{total} & P_{Side}/P_{total})



(b) R_{th_bottom} vs. (P_{top}/P_{total} & P_{Side}/P_{total})



(c) R_{th_side} vs. (P_{Bottom}/P_{total} & P_{Top}/P_{total})

Figure 2.19: Thermal resistances vs. dissipated heat plots.

Side. The factors α_{T-TS} with α_{T-TB} are the slopes of plot in Fig. 2.19a, α_{B-BT} with α_{B-BS} for Fig. 2.19b, and α_{S-TS} with α_{S-BS} in Fig. 2.19c. P is for heat power passing through a given surface. Three equations can express the junction temperature.

$$T_j = \left(R_{top\_min} + \frac{P_{side}}{P_{total}} \alpha_{T-TS} + \frac{P_{bottom}}{P_{total}} \alpha_{T-TB} \right) P_{top} + R_{h\_top} P_{top} \tag{2.52a}$$

$$T_j = \left(R_{bottom\_min} + \frac{P_{side}}{P_{total}} \alpha_{B-BS} + \frac{P_{top}}{P_{total}} \alpha_{B-TB} \right) P_{bottom} + R_{h\_bottom} P_{bottom} \tag{2.52b}$$

$$T_j = \left(R_{side\_min} + \frac{P_{top}}{P_{total}} \alpha_{S-TS} + \frac{P_{side}}{P_{total}} \alpha_{S-SB} \right) P_{side} + R_{h\_side} P_{side} \tag{2.52c}$$

In addition, the following equation must be also taken into account:

$$P_{total} = P_{top} + P_{side} + P_{bottom} \tag{2.53}$$

These control equations give an unique value for P_{top}, P_{bottom} and P_{side} that satisfy all boundary conditions. Then the new R_{th_top}, R_{th_bottom} and R_{th_side} values are obtained from their definition. However, if boundary conditions are changed on cooling surfaces, a new solution has to be calculated, in order to get a new value for the three thermal resistances.

The presented methodology allows obtaining simple BCI CTM, with a few 3D simulations.

Comparing the results for the generated CTM and 3D detailed model (COMSOL), a small error is still acceptable. Nevertheless, if extreme cooling conditions are applied on the three surfaces simultaneously, an error becomes above 10% and the results become meaningless. It is a consequence of the assumption, the thermal resistance values follow the linear characteristic (Fig. 2.19) with boundary condition changes.

Nonlinear control equations

It is possible to reduce the abovementioned error by adapting the control equations. The thermal resistance values do not actually evolve linearly as it has been assumed above. Fig. 2.20 shows the results for R_{bottom} for conditions between the two points. The curve shape would depend on the geometry of the studied structure and the value of applied convection. Although in many cases the linearity of contributions might be

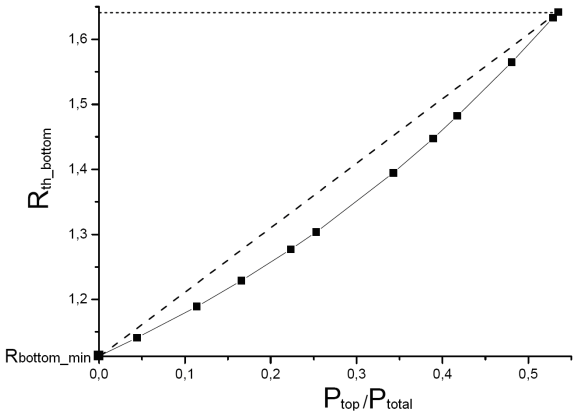


Figure 2.20: Actual R_{th_bottom} compared to linear assumption.

a very good approximation, some complexity can be added to the extraction procedure in order to obtain a much more precise model.

A series of 3D simulations have been carried out, the results fit well with the following law:

$$\Delta R_{th} = \frac{a}{b + \left(\frac{P_x}{P_{total}}\right)^c} \tag{2.54}$$

Then, the resistance definitions will be as follows instead of those in Eq. 2.51a:

$$R_{th\_Top} = \frac{a_{T-TB}}{b_{T-TB} + \left(\frac{P_{bottom}}{P_{total}}\right)^{c_{T-TB}}} + \frac{a_{T\text{ - }TS}}{b_{T-TS} + \left(\frac{P_{side}}{P_{total}}\right)^{c_{T-TS}}} - R_{top\_\min} \tag{2.55a}$$

$$R_{th\_Side} = \frac{a_{S-SB}}{b_{S-SB} + \left(\frac{P_{bottom}}{P_{total}}\right)^{c_{S-SB}}} + \frac{a_{S\text{ - }ST}}{b_{S-ST} + \left(\frac{P_{top}}{P_{total}}\right)^{c_{S\text{ - }ST}}} - R_{side\_\min} \tag{2.55b}$$

$$R_{th\_Bottom} = \frac{a_{B-BT}}{b_{B-BT} + \left(\frac{P_{top}}{P_{total}}\right)^{c_{B-BT}}} + \frac{a_{B-BS}}{b_{B-BS} + \left(\frac{P_{side}}{P_{total}}\right)^{c_{B-BS}}} - R_{bottom\_\min} \tag{2.55c}$$

And the equation system to solve the new power values will be:

$$T_j = \left[\frac{a_{T-TB}}{b_{T-TB} + \left(\frac{P_{bottom}}{P_{total}}\right)^{c_{T-TB}}} + \frac{\mathrm{a_{T\ -\ TS}}}{b_{T-TS} + \left(\frac{P_{side}}{P_{total}}\right)^{c_{T-TS}}} - R_{top\_\,\mathrm{min}} \right] P_{top} + R_{h\_top} P_{top} \quad (2.56a)$$

$$T_j = \left[\frac{a_{S-SB}}{b_{S-SB} + \left(\frac{P_{bottom}}{P_{total}}\right)^{c_{S-SB}}} + \frac{\mathrm{a_{S\ -\ ST}}}{b_{S-ST} + \left(\frac{P_{top}}{P_{total}}\right)^{c_{\mathrm{S\ -\ ST}}}} - R_{side\_\,\mathrm{min}} \right] P_{side} + R_{h\_side} P_{side} \quad (2.56b)$$

$$T_j = \left[\frac{a_{B-BT}}{b_{B-BT} + \left(\frac{P_{top}}{P_{total}}\right)^{c_{B-BT}}} + \frac{\mathrm{a_{B-BS}}}{b_{B-BS} + \left(\frac{P_{side}}{P_{total}}\right)^{c_{B-BS}}} - R_{bottom\_\,\mathrm{min}} \right] P_{bottom} + R_{h\_bottom} P_{bottom} \quad (2.56c)$$

To complete the equation system, Eq. 2.53 is still valid. The compact thermal model obtained this way is more reliable with the inconvenience that the number of thermal simulations is much bigger.

2.4.3. Multiple Heat Sources

The innovative dynamic CTM extraction method is especially conceived for multiple coupled heat sources, like multi-chip power electronics systems. The procedure is illustrated with one example based on a multi-chip power module manufactured by Freescale. The characterised device is a new intelligent power component for automotive applications, containing four smart MOSFETs (labelled HS0 to HS3 in Fig. 2.21). The transistors are controlled by a logical unit, integrated in the same package. .

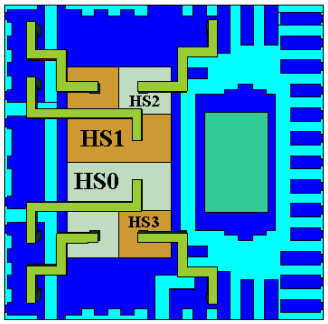


Figure 2.21: Model of the multi-chip component with active MOSFETs HS0 to HS3

Steady state CTM

The data for the CTM extraction is generated by 3D thermal transient simulations in various dissipating conditions of the MOSFETs using COMSOL (Fig. 2.21). The

multi-chip device temperature evolution is obtained on every chip. Due to the symmetric geometry of the system, the four heat sources CTM generation is simplified. In this case only two transient thermal simulations are necessary.

The example of the temperature results is clearly visible in Fig. 2.22.

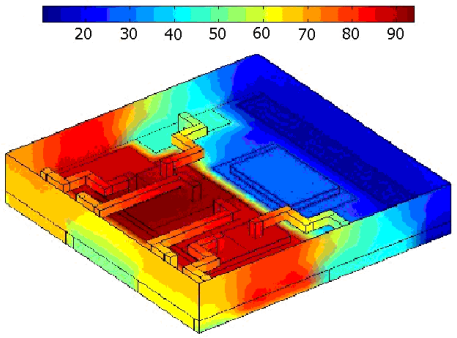
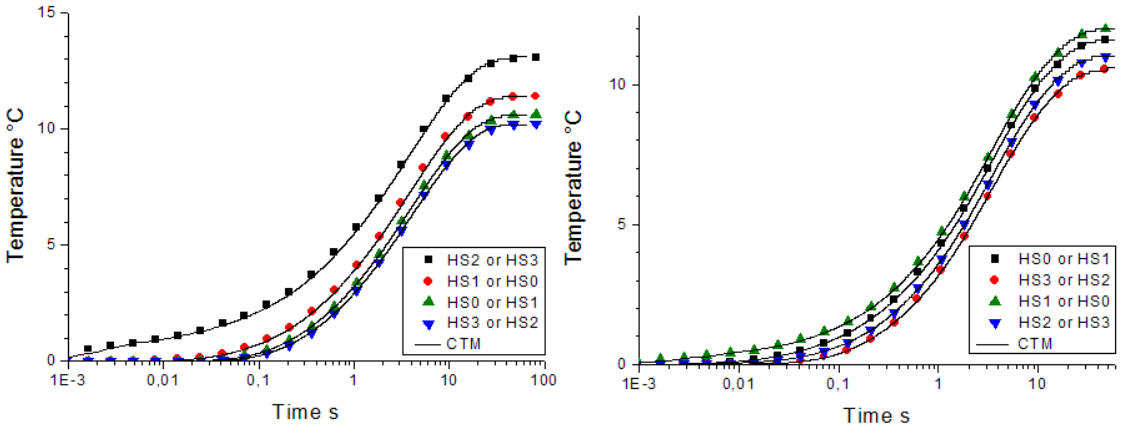


Figure 2.22: Example of 3D resulting temperature mapping after 100s, only HS1 is dissipating.

The thermal coupling is based on OTCP, which takes into account the mutual effect between heat sources. The OTCP is extracted as follows: (1) dissipating the power in one of the sources, and (2) taking the temperature of both active and inactive heat sources. Then, (3) the heat sources are swapped and the process is repeated. Knowing the dissipated power in each case, the steady state CTM can be extracted from equilibrium junction temperatures. Transient temperature responses of the MOSFETs are shown in Fig. 2.23. The steady state (equilibrium) temperatures are given in the Tab. 2.2. The steady-state CTM can be constructed using four branches of



(a) HS2 or HS3 dissipating 1W.

(b) HS0 or HS1 dissipating 1W.

Figure 2.23: Results of transient simulations obtained from COMSOL 3D (dots), together with CTM curve fitting (lines).

	HS0	HS1	HS2	HS3
HS0	12.2 °C	11.05 °C	10.62 °C	11.41 °C
HS1	11.05 °C	12.2 °C	11.41 °C	10.62 °C
HS2	10.57 °C	11.62 °C	13.06 °C	10.23 °C
HS3	11.62 °C	10.57 °C	10.23 °C	13.06 °C

Table 2.2: Equilibrium temperatures corresponding to the transient thermal curves in Fig. 2.23. The diagonal corresponds to self-heating. P = 1W

thermal resistances in series, with a current source (Fig. 2.24). In order to consider the interactive effect between heat sources, the following procedure is applied: First, for the case where only the device HS0 is dissipating, the result from Fig. 2.23b is considered. The active heat source HS0 is attached to the source node of its corresponding branch. Other inactive sources are represented as nodes, and placed in decreasing temperature order: HS3, HS1, HS2. Each node is defined as the coupling point between devices.

The steady-state CTM can be constructed using four branches of thermal resistances in series, with a current source (Fig. 9). In order to consider the interactive effect between heat sources, the following procedure is applied: First, for the case where only the device HS0 is dissipating, the result from Fig. 8b is considered. The active heat source HS0 is attached to the source node of its corresponding branch. Other inactive sources are represented as nodes, and placed in decreasing temperature order: HS3, HS1, HS2. Each node is defined as the coupling point between devices. The thermal resistance between the node C0 and coupling point C0-3 is defined as:

$$R_{0(0-3)} = \frac{T_{C0} - T_{C0-3}}{P} \tag{2.57}$$

TC0 is the temperature of the node C0; T0-3 is the temperature of the coupling point (C0-3) between HS0 and HS3, where P is the dissipated power in the HS0 source. The thermal resistance between coupling points C0-3 and C0-1 is defined as:

$$R_{0(3-1)} = \frac{T_{C0-3} - T_{C0-1}}{P} \tag{2.58}$$

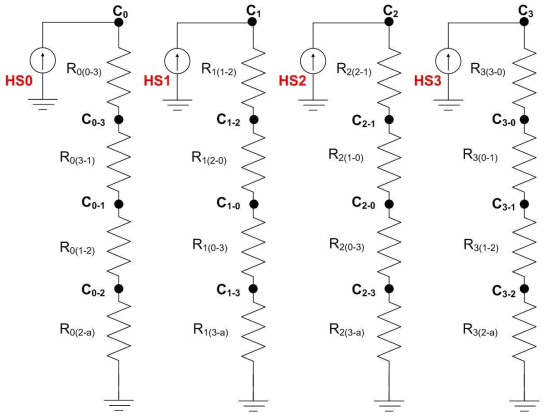
and so for the rest of resistances in the HS0 branch:

$$R_{0(1-2)} = \frac{T_{C0-1} - T_{C0-2}}{P} \tag{2.59a}$$

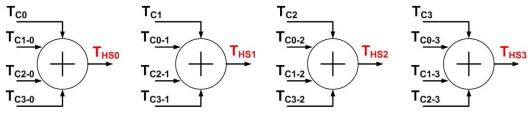
$$R_{0(2-a)} = \frac{T_{C0-2} - T_{C0-a}}{P} \tag{2.59b}$$

where index a is for ambient.

The same procedure is carried out by dissipating power only in HS1, HS2 and HS3 devices. Thermal resistance values in the respective branch are extracted. For the example described here, the thermal resistance values in the Tab. 2.3 are obtained.



(a) The thermal coupling points.



(b) Heat source temperatures resulting from the superposition of every heat source.

Figure 2.24: Steady state CTM.

As illustrated in Fig. 2.24a, at the bottom of each branch, the model establishes the temperature of each heat source as the sum of a resulting temperature in coupling points for every branch. For example, the actual temperature of HS0 device is due to its self and mutual heating.

$$T_{HS0} = T_{C0} + T_{C1-0} + T_{C2-0} + T_{C3-0} \tag{2.60}$$

HS0	HS1	HS2	HS3
R0(0-3)=0.4	R1(1-2)=0.4	R2(2-1)=1.65	R3(3-0)=1.65
R0(3-1)=0.57	R1(2-0)=0.57	R2(1-0)=0.79	R3(0-1)=0.79
R0(1-2)=0.48	R1(0-3)=0.48	R2(0-3)=0.39	R3(1-2)=0.39
R0(2-a)=10.57	R1(3-a)=10.57	R2(3-a)=10.23	R3(2-a)=10.23

Table 2.3: Thermal resistance values for all branches in Fig. 2.24.

Dynamic CTM

The extension from static to dynamic model is performed as follows: (1) The Cauer network is used (Fig. 2.25), that is the thermal capacitances connected to the ground

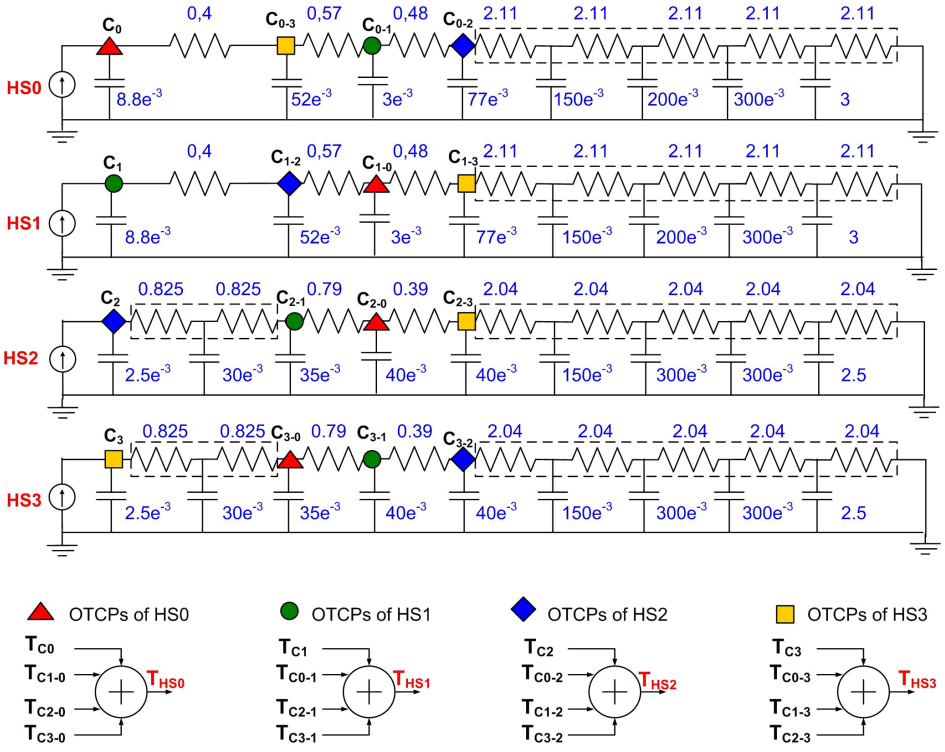


Figure 2.25: Transient CTM, resulting of the extension of the static model.

are added to each node. (2) The thermal resistance values and the coupling points of the static model are kept. (3) To increase the precision of the model, each resistance of the static model is divided into several ones in series, if its value is bigger than 10% of the total response.

The steady state behaviour of the structure of Fig. 2.25 will be identical to that of Fig. 2.24. The capacitances are optimised to fit the reference curves in Fig. 2.23.

The model validation has been carried out using the dissipation conditions in the Tab. 2.3. The results and simulated behaviour are compared in Fig. 2.23. It is found that the deviation with respect to the COMSOL 3D thermal simulations varies in 2% maximum.

Finally, the model results are in good approximation with the system thermal behaviour. Comparison between 3D simulations and our CTM prove the validity of OTCP principle also for transient modelling.

CHAPTER 3

Development of the electro-thermal simulation tool

The main goal during the development of the electro-thermal simulator code, was to make its code compatible with the industry standard RF simulator Advanced Design System (ADS) from Agilent Technologies.
The electro-thermal simulator has experienced several stages of development. The 1st stage was based on external command line code. The 2nd version of the simulator has been incorporated as an integral part of ADS environment. Further, the 3rd version of code has been based on external, GUI[1] routines. More details about applied strategies can be found in section 3.1.

The significant improvements have been performed passing from 1st to the 2nd version of the simulator, since new strategy for dissipated power calculation have been developed, and as a consequence new ETFB[2] types.
In particular, in the 1st, version of the electro-thermal simulator, a dissipated power was calculated externally, according to the definition presented in the Eq. 3.1 and in Fig. 3.1. It required ETFB type, which was performing a calculation of a dissipated power.

$$P_{diss} = P_{in} \cdot V_{in} + P_{out} \cdot V_{out} \tag{3.1}$$

From the 2nd approach, a dissipated power is calculated internally by a Compact

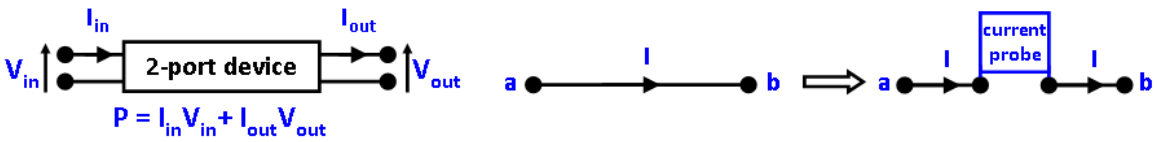


Figure 3.1: External power dissipation requires circuit splitting in order to obtain the input and output currents.

[1]Graphical User Interface
[2]Electro-Thermal Feedback Block

Device Model. The model can be distributed with ADS software or either developed as a custom one and supplied in the library called "Design Kit". Compact Device Models except a thermal node T have also a dissipated power P one, as shown in Fig. 3.2. More detailed study is available in section 3.3.1. As an immediate consequence, a strategy

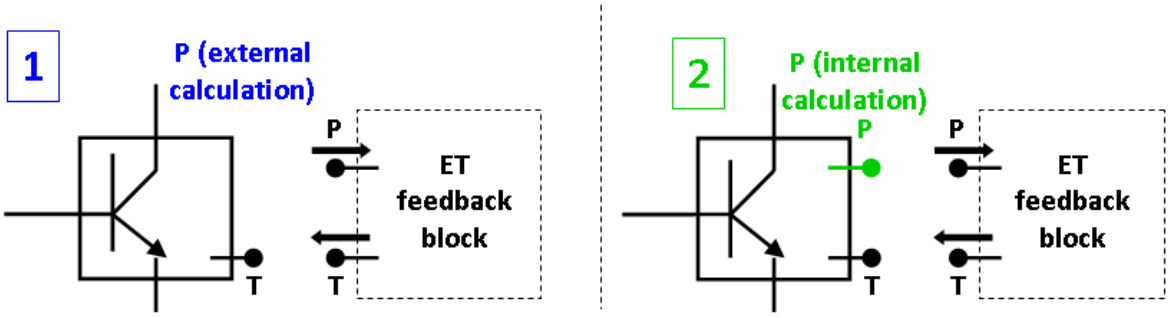


Figure 3.2: Dissipated power calculation is performed internally by a device model embedded within ADS software.

for creating electro-thermal feedback blocks (ETFBs) had to be changed. From now on an ETFB do not has to calculate a dissipated power within their structure. The detailed study of the dissipated power calculation inside an ETFB is as follows:

External power calculation performed within ETFB, requires four nodes for each device attached to ETFB: two for currents, one for voltage and one for temperature increase above ambient. Finally, for each device, which was taking part in electro-thermal phenomena, four nodes in block were used. For details refer to Fig. 3.5 on the page 45.
For ETFB creation, the Verilog-A language is used, which is the modern behavioural language devoted to description of the analog circuit.

Internal power calculation performed inside a device model (not within ETFB) requires only two nodes for each device attached to ETFB. The first one is an input of the dissipated power, which is already calculated, and the second one is the output, that is the temperature increase above ambient. Further, the internal dissipated power calculation performs the calculation correctly not only for DC, but also for AC and transient analyses [50]. The ETFBs devoted to internal dissipated power calculation are created using:

- Verilog-A language.
- SDDs (Symbolically Defined Devices). The Symbolically-Defined Device is an equation-based component that enables to quickly and easily define custom, non-linear components. These components are multi-port devices that can be modelled directly on a schematic. User defines an SDD by specifying equation that relate port currents, port voltages and their derivatives. Equations can also reference the current flowing into another device. Once a

model is defined, it can be used with any circuit simulator in ADS. Derivatives are automatically calculated during the simulation.

- Equivalent networks — the traditional, Spice-like approach. In this case ADS uses its embedded components like resistances, current controlled current sources (CCCS) and current controlled voltage sources (CCVS). The portability of the approach is restricted to the ADS platforms. For the transient simulation it is necessary to use external preprocessor, which calculates thermal impedance parameters (thermal resistance Rth and thermal capacitance Cth).
  The detailed description of ETFB related to Internal power calculation is described in section 3.3.2 on page 54.

3.1. Applied strategies

External command-line code. The approach has been published at the IEEE MIEL 2006 Conference [51]. This solution has been applied only to bipolar transistors. The maximum amount of devices connected to the feedback blocks is 25. Due to external power calculation, the strategy exhibits the "node-splitting" problem, and can be applied only to DC solution [50].

Internal ADS code. The approach, realized completely within ADS environment has following issues:

- The dissipated power is calculated internally.
- With respect to the external command-line approach, the amount of connected devices to the single ET block is increased up to 50.
- Various ETFBs are used, developed for internal power calculation.

External GUI code. The approach is based on external GUI[3]. Its main feature is the possible extension to other commercial simulators (e.g. Mentor Graphics "Eldo" or Cadence). As in the previous approach, the same ET feedback components are used.

3.2. External command-line approach

In this approach, the software is subdivided into three basic blocks, namely:

[3]Graphical User Interface Routine

- Pre-processing routine.
- The simulator ADS core.
- A post-processing code.

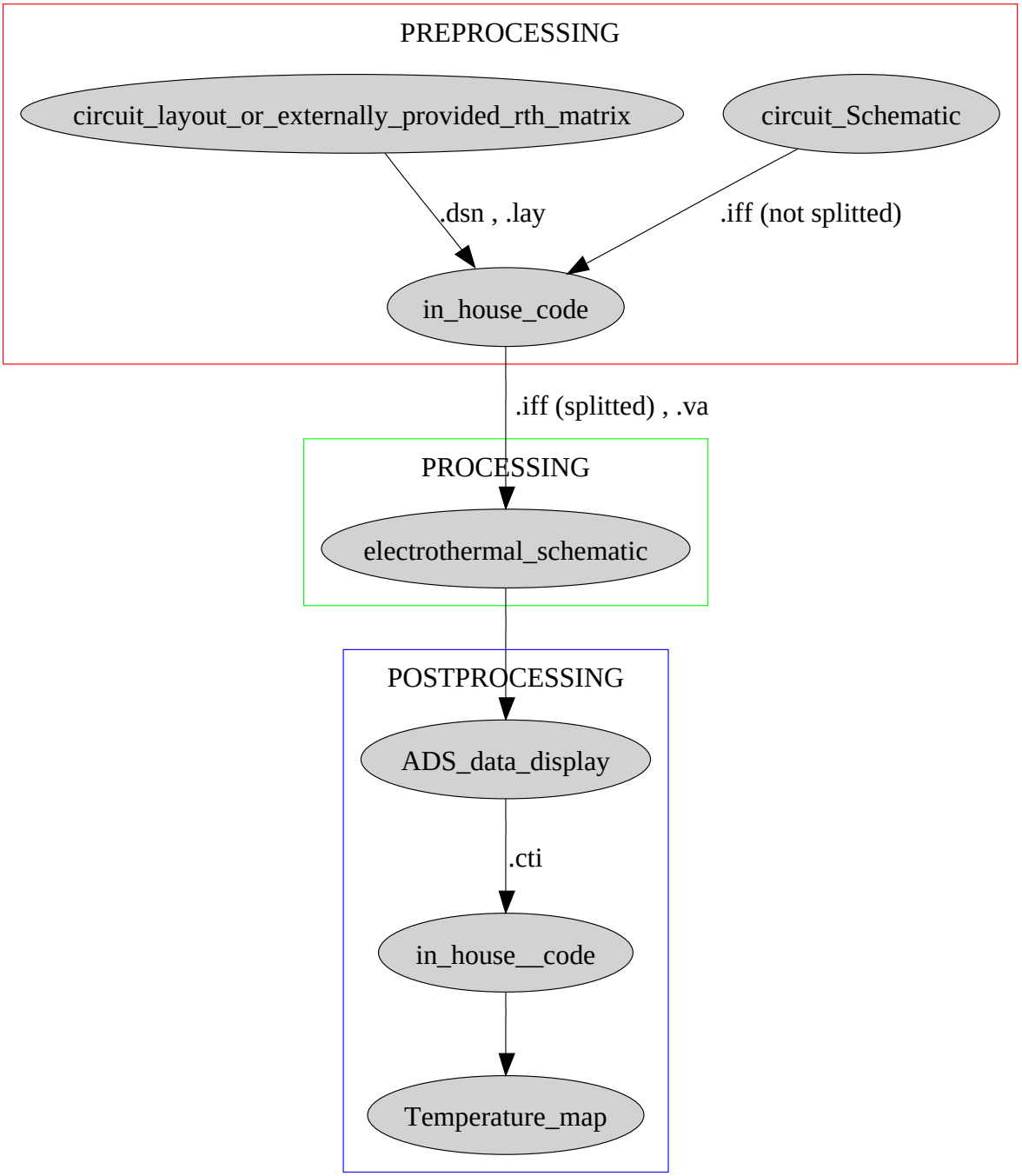


Figure 3.3: Simulator structure for external command-line approach.

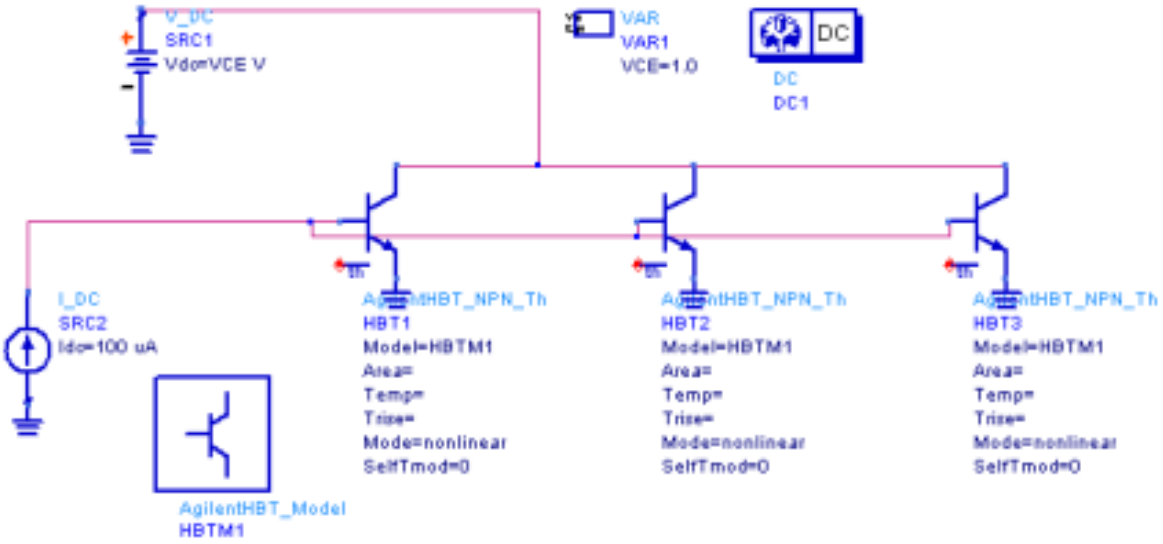
3.2.1. Preprocessing

The pre-processing routine uses a recently developed analytical approach devised to evaluate the temperature distribution due to a heat source shaped either as a rectangle or as a rectangular parallelepiped and located at an arbitrary depth within the domain. In particular, such a technique makes use of closed-form expressions attained via the combination of:

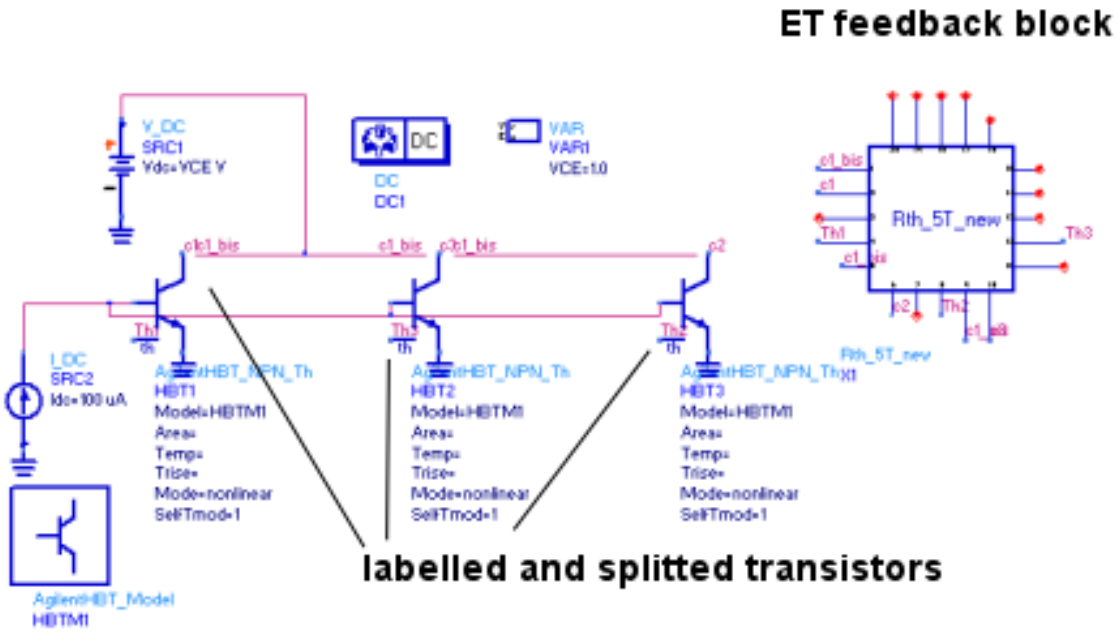
- The analytical approach [35] to calculate the temperature field in a homogeneous and indefinitely extended domain.
- The image method – largely used in electrostatics – to account for boundary conditions.

It is clear that two basic advantages are achieved, in respect to the strategies presented in the literature. First, the above approach is based on closed-form expressions, which noticeably lowers the CPU time with respect to the Fourier series and Green functions expansion techniques, especially for domains with large chip-to-source size ratio; second, an enormous reduction in time/memory requirements is obtained compared to numerical approaches, such as finite elements or finite differences, and the user is not involved any more in troublesome discretisation aspects. On the basis of the aforementioned approach, the pre-processing code automatically evaluates self and mutual thermal resistances from the layout file associated to the circuit. In particular, the code scans file for detecting the coordinates of the emitter window (i.e., the projections of the heat sources on the top surface); afterwards, the thermal resistance matrix is computed. Not that such a matrix can be also "directly" provided by the user. This is needed when analysing domains with complex geometries (as e.g., high-speed BJTs with trench SOI isolation in which the thermal behaviour is troublesome – or even impossible to be modelled analytically. In this case, one can preventively extract the thermal resistance values by means of numerical 3-D thermal tools (as e.g., Femlab) or experimental techniques. In detail, the preprocessing tasks are:

1. The standard – purely electrical – ADS schematic is exported to an intermediate file format (.iff) which is automatically modified in order to account for self-heating effects and thermal interactions. The modification is shown in Fig. 3.4
2. The thermal resistance matrix associated to the circuit is obtained by processing the layout file (.dsn) in pre-processing stage. In particular:
   (a) the layer where the emitter windows (which correspond to the heat sources "projections" on the top surface) is/are specified. This allows accessing to the heat sources coordinates along the x and y axes;



(a) Before preprocessing: an isothermal schematic.



(b) After preprocessing: an electro-thermal schematic with ETFB.

Figure 3.4: Translation process during processing for external command-line version of the simulator.

(b) the thermal properties of the semiconductor material and the "vertical" geometry (thickness and depth) of the heat sources (which actually coincide with the base-collector junctions) are assigned;

(c) the code evaluates the thermal resistance matrix through the approach proposed in [35], which allows to obtain effective closed-form analytical expressions. As an alternative, it is possible to specify "directly" the values of the thermal resistances.

3. Afterwards, the code creates a Verilog-A file (.va) that incorporates the thermal electro-thermal interaction. Such a file corresponds to a new block that is inserted into the original .iff file and automatically connected to the active devices. Starting from the collector current and the collector-to-emitter voltage of each transistor of the structure, the Verilog-A code evaluates the dissipated powers, and - through the thermal resistance matrix - the temperature increases above ambient that are, in turn, inputs of the active devices. Hence, we can refer to such a block as a new ETFB component, which is treated by ADS as an embedded element. Once this component is properly bounded to the "original" - isothermal - circuit, a new .iff file accounting for electro-thermal effects is realized. Note that the Verilog-A-based approach has been chosen due to the extremely high flexibility of this analog behavioural language towards the analytical description of components. Such a technique is being also employed for extending the models of ADS elements not intrinsically equipped with a "thermal node", such as diodes, resistances, MOSFETs.

Preprocessing restrictions

1. A first problem is related to the splitting of the collector node needed in order to determine the collector current and collector-emitter voltage in the electro-thermal analysis. After identifying all the transistors embedded in the original schematic (Fig. 3.5a), the code separates the collector terminal C_i of each device into a pair of nodes C_{ia}, C_{ib} (Fig. 3.5b) that are tied to two input terminals of the Verilog-A block and electrically shortened within the block itself to make the collector current I_{Ci} accessible (Fig. 3.5c).

2. Secondly, the Verilog-A solution implemented in the simulator works in a discrete way – depending on the number of transistors given in the schematic, a specific ET feedback component is automatically inserted. In particular:

   - 1 to 5 transistors - inserting and connecting block *Rth_5T_new* (max. 5 transistors, Fig. 3.6a).
   - 6 to 10 transistors - inserting and connecting block *Rth_10T_new* (max. 10 transistors, Fig. 3.6b).
   - 11 to 15 transistors - inserting and connecting block *Rth_15T_new* (max. 15 transistors, Fig. 3.6c).

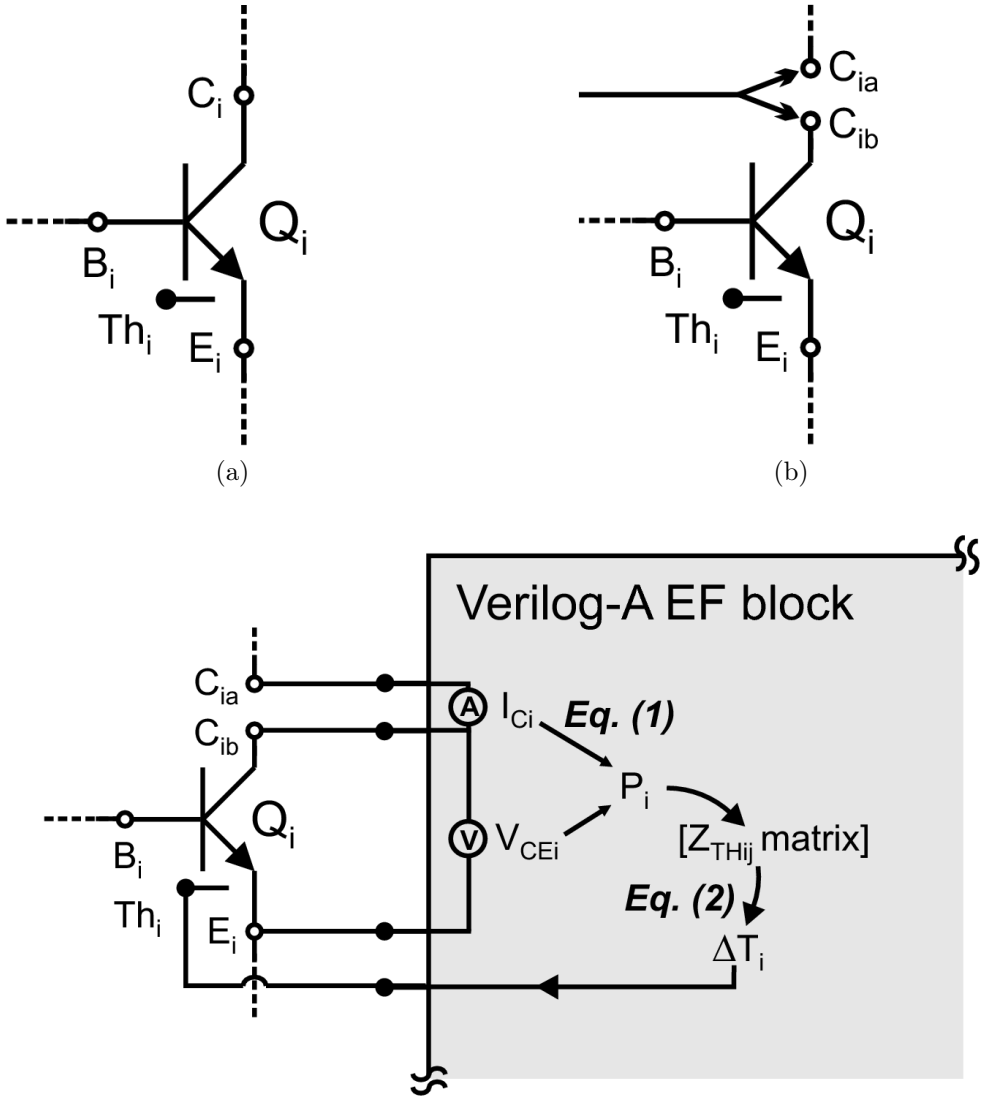


Figure 3.5: Collector "node-splitting issue" in the external command-line approach.

- 16 to 20 transistors - inserting and connecting block *Rth_20T_new* (max. 20 transistors, Fig. 3.6d).
- 20 to 25 transistors - inserting and connecting block *Rth_25T_new* (max. 25 transistors, Fig. 3.6e).

The library of available Verilog-A ET feedback blocks is shown in Fig. 3.6. However, the maximum amount of devices connected is restricted up to 25 transistors. It is related to the maximum amount of code accepted by Verilog-A compiler in a single component.

3.2.2. Processing

As mentioned above, the Verilog-A block including the thermal resistance matrix, is created and automatically connected to the active devices of the original (i.e., isothermal) schematic. The Verilog-A analog behavioural language has been chosen due to its high flexibility toward the analytical description of components. The resulting block is treated by ADS as an embedded element and is exploited to handle the electro-thermal feedback. In particular, it evaluates the powers dissipated by all the transistors included in the circuit starting from collector currents and collector-to-emitter voltages, and, through the thermal resistance matrix, the temperature increases above ambient, which, in turn, are applied to the thermal node of the active devices. A new "non-isothermal" circuit is therefore obtained and solved by the main core via the embedded ADS engine. The new file contains the electro-thermal network given by the proper connection of (i) the original circuit and (ii) the new ETFB component. Once this file is imported into ADS, the new electro-thermal – self-consistent – schematic can be displayed. At this stage, the electro-thermal simulation can be performed via the ADS engine. It is remarkable that the creation, the placement and the automatic connection of such a block is entirely handled by the code.

3.2.3. Post-processing

Lastly, the post-processing code is adopted for handling the simulation results (e.g., storing data in files) and for evaluating the temperature maps under assigned bias conditions over a chosen grid. The post processing stage is charged (partially) to ADS and (mainly) to another post-processing code. In the data-display window of ADS are directly plotted quantities as the voltages and currents, however it is not possible to plot the three-dimensional thermal maps. After exporting the simulation results to the citifile (.cti) format, it is possible to visualise temperature maps and contour plots at different DC bias point. For temperature maps it is required to specify the chip area by means of an enclosing rectangle on new layer in the layout file (.dsn).

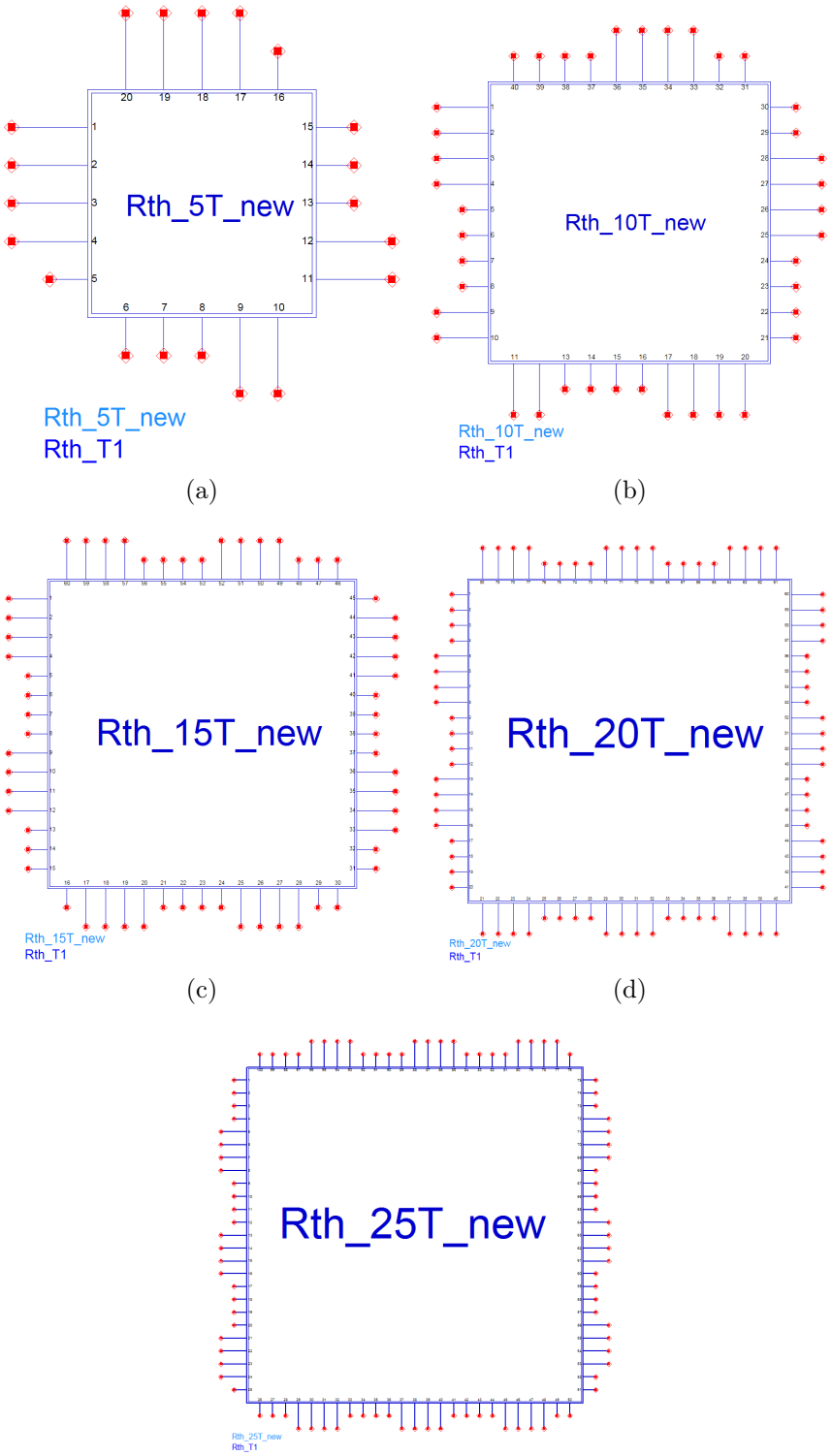


Figure 3.6: Library of available Verilog-A ETFBs for external-command line approach: 5, 10, 15, 20 and 25 transistors in each block.

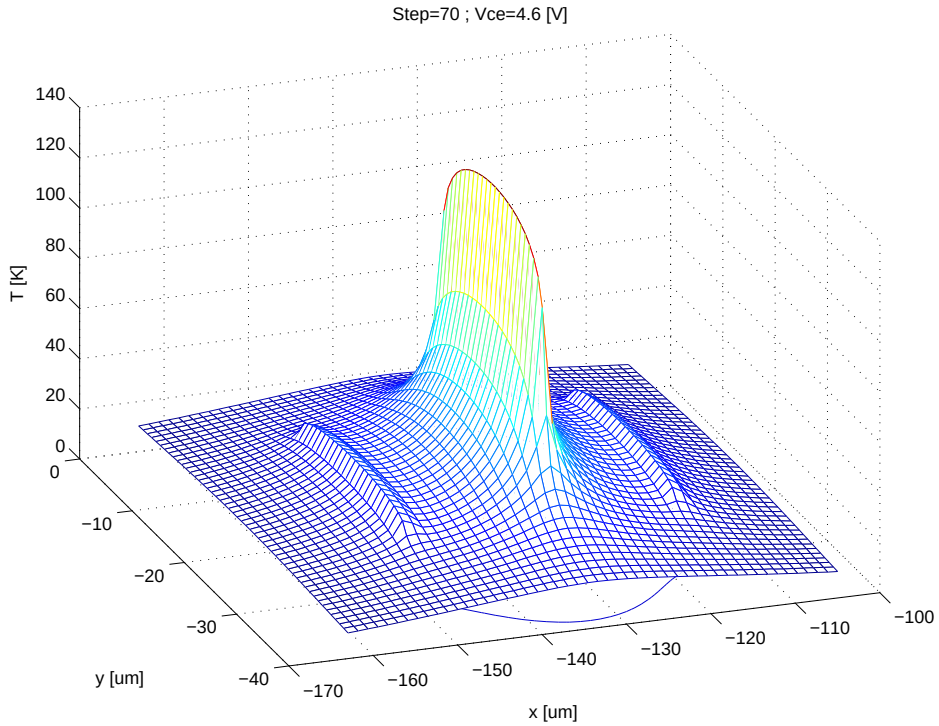


Figure 3.7: Postprocessor output as a result of ET simulation for 3-emitter finger HBT device.

3.2.4. Approach optimisation

1. The pre-processor code *lay2ver.m* whose task is to evaluate the thermal resistance matrix through the SEHS[4] technique, has been corrected, improved and optimised.
   The pre-processing code does not repeat the evaluation of symmetrical mutual thermal coupling resistances in the Verilog-A file, so that memory is saved during the compilation of the Verilog-A code for the ET feedback block. This does not help in terms of increasing the number of pins available.

2. The pre-processor code *iff2iff.m* has been improved and optimised. The one-file code has been subdivided into several small functions, thus making it more effective.

3. The post-processor code *data2thmap.m* has been improved and optimised.
   The calculation speed has increased. E.g. for 1000 by 1000 mesh - (one million points) the calculation cost is strongly reduced. Another advantage is that the code is able to plot various thermal maps for various steady state solutions without restarting the postprocessor and taking into account simulation output again. An example of the thermal map is shown in Fig. 3.7.

[4]Superposition of Elementary Heat Sources

3.3. Approach based on internal ADS code

It is possible to include electro-thermal steady-state simulation inside ADS, without resorting to other programs. Conversely to SPICE, ADS has built-in programming language called AEL[5], which is a general purpose programming language, in structure similar to well-known C language. AEL is used to configure, customise and extend the capabilities of the design environment. Like C, AEL has an extensive set of built-in function libraries, including functions for:

- File input/output (operation on ASCII text files of various types).
- Math (basic math functions, excluding a calculation of integrals).
- String manipulation.
- List handling (e.g. creating list of components in both layout and schematic).
- Database query (e.g. traversing the project/design or ADS environment).

In general, one can use AEL for tasks like:

- Organising libraries and palettes of components. E.g. a user can add developed palettes of components either in schematic and layout.
- Defining the interface to new user-defined components For example a developed function can be added in a schematic/layout as a menu.
- Creating new components with layout artwork The phrase "artwork" means the layout representation of the simulation model in schematic. In ADS, thanks to AEL, the layout representation might be parameterised, that is generally speaking dependent on the variable/parameter in the schematic. ADS provides some "fixed artworks", which are used for layout items that do not change size or shape based on parameter settings. For example, an SOT23 package outline is the same for any device with that package, regardless of the device operating parameters.
- Defining custom layout artwork functions The idea is shown above, however these types of artwork functions are developed by a user.
- Defining the interface to discrete-valued simulation components. It is possible using the Data Access Component (DAC). DAC is used in a schematic in order to read the various types of industry-standard text files, setting this way the parameter values for certain components of the model, which reference individual values from the data file.

[5] Application Extension Language

- Creating custom utility functions, such as parts list generators and bill of materials.
- **Automating** routine tasks, such as repetitive command sequences, batch analysis, or optimisations.

Hence, at the beginning AEL has been chosen to substitute the preprocessor tasks, which are:

1. Extraction of the thermal properties of circuit from layout.
2. Translation of the purely electrical schematic into electro-thermal one.

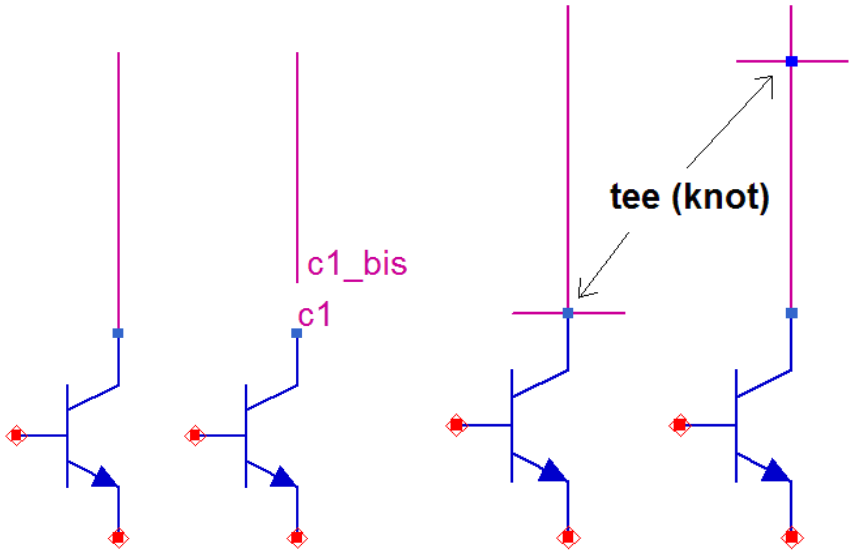
However, with AEL language it is not possible to perform a node splitting, used with external command-line approach. In particular:

1. In the external command-line approach, as shown in Fig. 3.8, in order to get the dissipated power, the output node had to be split. However, using internal AEL code it is impossible to split the output node wire for all cases, that may occur in the circuit since tee/knot coordinates are necessary to perform the splitting operation. It is not possible to obtain, using public AEL functions. Probably it is possible to obtain a tee/knot coordinates using advanced AEL functions, restricted for ADS developers.

2. Further, a shortening wire operation is very cumbersome to perform in AEL. This task would be possible to accomplish, saving the wire positions in a memory, deleting the wire and attaching the "shortened" wire in a schematic.
   In order to overcome above limitations, a new solution has been developed, based on improvement of ADS built-in models.

3.3.1. Improved ADS compact models

In an electro-thermal simulation, for the calculation of temperature increase above ambient, one needs dissipated power and thermal resistance.
In the external command-line approach described in section 3.2 on page 40, the dissipated power was externally calculated by definition, outside of a compact model. In this way, the dissipated power is the sum of dissipated power in the input and output. Such approach is suitable only for steady-state (DC) electro-thermal simulation [50]. It requires also an additional splitting and creates complicated algorithms to perform that task.
However , ADS introduces new compact models, supplied with additional thermal node, where the dissipated power is already correctly calculated for the case of transient, steady-state and ac simulation. The thermal circuit of the ADS model is shown in Fig.



(a) Node-splitting issue in external command-line approach, in order to get the output current

(b) The "tee" description, used in ADS terminology. Tee is a "root" node, that has several wire branches.

Figure 3.8: Node-splitting problems appear also when using ADS internal language AEL.

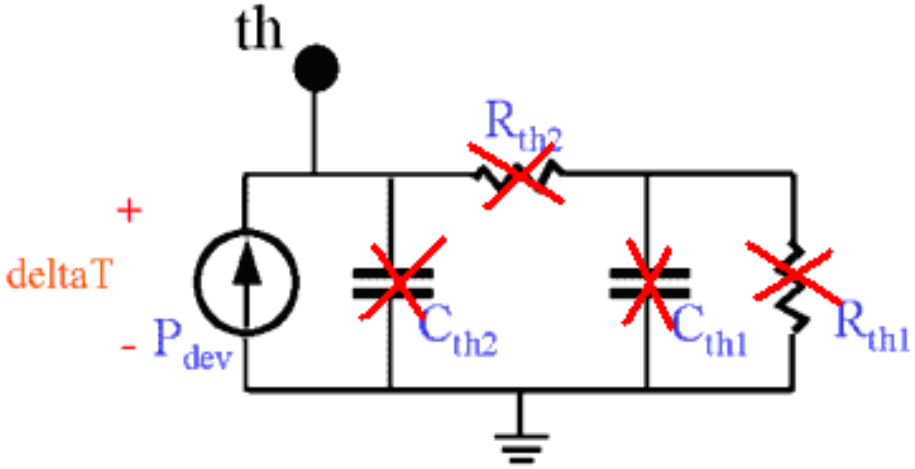


Figure 3.9: Thermal circuit of the model. All the internal capacitances and resistances must be deactivated in order to obtain the dissipated power, which is already calculated within the device compact model.

3.9. The dissipated power, is calculated by the current source P_{dev}. In order to obtain the dissipated power P_{dev} from the ADS compact model, following procedure must be performed:

1. All the internal thermal resistances R_{th} and capacitances C_{th} inside the model have to be deactivated, setting the value to infinite or zero. Some of the models can have one or several thermal resistances R_{th}, as shown in Fig. 3.9 . So what is to be done, is setting R_{th1} value to infinite (e.g. 1e100 etc.) and R_{th2} value to zero. Thermal capacitances C_{th} should be set to zero. It is done setting the model parameters, as shown in Fig. 3.10.

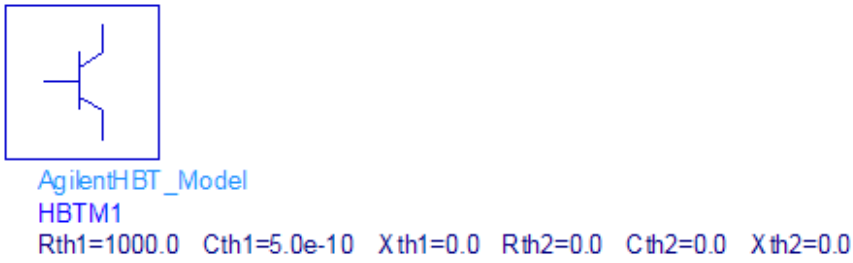


Figure 3.10: Model used in ADS and its thermal network parameters: thermal resistance Rth, thermal capacitance Cth, temperature exponent for thermal resistance Xth.

2. The CCCS (Current Controlled Current Source) is attached to the thermal node. An output from CCCS is a current, which is an internally calculated dissipated power P_{dev}. It is visible in Fig. 3.11.

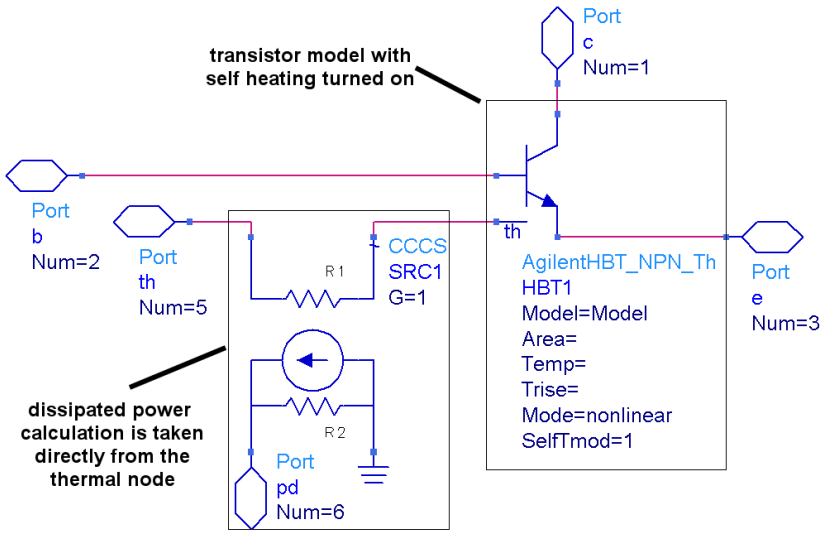


Figure 3.11: Improved model inside, together with the dissipated power calculation block.

3. A model obtains new schematic representation, receiving extra node, so called dissipated power node (pd) attached to the output from the dissipated power calculating block (CCCS). The footprint is shown in Fig. 3.12. Beside the new

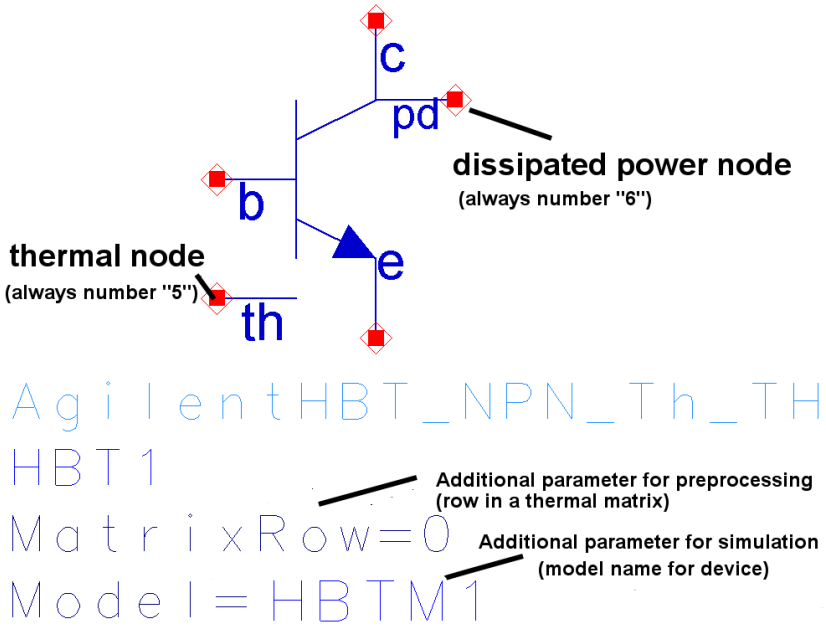


Figure 3.12: Improved model footprint, with additional parameters

schematic representation, all improved compact models receive have pin numeration, where:

- thermal node pin (th) obtains pin number "5".
- dissipated power node pin (pd) obtains internal pin number "6".

Although not visible for a user, the pins numbers are important for the AEL preprocessing functions, which can automatically recognise them.

4. As shown in Fig. 3.12, a model receives new additional parameters, which are:

   - *MatrixRow* – a number, which indicates row number in a thermal resistance/impedance matrix. This parameter is necessary for layout preprocessing functions in AEL.
   - *Model* – a string, which indicates the model name attached to the device. E.g. for the one shown in Fig. 3.10 the model name is *HBTM1*.

   Concluding, enhancing ADS models lets to:

   - Avoid node-splitting operations. Compact models are directly connected to a ETFB, without performing the splitting operation.

➤ Calculate the dissipated power correctly. Currently, the dissipated power is calculated correctly for AC/DC/transient simulations and the models are connected directly to the thermal network.

The improved devices are available in so called Design-Kit library. Design Kits are libraries design for all ADS platforms.
The name of the design kit is set to "ELT-Design-kit", where "ELT" is the short form for "electro-thermal". The models are available in the tab "ELT-models", as shown in Fig. 3.13: In particular after selecting the tab "ELT-Devices" from the "ELT Design

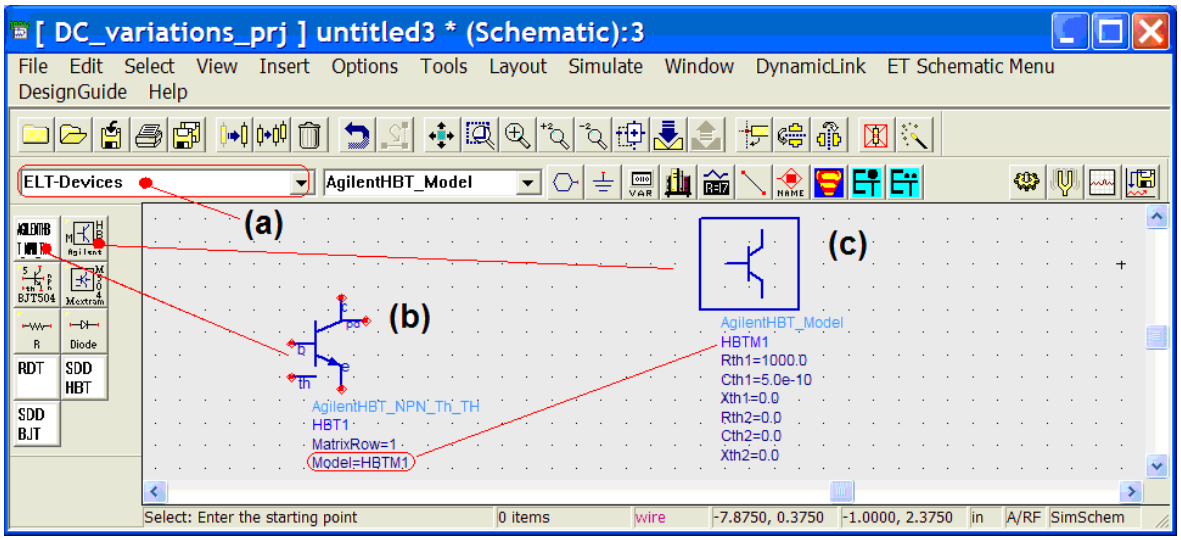


Figure 3.13: "ELT Design-kit" idea: (1) tab with models, so called "ELT-devices"; (b) "AgilentHBT_NPN_Th_TH" device, available after clicking at the button indicated by red point; (c) HBTM1 model, attached to the device, with default thermal parameters.

Kit", it is possible to select a desired device with its model, and put it into circuit schematic.
Concluding, the "ELT design-kit" enables the automation of electro-thermal preprocessing, i.e. translation of original isothermal schematic to electro-thermal one directly in ADS environment. Devices, models and ETFBs are recognised on any ADS platform, as well for internal as external preprocessing routines.

3.3.2. Thermal network approaches

The 2nd and 3rd strategies applied for the development of the electro-thermal simulator code (based fully on internal or external code), imply another type of thermal networks than used with the external command line approach.
Here the input of a ETFB[6] is a dissipated power (current signal), and the output is the

[6]Electro-Thermal Feedback Block

temperature (voltage signal). However, previously in the approach based on external command-line, a dissipated power was calculated externally, based on the definition of power P= V · I. In other approaches, the dissipated power is calculated internally, inside models. As a result, a thermal network is connected directly to the device via dissipated power node and thermal node.
Several strategies have been implied for thermal networks. In particular:

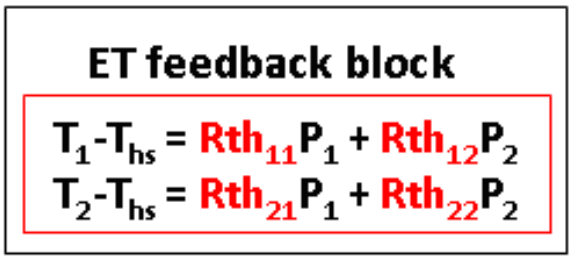
- Verilog-A thermal network.
- SDD thermal network.
- Spice-like thermal network.

All thermal networks approaches are using well known relation between electrical and thermal phenomena. The dissipated power (input) is a current, and the temperature increase above ambient (output) is a voltage. In Fig. 3.14 is demonstrated, how from thermal networks equations (Fig. 3.14a), is generated the circuit idea (Fig. 3.14b), that realizes exactly the equations in (Fig. 3.14a. The circuit representation of a thermal network is called ETFB[6] (Fig. 3.14c). Every type of ETFB has the same circuit representation, that is the schematic footprint, and calculates the temperature increase above ambient using the same approach. The difference between various types of ETFBs, like SDD, Spice-like or Verilog-A is actually in their specific circuit implementation. However for a user is not important, which kind of thermal network, that is ETFB, is used. The simulation results are always the same.
As improved devices, the electro-thermal feedback blocks are available in so called "design-kit" library. "Design kits" are distribuitable libraries between all ADS platforms. The name of the design kit is "ELT-Design-kit", where "ELT" is the acronym form for "electrothermal". The ETFBs for both 2nd and 3rd approaches are available in the tabs:

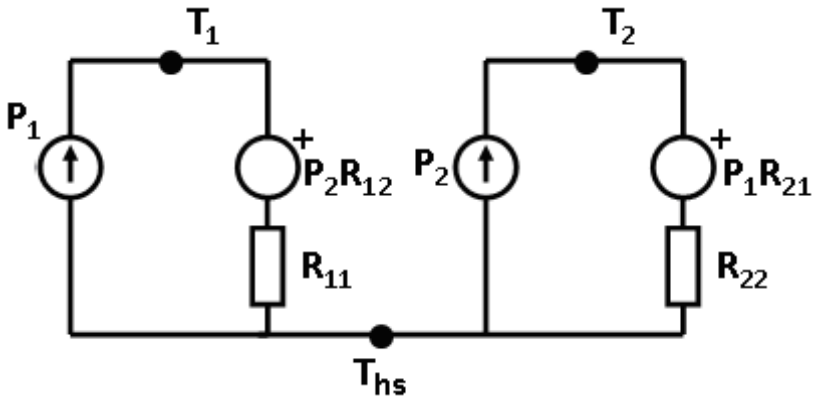
ELT-SDD-blocks In this tab are available all ETFBs, built by means of SDD implementation. For details refer to section 3.3.3.

ELT-Spice-blocks In this tab are available all ETFBs, built by means of Equivalent networks implementation. For details refer to section 3.3.4.

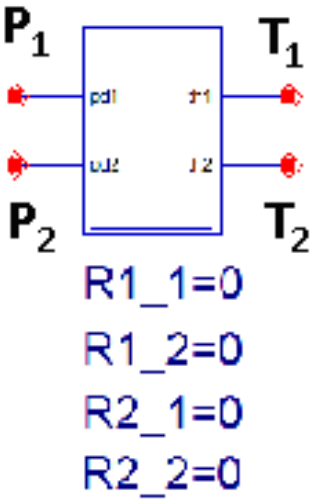
ELT-Verilog-A-NEW-blocks In this tab are available all ETFBs, built by means of Verilog-A implementation. These Verilog-A ETFBs circuits can cooperate only with the 2nd or 3rd approaches, based on internal/external code. For details refer to section 3.3.5.
External command-line approach is supported by ETFBs in the tab "ELT-Verilog -A-OLD-blocks" in the ELT-Design-kit. For details refer to section 3.2.



(a)



(b)



(c)

Figure 3.14: ETFB idea: Circuit in (b) realizes the equations in (a). Circuit schematic representation in (c).

3.3.3. SDD thermal network

A Symbolically-Defined Device (SDD) component enables to create equation based, user-defined, nonlinear devices. An SDD is a multi-port component, which is defined by specifying algebraic relationships that relate the port voltages, currents, and their derivatives, plus currents from certain other devices.
As shown in Fig. 3.15, the SDD thermal networks are available in the in the tab "ELT-SDD-blocks" of the "ELT design-kit" library. The general structure for SDD

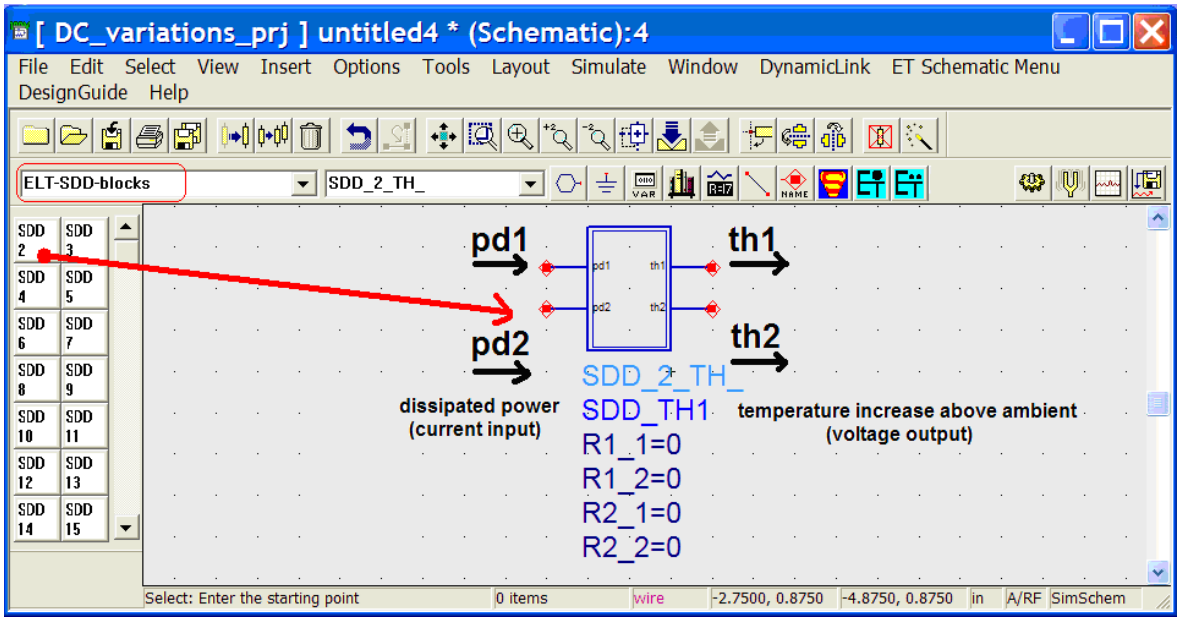


Figure 3.15: After selecting "ELT-SDD-blocks" tab, the palette of available SDD electrothermal feedback blocks is visible on the left side of the ADS schematic window.

thermal network is as follows:

1. Get the dissipated current on the "pd" nodes (pins), which are shown in Figs. 3.15 & 3.16(a).

2. Change the current into voltage by means of CCVS (Fig. 3.16(b)) to distribute the voltage to all the SDD "inner" components, without loosing the information about the dissipated power, since the voltage signal is the same as well at the output of the CCVS, as on each node of the SDD "inner" component (Fig. 3.16(c)).

3. In the SDD "inner" component (Figs. 3.16(c) & 3.17), withdraws the voltages on the input ports, and on the output port change them into current. Further each current is multiplied by a certain thermal resistance (Fig. 3.17(c)). All products contribute to a final equation, as shown in Fig. 3.17 (b).

4. Finally a temperature increase above ambient is calculated by changing the final sum (Fig. 3.17(b))into a voltage by means of CCVS and output the temperature at the "th" nodes (Fig. 3.17 (c)).

Concluding, the SDD ETFBs consist of three "schematic levels":

1. 1st level, shown in Fig. 3.15, is used by a user in a circuit schematic.
2. 2nd level, shown in Fig. 3.16.

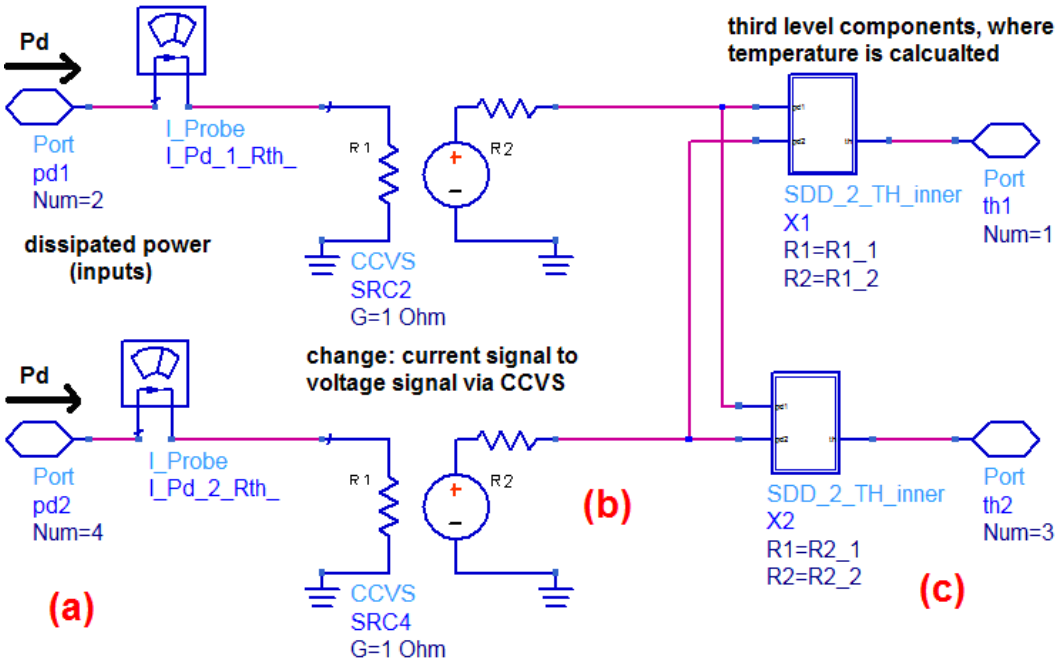


Figure 3.16: 2nd level SDD thermal network. The input in (a) is a dissipated power (pd ports), translated into voltage by CCVS sources (b). Voltage is passed as input into "third-level SDD blocks" (c).

3. 3rd level, shown in Fig. 3.17.

In the 2nd level of the SDD thermal block, the current probes called "I_Pd_1_Rth" and "I_Pd_2_Rth" are inserted to display the dissipated power. The current probes are present to check whether the device dissipates power and the temperature increase above ambient is correctly calculated.

Example of attaching SDD thermal network

In the example presented in Fig. 3.18, two *AgilentHBT_NPN_Th_TH* transistors are connected in parallel in the case of constant base current. The base current is

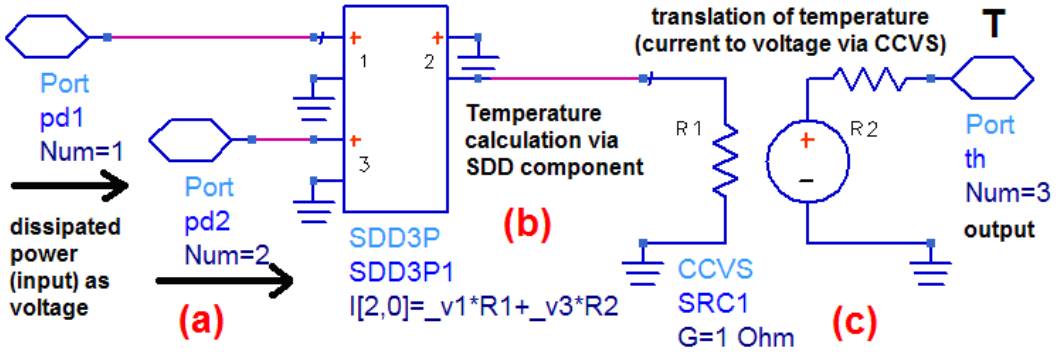


Figure 3.17: 3rd level SDD blocks which execute calculation of temperature increase above ambient. Output current (temperature) is translated into voltage signal by CCVS sources.

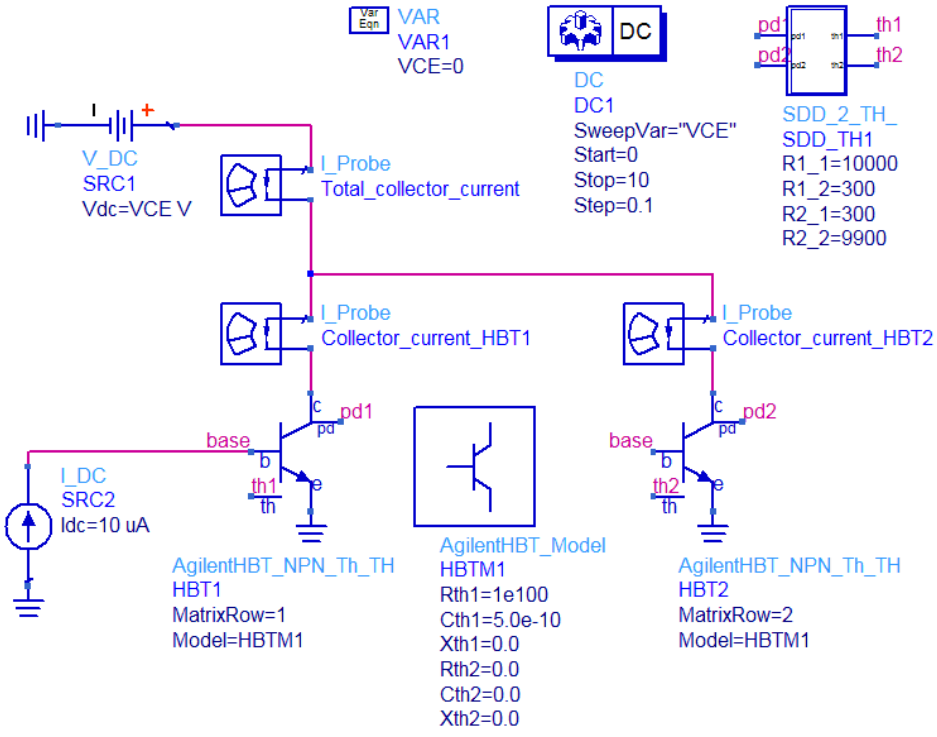


Figure 3.18: Simulation example with SDD thermal network for two transistors connected in parallel.

set to 10μA, and the collector voltage is changed from 0 to 10 Volts with a 0.1 Volt step. The transistors have common model, namely *HBTM1*, which internal thermal resistance *Rth1* is set to the value of 1e100 Ω, in order to be recognised as infinite one by a simulator. As a consequence, a current flow through *Rth1* is not considered. It enables the connection of external thermal resistances via *SDD_TH1* component. In particular *R1_1* & *R2_2* resistances (diagonal) are self heating ones and have very large value (transistors might be realized in silicon on glass technology), the mutual thermal resistances *R1_2* and *R2_1* are set equally for the value of 300 K/W. The slight discrepancy between the self-heating resistances of 100 K/W is a reason of the bifurcation occurrence, shown in Fig. 3.19a. As it is visible in Fig. 3.19, the temperature increase above ambient is calculated correctly, comparing the results from Fig. 3.19b and in Fig. 3.19c, where the temperature is calculated by following equations:

$$\begin{aligned} T_1 - T_{hs} &= P_1 \cdot R1\_1 + P_2 \cdot R1\_2 \\ T_1 - T_{hs} &= P_1 \cdot R2\_1 + P_2 \cdot R2\_2 \end{aligned} \tag{3.2}$$

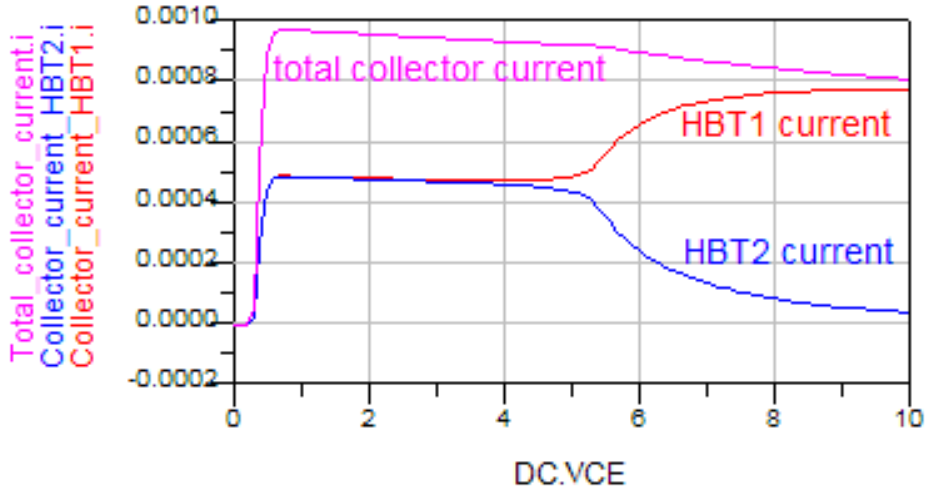
3.3.4. "SPICE-like" thermal network

In "SDD-thermal network", the blocks were using specific components available in ADS environment.
In the approach described in this section, the components used to build the ETFBs, are commonly used in all circuit simulators. In particular these are:

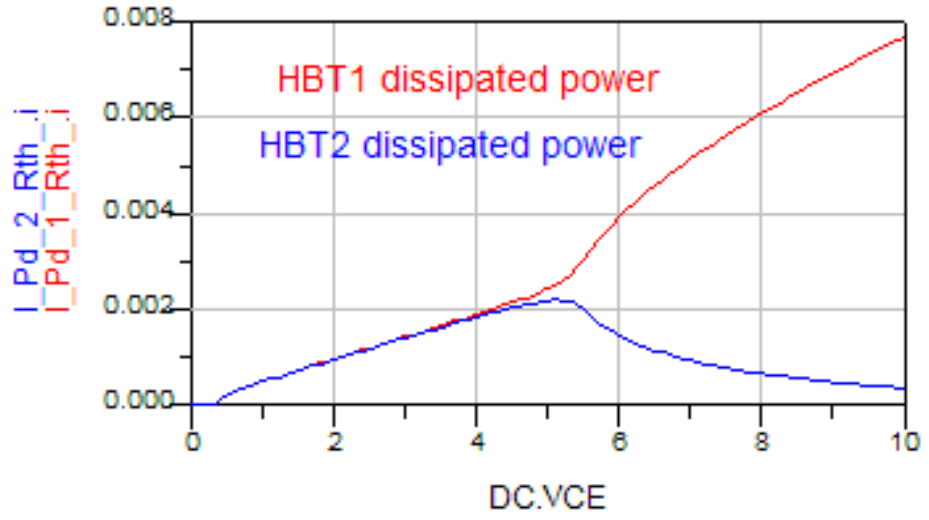
- Resistors.
- Current Controlled Voltage Sources (CCVS).
- Current Controlled Current Sources (CCCS).
- Voltage Controlled Current Sources (VCCS).
- Voltage Controlled Voltage Sources (VCVS).

As shown in Fig. 3.15, the Spice-like thermal networks are available in the "ELT design-kit", in the tab "ELT-Spice-blocks". The name of the components begins with "Net".
The general structure for SPICE-like thermal network is as follows:

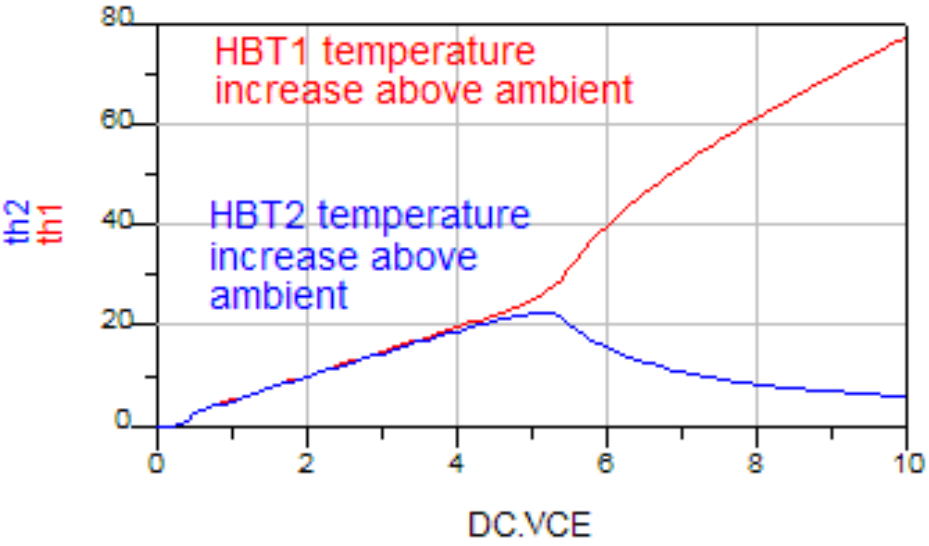
1. Get the dissipated current on the "pd" nodes (pins) shown in Figs. 3.20 & 3.21.
2. Change the current into voltage by means of CCVS (Fig. 3.21). In this way distribute the voltage to all the Splice-like "inner" components, without loosing the information about the dissipated power.



(a) For the VCE larger than 4,5 V collapse of the total current gain occurs.



(b) Further, the "hotter" transistor finger ...



(c) ... starts to bear whole current.

Figure 3.19: Result of the SDD example simulation.

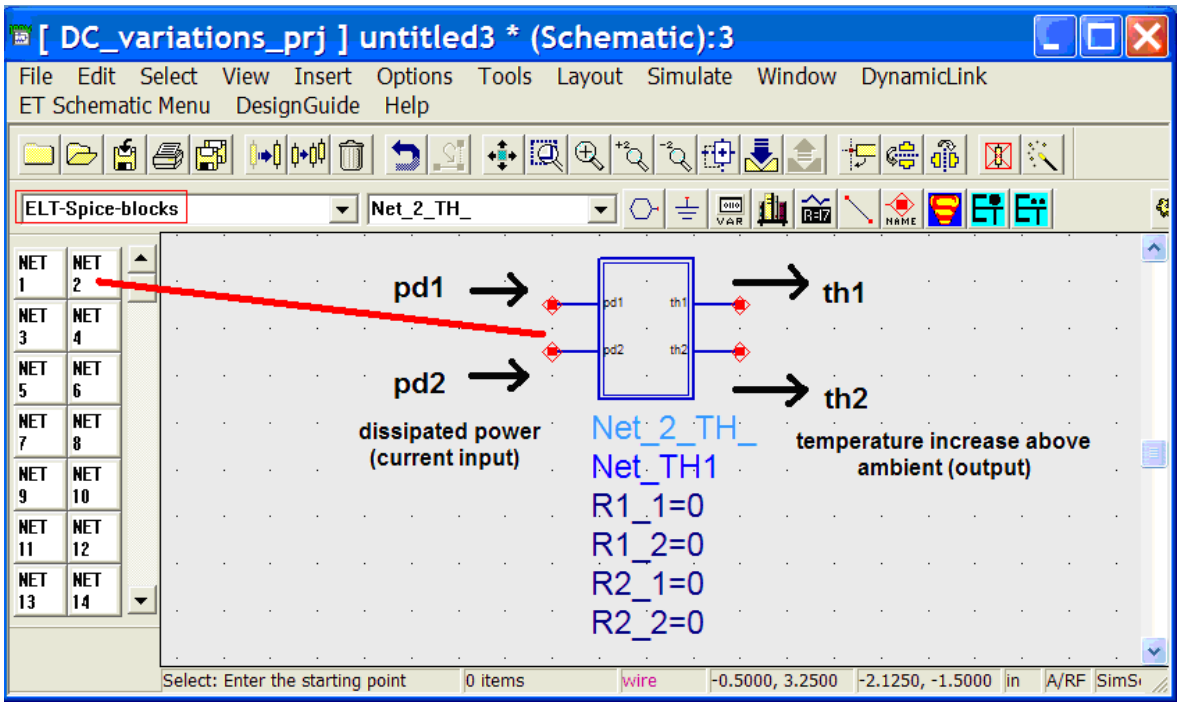


Figure 3.20: Selecting "ELT-Spice-blocks" tab, all the palette of available SPICE-like electrothermal feedback blocks is visible on the left side of the ADS schematic window.

3. In the Spice-like "inner" component (Figs. 3.21 & 3.22) the voltages on the input ports are withdrawn, and on the output port changed to current signal. Further each current is multiplied by certain thermal resistances and sum all these products.

4. Finally a temperature increase above ambient is calculated and changed the into voltage signal by means of CCVS. An output of a temperature increase above ambient is on the nodes "th", as shown in Fig. 3.22.

Example of attaching Spice-like thermal network:

As in the previous example described in section 3.3.3, instead of SDD thermal block the Spice-like thermal network has been attached. Results are identical as in the SDD case (Fig. 3.19) and shown in Fig. 3.24:

3.3.5. Verilog-A thermal network

With respect to the external command-line approach, the Verilog-A ETFBs used for 2nd and 3rd version of the electrothermal simulator, are changed. Similar to the SDD & Spice-like thermal networks approaches, the input is a dissipated power (current) and the output is a temperature increase above ambient (voltage). The thermal resistances

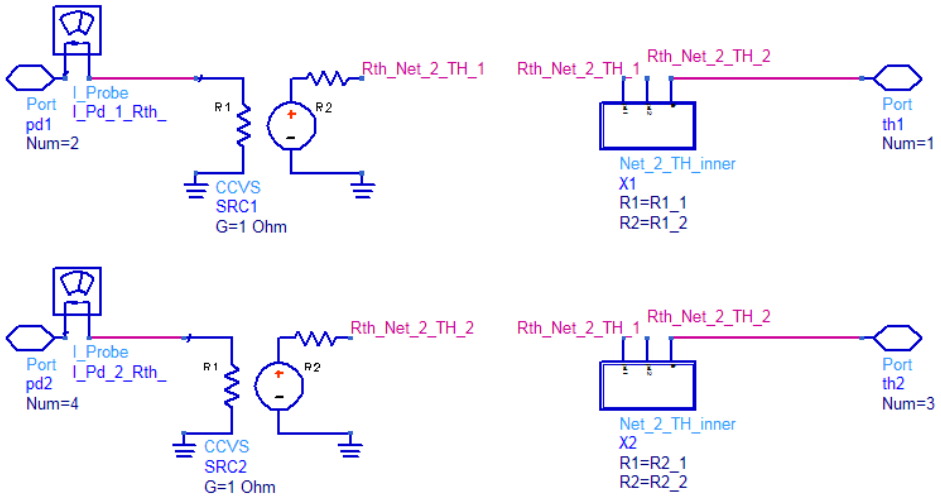


Figure 3.21: SPICE-like thermal network second level. Firstly, as in the SDD case, the current signal (pd) is translated into voltage by CCVS sources and passed as an input into "Spice-like 3rd level" (X1 and X2 instances), together with externally set thermal self/mutual thermal resistances R1_1, R1_2, R2_1, R2_2.

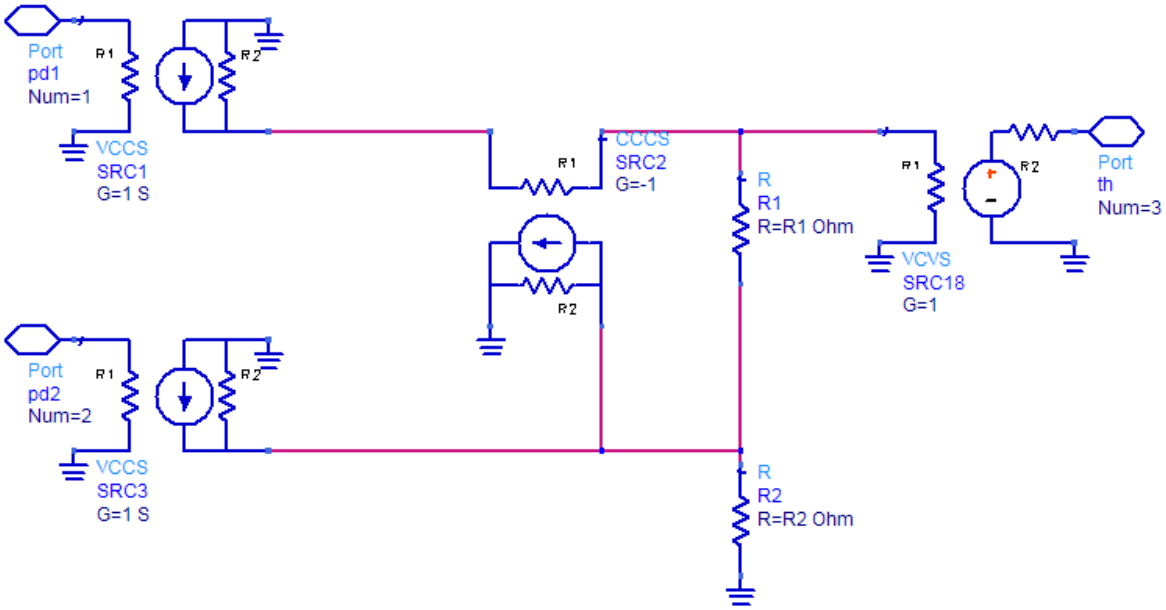


Figure 3.22: "Spice-like thermal network third level" – inner components. The voltage signal is translated into current by VCCS. The output equation realized by this block is Th = Pd1 · R1 + Pd2 · R2. The values of R1 & R2 are passed from the 2nd level design.

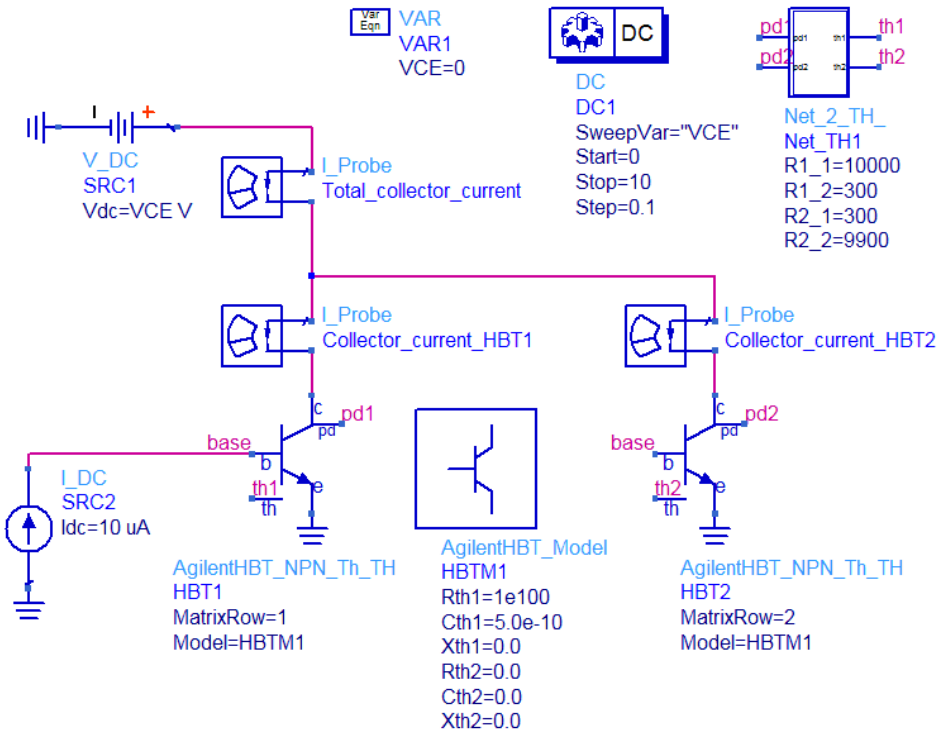
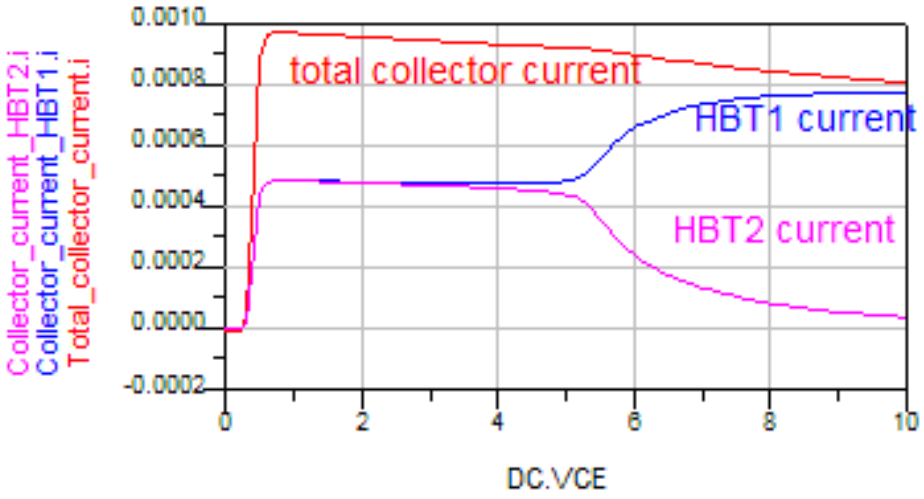


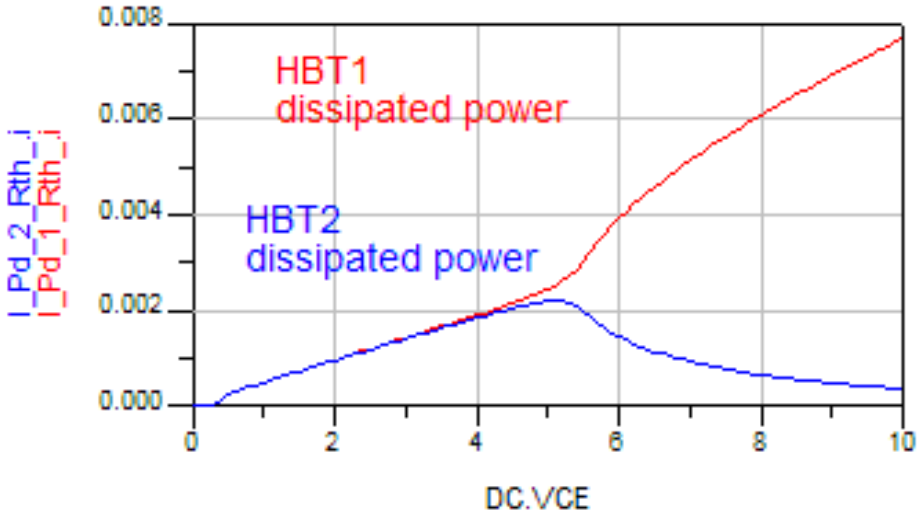
Figure 3.23: Simulation example with SPICE-like thermal network for two transistors connected in parallel.

are passed as the parameters in the circuit schematic. It means that a Verilog-A ETFB can be supplied directly in the design-kit as in a compiled form, without necessity of source code distribution from the Design-kit provider. An example of Verilog-A ETFB is shown in the Listing 3.1.

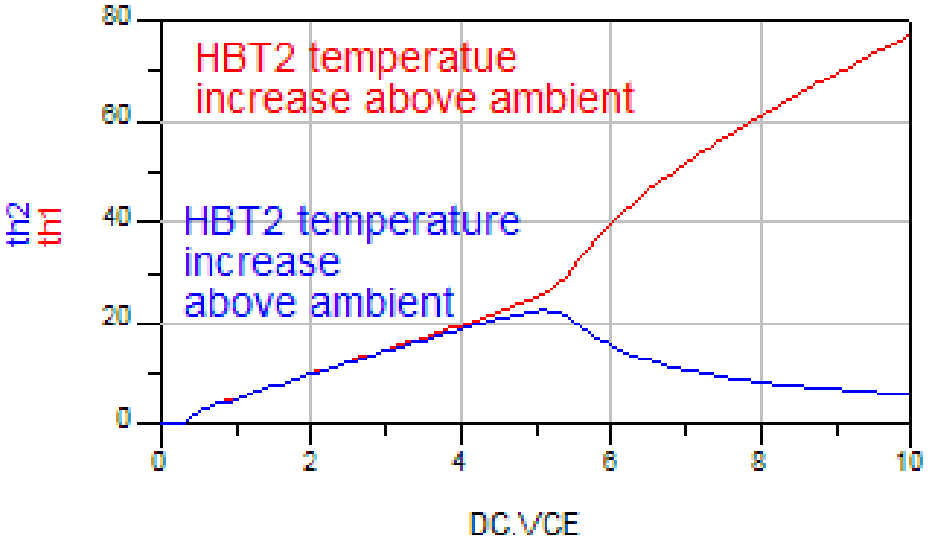
```
1  `include "disciplines.vams"
2  `include "constants.vams"
3
4  module VerilogNew_2_TH_
5    (
6    th1,pd1,
7    th2,pd2
8    );
9
10   inout pd1,th1 pd2,th2;
11   electrical pd1,th1,pd2,th2;
12
13   parameter real R1_1=0, R1_2=0;//initial parameters
14   parameter real R2_1=0, R2_2=0;//set in schematic
15
16   analog begin
```



(a) For the VCE larger than 4,5 V collapse of the total current gain occurs.



(b) Further, the "hotter" transistor finger ...



(c) ... starts to bear whole current.

Figure 3.24: Result of the SPICE-like thermal network example simulation.

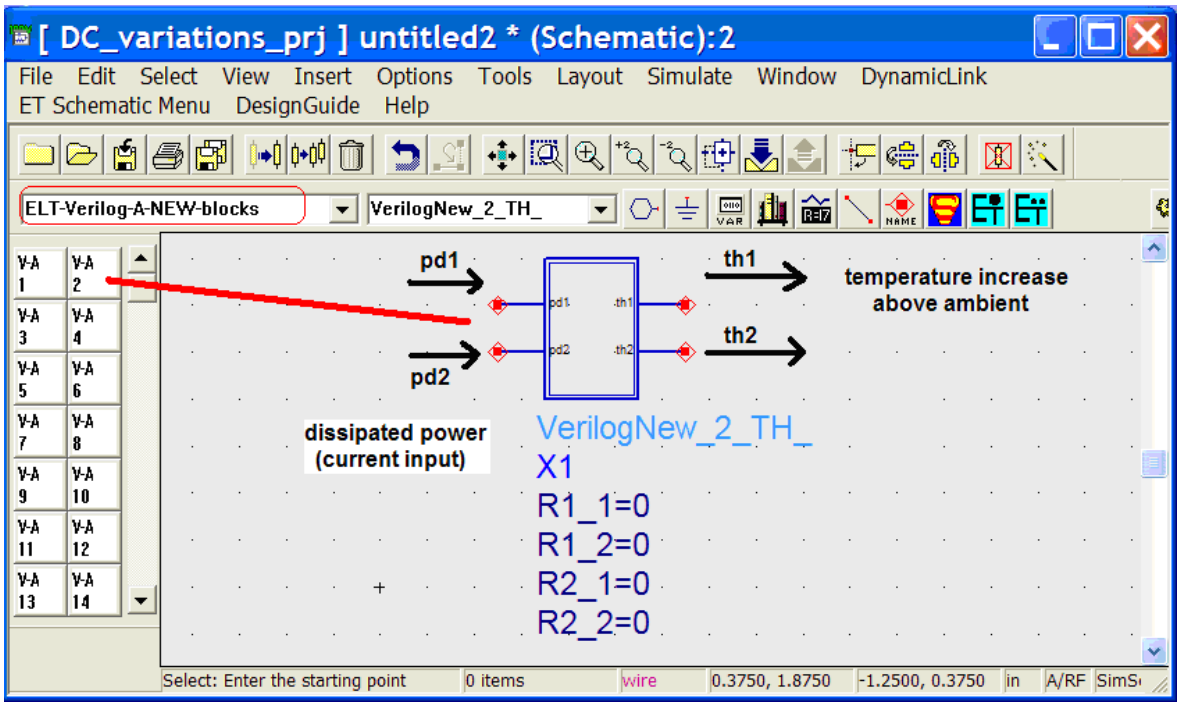


Figure 3.25: Verilog-A ETFB used for 2nd and 3rd simulator version. The circuit is completely described by a text-file, as shown in the Listing 3.1.

```
17        V(th1) <+ I(pd1)*R1_1 + I(pd2)*R1_2;
18        V(th2) <+ I(pd1)*R1_2 + I(pd2)*R2_2;
19     end
20  endmodule
```

Listing 3.1: Verilog-A ETFB code. The module is presented in Fig. 3.25

Example of attaching Verilog-A thermal network:

Results of the simulation, shown in Fig. 3.27, are the same as in Fig. 3.19 and in Fig. 3.24 for other types of ETFBs described in previous sections.
A visible difference in the approach between Verilog-A, and other solutions presented previously is that it is not possible to insert the current probe inside a Verilog-A ETFB, since it is not possible move inside the hierarchy of the circuit. The current probes are attached to the dissipated power nodes, as shown in Fig. 3.26.

3.3.6. Design-kit libraries

In order to distribute the models and electrothermal feedback blocks, the library of these components has been created. In ADS such distributed libraries are called "Design

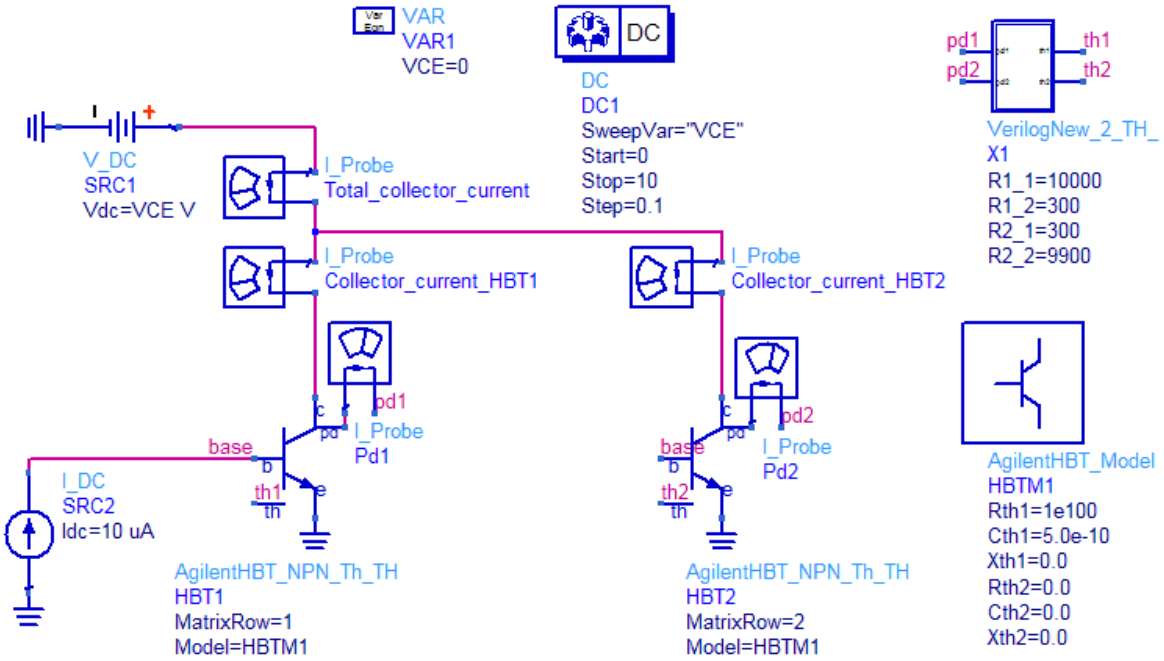
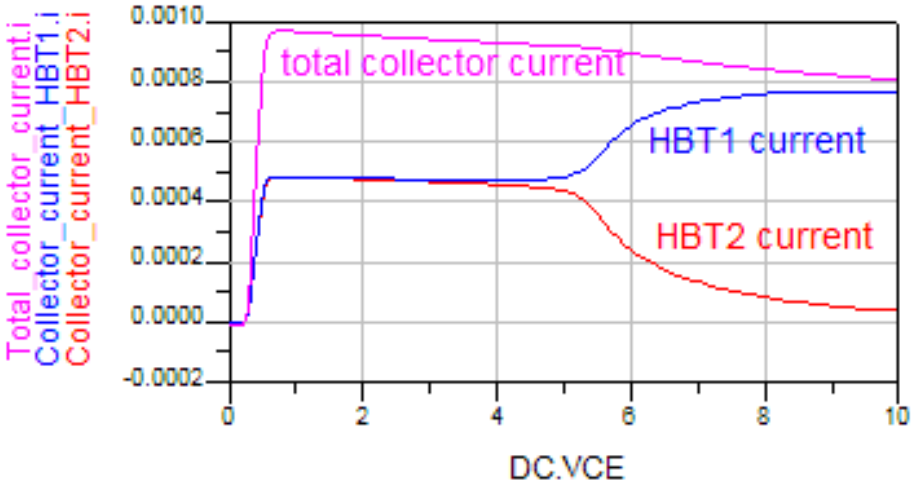


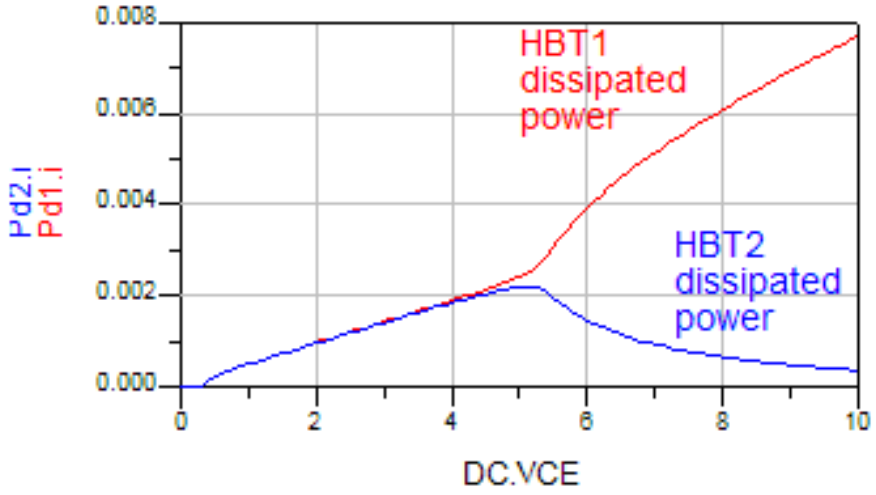
Figure 3.26: Simulation example with Verilog-A ETFB for two transistors connected in parallel.

Kits". In our case the Design Kit is called "ELT design kit". For brief concept about "Design-Kits" please refer to Fig. 3.28 and the Listing 3.2. For the details about Design Kit installation, please refer to the ADS documentation [52].

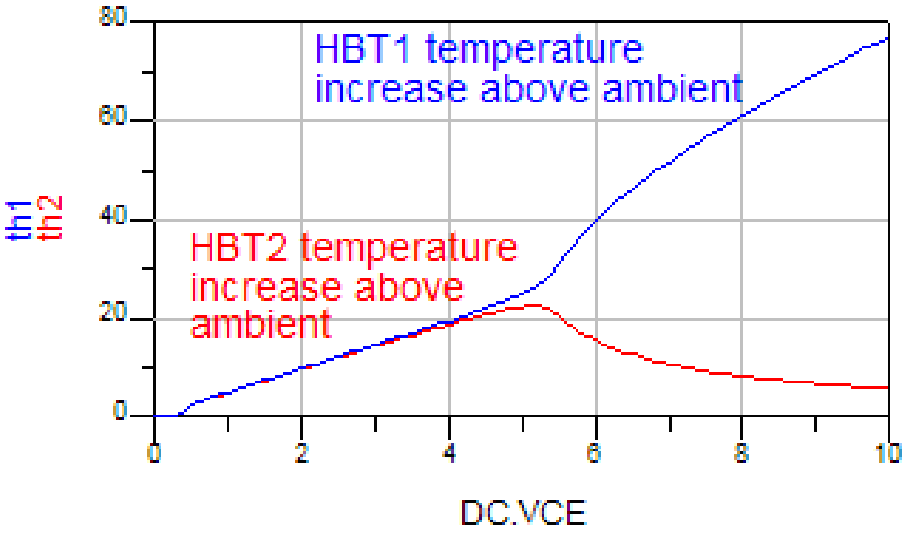
```
1  ======================================
2    Design Kits Levels
3  ======================================
4  Level: SITE
5    File location:
6    \$HPEESOF_DIR/custom/design_kit/ads.lib
7    Default: disabled
8
9  Level: USER
10   File location:
11   \$HOME/hpeesof/design_kit/ads.lib/
12   Default: enabled
13
14 Level: STARTUP
15   File location:
16   \$HOME/ads.lib
17   Default: enabled
18
```



(a) For the VCE larger than 4,5 V collapse of the total current gain occurs



(b) Further, the "hotter" transistor finger . . .



(c) . . . starts to bear whole current

Figure 3.27: Results of electro-thermal simulation with Verilog-A ETFB.

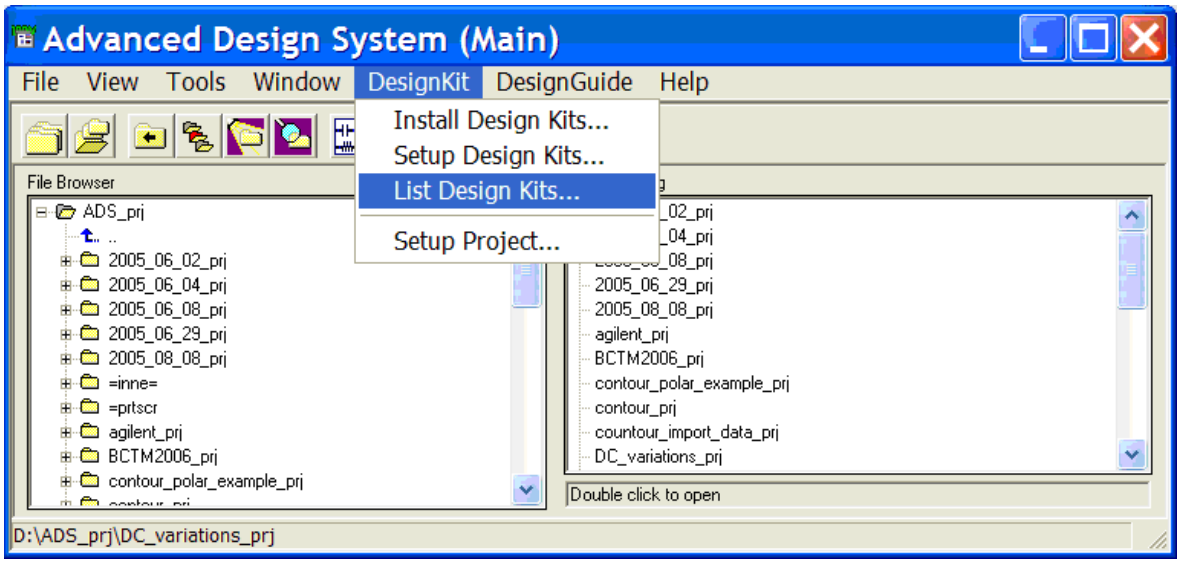


Figure 3.28: How install, configure and obtain an information about Design-Kits in ADS.

```
19 Level: PROJECT
20   File location: N/A
21   Default: enabled
22
23 Level: PROJECT - DC_variations_prj
24   File location:
25   \$HOME/DC_variations_prj/ads.lib
26   Default: enabled
27
28 ======================================
29   Design Kits
30 ======================================
31 Design Kit: ELT_Design_Kit
32   Path: \$HOME/HPEESOF/elt_designkit/
33   Boot file: de/ael/boot
34   Version: version_v1
35   Load level: USER
36   Default: enabled
37   Status: enabled
```

Listing 3.2: ELT Design kit detailed information.

After correct installation, the Design kit is available in menu as shown in Fig. 3.29. Currently, ELT Design Kit has following component libraries:

- ELT Devices.

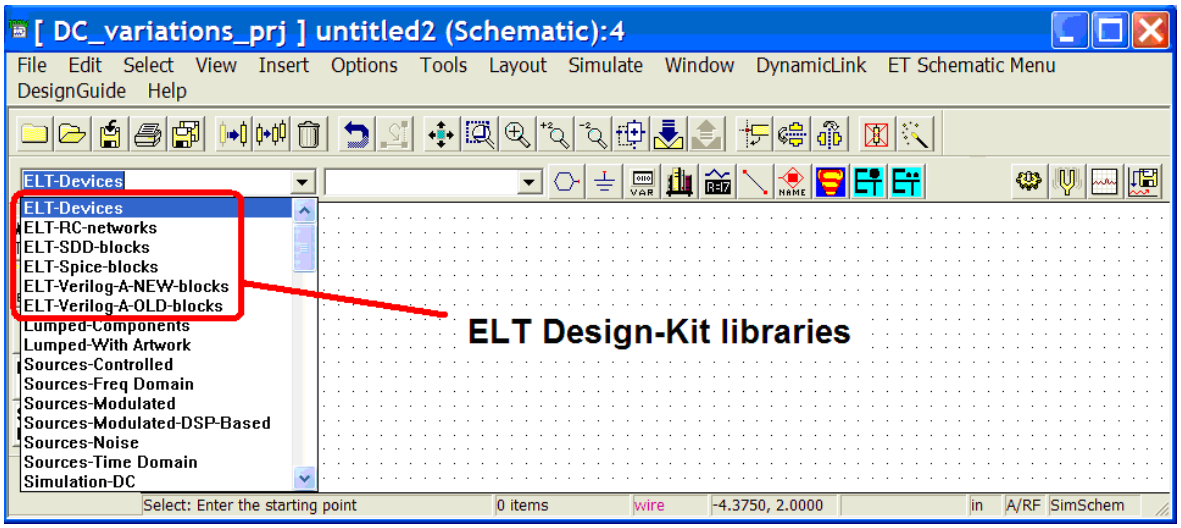


Figure 3.29: ELT Design Kit after correct installation is visible in the ADS schematic menu.

- ELT SDD-blocks.
- ELT RC-networks.
- ELT Spice-blocks.
- ELT-Verilog-A-New-blocks.
- ELT-Verilog-A-OLD-blocks.

In the ***ELT Devices*** library block, following devices are available:

- **AgilentHBT_NPN_Th_TH**, with footprint shown in Fig. 3.30a, is an embedded ADS *AgilentHBT_NPN_Th* device of HBT transistor with additional power dissipating node for electrothermal simulation purposes. The detailed strategy of model creation procedure is described in section 3.3.1. A model like *AgilentHBT_Model* must be attached in order to start a simulation.
- **AgilentHBT_Model**, with footprint shown in Fig. 3.30b, is a standard embedded ADS HBT model both for PNP and NPN devices. It is placed in the library for a user convenience.
- **M504_BJT5_NPN_TH**, with footprint shown in Fig. 3.30c, is an embedded ADS *M504_BJT5_NPN* device of BJT NPN transistor with additional power dissipating node for electrothermal simulation purposes. The detailed strategy of model creation procedure is described in section 3.3.1. A model like *MEXTRAM_504_Model* must be attached in order start a simulation.

- **MEXTRAM_504_Model**, with footprint shown in Fig. 3.30d, is a standard embedded ADS BJT model for both PNP and NPN devices. It is placed in the library for a user convenience.
- **R_TH**, an electrothermal resistor with footprint shown in Fig. 3.30e. Resistor is created in Verilog-A language, as visible in the listing 3.3.

```
1  `include "disciplines.vams"
2  `include "constants.vams"
3
4  module R_TH(p, n, th, pd);
5
6   inout p,n,th,pd;
7   electrical p,n,th,pd;
8   parameter real Ro=0;
9
10  analog begin
11   V(pd) <+ I(p,n) * V(p,n);
12   V(p,n)<+ (Ro*pow(((V(th)+300)/300),2))*I(p,n);
13  end
14 endmodule
```

Listing 3.3: Electrothermal resistor R_TH code

- **Diode_TH**,with footprint shown in Fig. 3.30f, is an electrothermal diode created using Verilog-A language.
- **SDD_rdt_TH**, with footprint shown in Fig. 3.30g, is a BJT electrothermal transistor designed using SDD language. Device is working only in active region, since electrothermal effects occur here. The advantage is the reduced calculation time, since equations describing saturation region are not included.
- **SDD_HBT_TH**, with footprint shown in Fig. 3.30h, is a custom HBT electrothermal transistor model using SDD language. Device is working in active region. Device was used in the [53].
- **SDD_BJT_TH**, with footprint shown in Fig. 3.30i, is a BJT electrothermal transistor designed using SDD langage. Device is working only in active region. Device was used in the [54].

In the ***ELT SDD-blocks*** library block, following devices are available:

- **SDD_1_TH_ ... SDD_50_TH_**, footprints are shown in Fig. 3.31. These electrothermal feedback blocks are supporting up to 50 devices connected to one block.

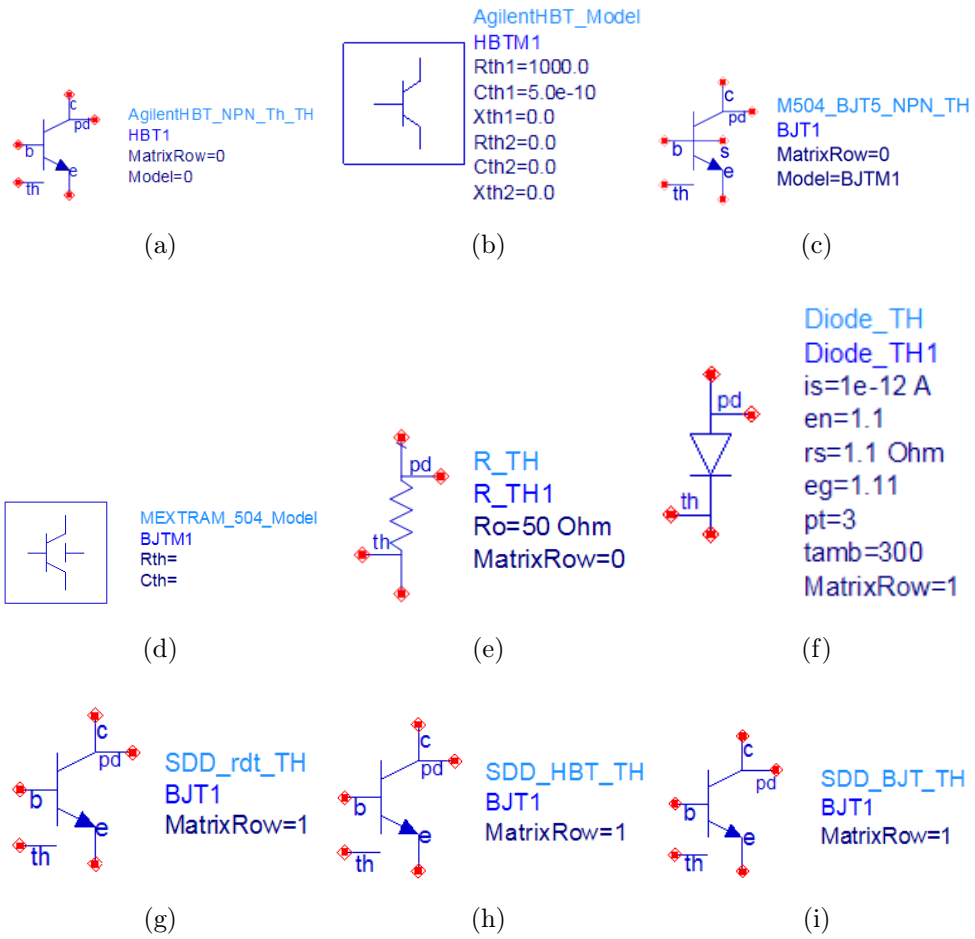


Figure 3.30: ELT devices in the ELT Design Kit.

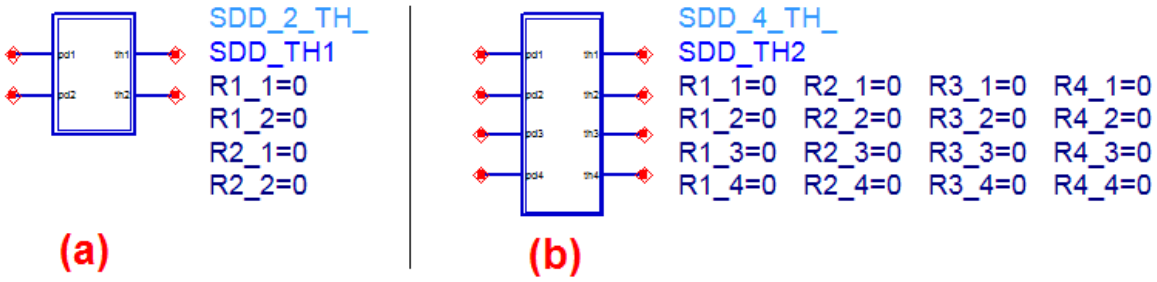


Figure 3.31: SDD blocks (a) for two devices (b) for four devices.

In the ***ELT Spice-blocks*** library block, following devices are available:

⋆ **Net_1_TH_ ... Net_50_TH_**, shown in Fig. 3.32 These electrothermal

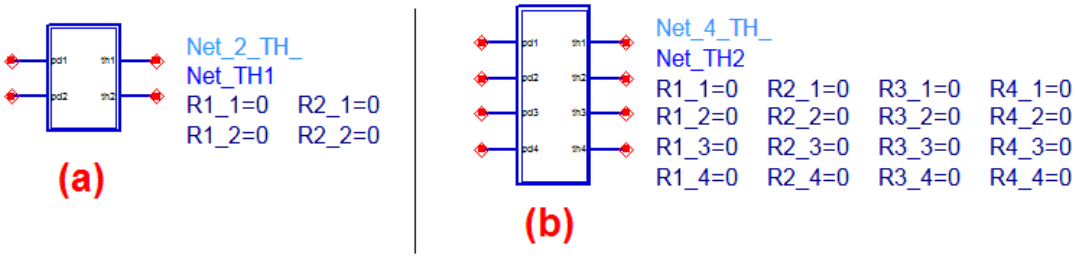


Figure 3.32: SPICE-like blocks (a) for two devices (b) for four devices

feedback blocks are supporting up to 50 devices connected to one block.

In the ***ELT-Verilog-A-New-blocks*** library block, following devices are available:

⋆ **VerilogNew_1_TH_ ... VerilogNew_50_TH_**, shown in Fig. 3.33 These

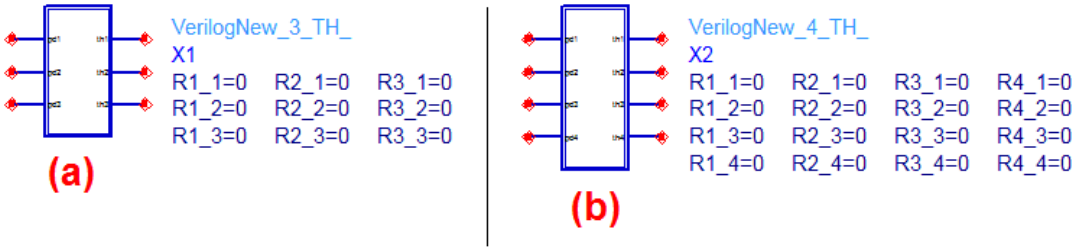


Figure 3.33: Verilog-A NEW thermal feedback blocks (a) for three devices (b) for four devices.

electrothermal feedback blocks are supporting up to 50 devices connected to one block. This is the boundary for the Verilog-A compiler in ADS 2006A.

In the ***ELT-Verilog-A-OLD-blocks*** library block, following devices are available:

⋆ **Rth_5_T_new, ldots , Rth_25_T_new**, shown in Fig. 3.6

This library block is not further under development.

In the ***ELT RC-networks*** library block, following devices are available:

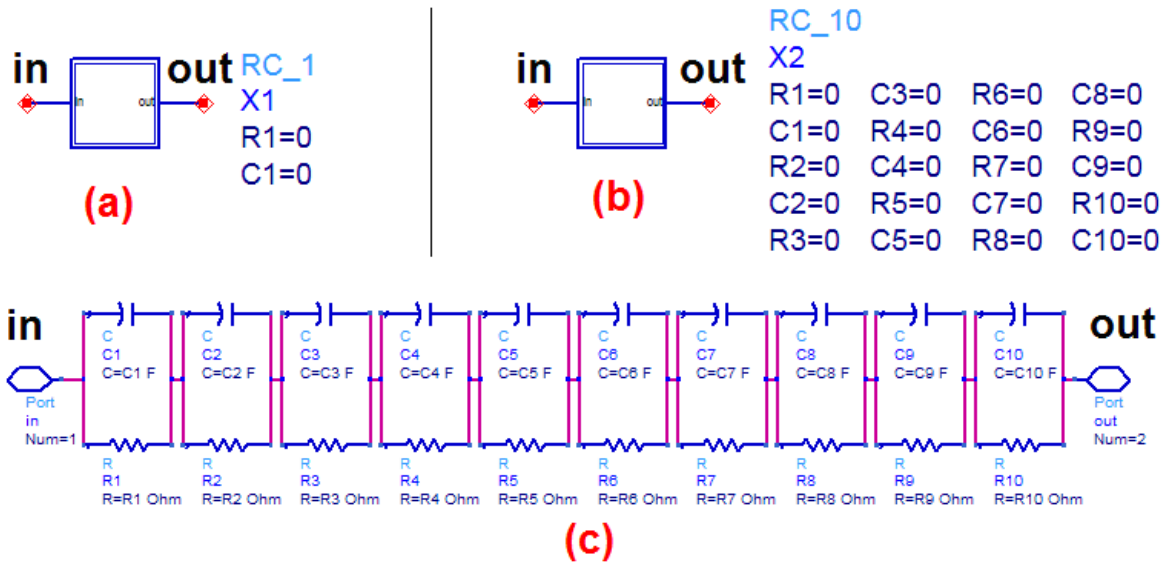


Figure 3.34: RC networks footprint, for the transient case (a) Footprint for 1 pair of RC, (b) Footprint for 10 RC pairs. (c) Inside the footprint for 10 RC pairs is Foster network.

- ⋆ **RC_1, RC_2, . . . ,RC_10**, footprints are shown in Figs. 3.34a &. 3.34b The devices used in this block are used for electrothermal transient simulation, and are to be used with an automated identification algorithm described in the appendix A on the page 123.

3.3.7. Embedded automatic preprocessing routines in ADS

The electrothermal preprocessing routines are realized for ADS schematic and ADS layout programs. The procedure of their use is as follows:

1. Firstly, a user creates the isothermal circuit schematic, e.g. like in Fig. 3.35a.
2. The isothermal schematic is translated into electrothermal one using schematic function "ET PUT & LABEL" function available in ADS schematic toolbar menu (Fig. 3.35b). The translated electrothermal schematic is shown in Fig. 3.36. In particular, function "ET PUT & LABEL":
   (a) Substitutes the devices with a corresponding one
   (b) If the model exists, sets the value of its internal thermal resistance to very high value (1e100).
   (c) Prepares the user-selected electrothermal feedback block. User selects the placement of ETFB in schematic by the mouse click.

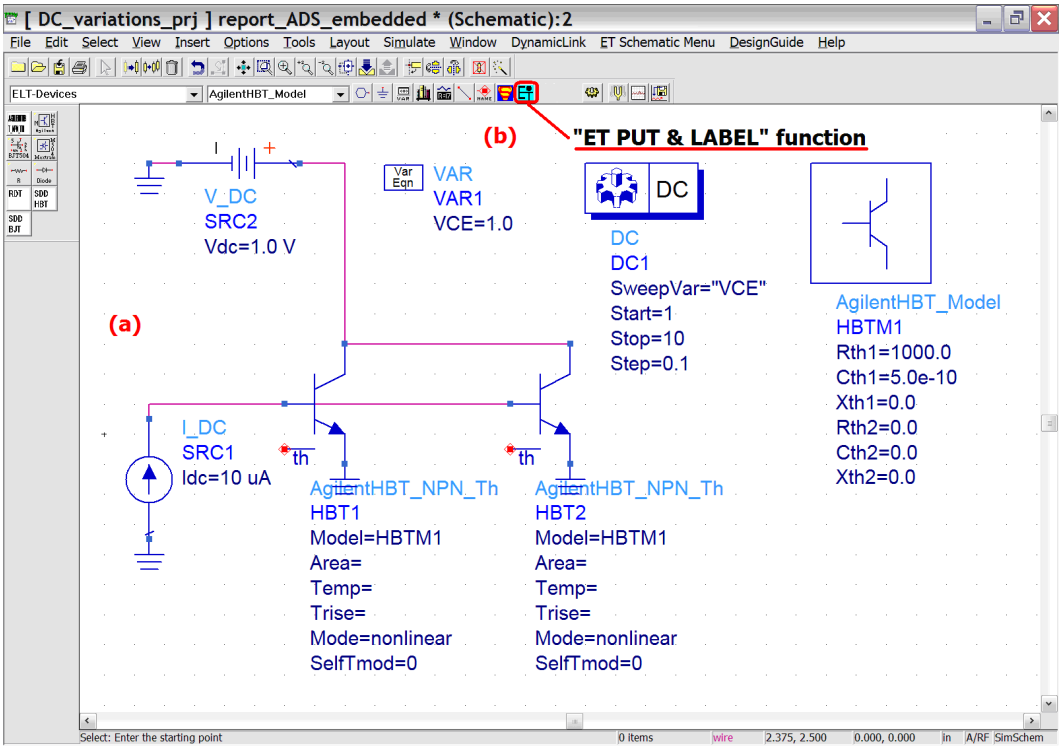


Figure 3.35: (a) Isothermal schematic for two transistors; (b) "ET PUT & LABEL" schematic function button.

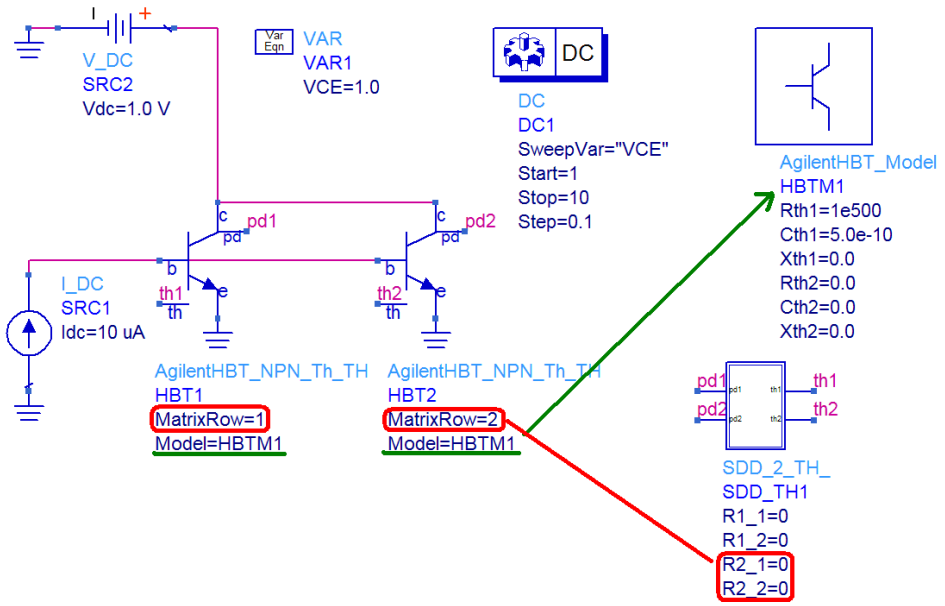


Figure 3.36: Electrothermal schematic for two transistors. Devices are correctly substituted and connected The values of the thermal resistances are not set yet.

(d) Connects the devices to the selected electrothermal feedback block (SDD, SPICE-like or Verilog-A) via "th" and "pd" labels.

3. Next, the preprocessing is moved to the ADS layout program, where the heat sources are identified, and assigned to the particular devices in a schematic. As shown in Fig. 3.37a, two rectangular heat sources are visible. The extraction of thermal resistances assigned to the heat sources is made by "ET RTH VALUE" function (Fig. 3.37b). Function stores the values of thermal resistances in a file.

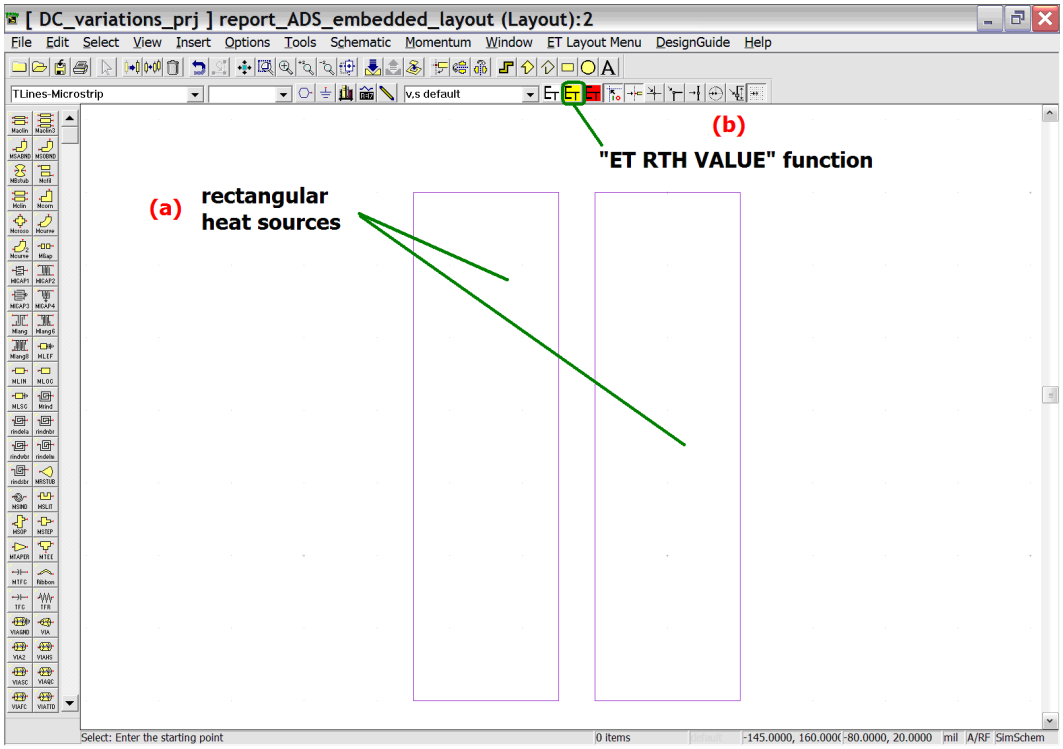


Figure 3.37: ADS layout window with (a) two rectangular heat sources; (b) "ET RTH VALUE" function button is visible above.

4. A file with thermal resistances is saved in an MDF format. Example listing is shown in Listing 3.4

```
1 REM thermal resistance matrix template file
2 REM ---------------------------------------
3 REM leave always this header together with
4 REM date and time and blank line in your
5 REM custom created files
6 REM ---------------------------------------
7 REM  Fri Feb 16 15:08:05 2007
8
9 VAR R1_1(1) = 10000
```

```
10 VAR R1_2(1) = 300
11 VAR R2_1(1) = 300
12 BEGIN DSCRDATA
13 % R2_2(1)
14 9900
15 END
```

Listing 3.4: MDF file with values of thermal resistances calculated from layout by function "ET RTH VALUE".

5. Finally in the ADS schematic program, thermal resistances are extracted from file (Fig. 3.38a) and assigned to the electrothermal feedback block, using the function "ET SET VALUES" shown in Fig. 3.38b.

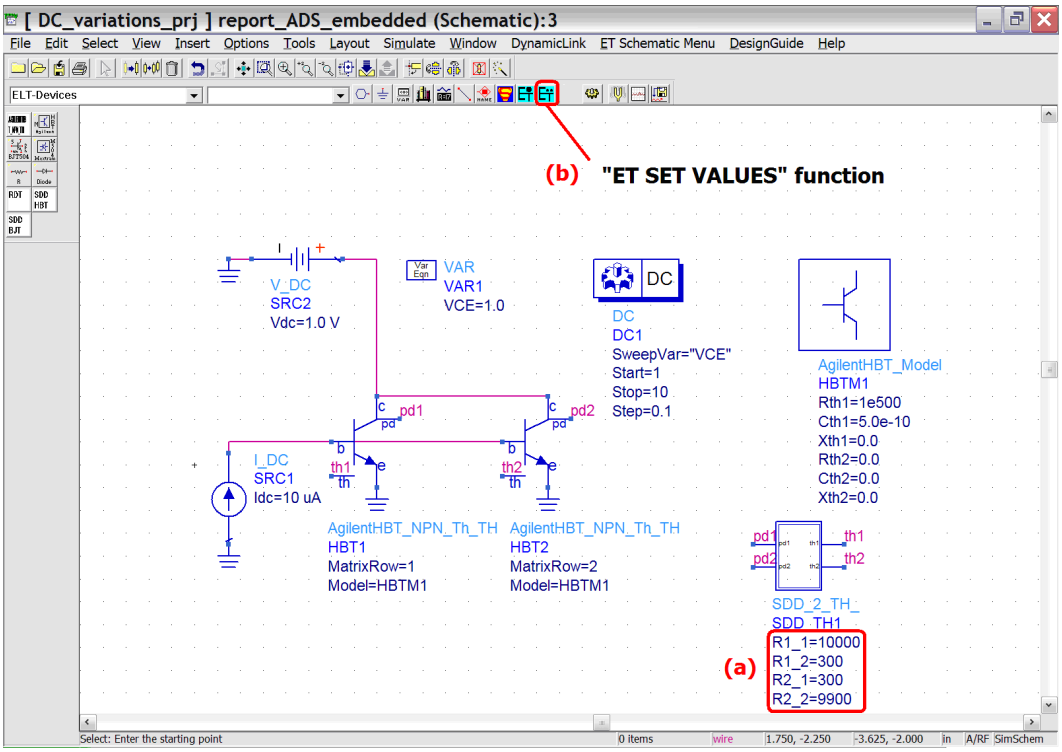


Figure 3.38: Final electrothermal schematic ready for simulation: (a)Thermal resistances are read from file by (b) "ET SET VALUES" function.

Finally, schematic shown in Fig. 3.38 is ready for electrothermal simulation.

All the ADS schematic and layout electrothermal functions are described in the next section.

Schematic functions in “ET Schematic Menu” :

It is possible to call all the menu functions As shown in Fig. 3.39

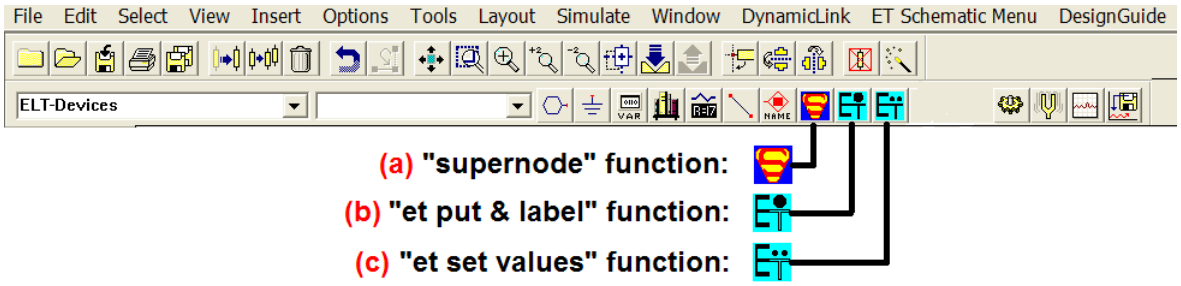


Figure 3.39: Schematic Electrothermal menu functions are available using menu buttons.

Schematic Steady-state preprocessing functions

Et put & label. This function traverses the design searching the electrical components for substitution with electrothermal ones. Then, for the purpose of the electrothermal simulation, it is also possible to set the parameter values of the models (attached to the transistors). setting the Rth/Rth1 parameters values as “1e300”. Developed models do not require to set abovementioned parameter values. In next step user puts the thermal network with the mouse click in the schematic. The placement of the thermal network component is fixed by taking the position from the left mouse click. So in opposite to the "Verilog-A based" solution where the position of the thermal feedback block was fixed, the user has a possibility to control the design and avoid the mistake of overlapping the labels. After putting the chosen feedback component, which may be one of three types (SDD, "Spice-like" or "Verilog New"), the translation of the electrical schematic into electrothermal one is done, however the external thermal resistances are still not set. Now, user must use the second function called "ET SET VALUES" described below.
This function is available in the schematic menu, as shown in Fig. 3.39b.

Et set values. this function sets values of thermal resistances, which are stored in the "Generic MDIFF file" in the "data" directory of the current ADS project. The "Generic MDIFF file" is an ADS standard data file and may be created by the user with custom values of thermal resistances or by one of the layout functions: "RTH VALUE" or "FORCE RTH VALUE", described below.
This function is available in the schematic menu, as shown in Fig. 3.39c.

Additional schematic functions:

Supernode. It is an additional function in order to automatically change the wire/pin label with the mouse click. Wire/pin labels in the schematic serve as the electrical connectors in the schematic. With each mouse click, the final string in the label, which is a number, is automatically increased. For example, lets assume that an input to this function are passed two strings: the "Th" and "1". Clicking anywhere in the schematic, the label changes according to the pattern: "Th1", "Th2", "Th3" etc. With *supernode* function it is also possible to save time during the library/schematic creation, however the main purpose of this function is the automatic labeling of the thermal feedback blocks.
This function is available in the schematic menu, as shown in Fig. 3.39a.

Schematic functions in "ET Layout Menu"

It is possible to call all the menu functions As shown in Fig. 3.40

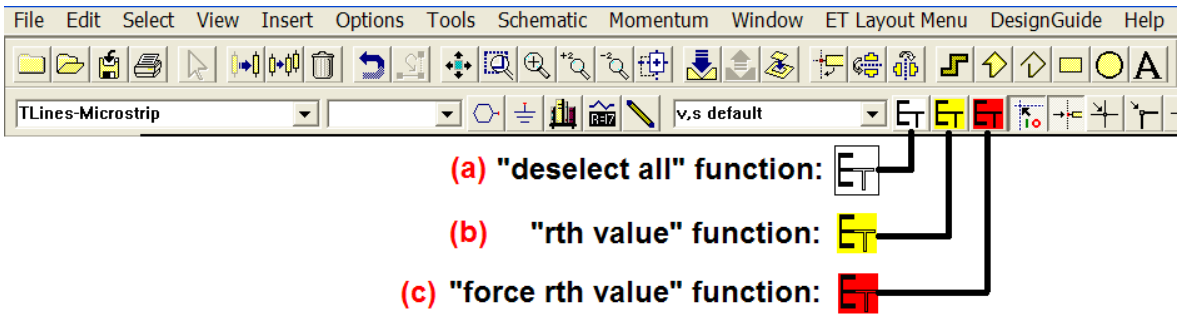


Figure 3.40: Layout Electrothermal menu functions are available using menu buttons.

Layout steady-state preprocessing functions :

Deselect all The calculation of the thermal resistance requires the selection of the heat source in the layout with the mouse click. In order to calculate the correct values of thermal resistances, the user should deselect all layout, because when the layout is huge (industrial layout design) the user could take into account wrong heat sources. In this way, the possibility of mistake is removed before the calculation routine starts. This function cleans also the temporary data in case of mistake.
This function is available in the layout menu, as shown in Fig. 3.40a.

Rth value Here, the user selects exactly one heat source (rectangle placed in the layout by the user) and runs once the function "RTH VALUE". Then, from the menu, the user selects the corresponding device in the schematic and is asked whether he wants to calculate the thermal matrix and store it in the file. If the

selection of the heat sources has finished the user should click "Yes". It means that the calculation of the thermal matrix starts and is based on the actual data. Calculated matrix is saved to Generic MDIFF file and the memory is cleaned from temporary program variables– in particular, the list of the heat sources is empty again. If the user selects "No", it means that the selection process has not finished yet and it is necessary to select others. Before the calculation of the thermal matrix it is not possible to select the same heat sources and assign the same devices twice. This strategy requires accuracy from the user. In case of mistake, when all the heat sources are selected and the user should have clicked "Yes" (in order to calculate the matrix and store in the file), it is not possible to select the heat sources (they were already selected) and assign them to the correct devices in the schematic. Reopening the current ADS project will not help, because function sets some global ADS variable. In this case, the function "FORCE RTH VALUE" should be used or ADS must be restarted.
This function is available in the layout menu, as shown in Fig. 3.40b.

Force rth value The aim of this function is the calculation of the thermal matrix according to the data stored in the memory and the saving of it into the "Generic MDIFF File" format. This function is available in the layout menu, as shown in Fig. 3.40c.

3.4. Approach based on external GUI code

The 3rd electrothermal simulator approach is based on external routines with Graphical User Interface. In this way:

- ⋆ The cooperation with various commercial circuit simulators can be achieved.
- ⋆ Graphical User Interface (GUI) is supported.
- ⋆ The electrothermal simulation program can be easily extended to various types of simulation, not only steady state, but also e.g. transient or ac with relatively small effort.
- ⋆ It is not necessary to have ADS to run the preprocessing.

In the present software version the external preprocessing is based on three types of file formats

1. For schematic: files with extension "iff" (Intermediate Format Files).
   The IFF translator provided by Agilent Technologies, is an EDA framework

integration software product that stores circuit component and connectivity information. This product enables to ex-change design information between ADS and other EDA frameworks that provide an IFF interface. Agilent's IFF Interface enables to generate IFF files from ADS Schematic information as well as receive IFF files from other design environments that support IFF translation. IFF format is compatible with several, market-leading RF simulators like ADS (Agilent), ELDO (Mentor Graphics) and CADENCE (Cadence Systems).

2. For layout: files with extension "dsn" (design files, standard ADS files for layout or schematic).
   ADS uses the same file extensions for layout designs and schematic designs. These files are initially pre-processed and recognized only by ADS simulator.

3. Configuration files.
   Additionally, the support of configuration file has been implemented, in order to store the assignment between heat sources in layout (dsn file) and particular devices in the schematic. This configuration is stored within the file in a specific format, recognized by the electrothermal preprocessor. Once a configuration file is saved, it is possible to choose it instead of layout file. Configuration files can be also created manually

The electrothermal preprocessing program is organised as a "wizard" i.e. as step by step program, that is not organised in one window, but in several adaptive windows of the same size.

3.4.1. External GUI code in practice

Below the demonstration of GUI electrothermal simulator will be performed. In examples will be processed the same isothermal schematic, shown in Fig. 3.41 but using different approaches. In particular (1) the first example will use *schematic iff* file and *layout dsn* file (2) the second example will use the same schematic iff file and configuration file based instead of previously used.
However, in order to pass to the examples, we need to:

1. Export the schematic to iff file This process will be shown below.

2. Create the layout file. This step will be omitted .

Schematic IFF format export

The export to iff format is very easy. For the circuit in Fig. 3.41, the *iff export* operation will be performed. As shown in Fig. 3.42, after clicking **File** > **Export**, appears menu shown in Fig. 3.43. In this window, a user selects the file type "IFF" and the destination

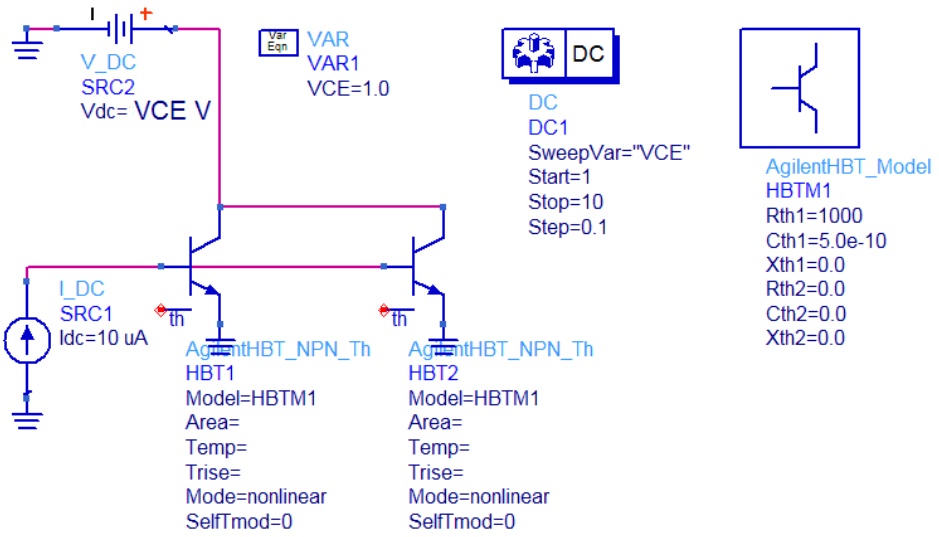


Figure 3.41: Isothermal schematic used in examples with external GUI preprocessor.

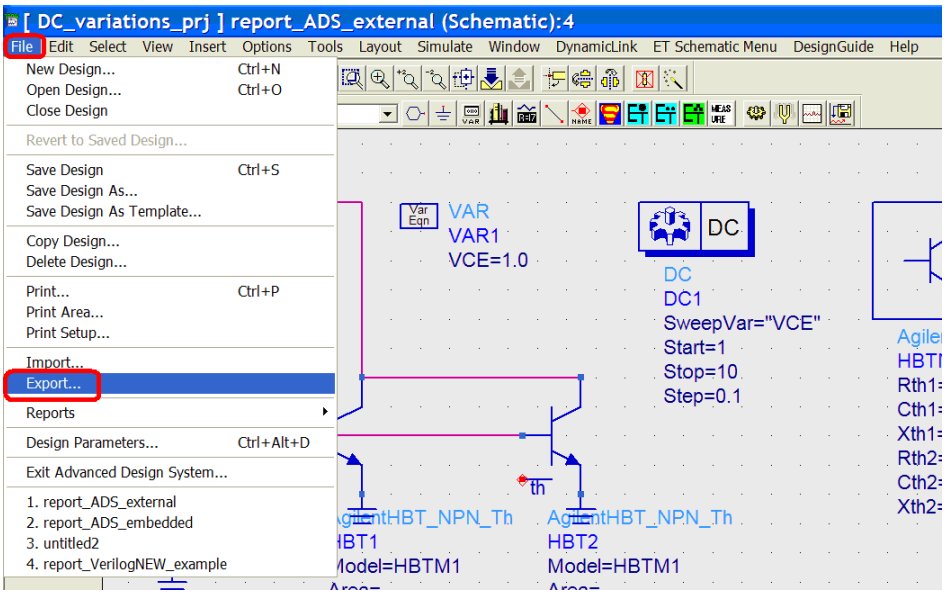


Figure 3.42: ADS IFF export operation steps. (1) From schematic menu user selects "File" and then "Export".

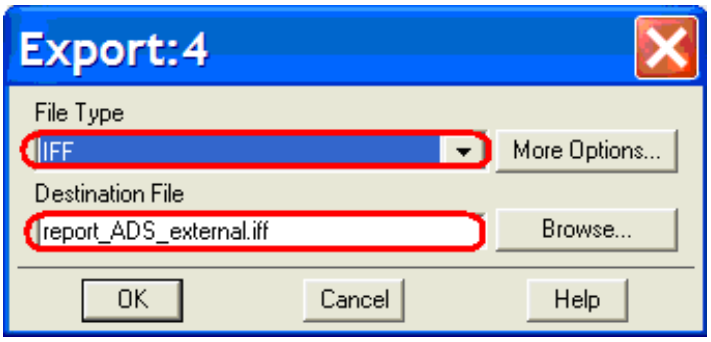


Figure 3.43: ADS IFF export operation steps. (2) IFF export window.

file name. Any other option should be selected, because it might disturb the simulator functionality. Finally, after clicking OK in the window shown in Fig. 3.43, schematic file is exported to *iff format*, and the log file shows in notepad, as demonstrated in Listing 3.5.

```
1  iffexport.log
2  IFF translation ($Revision: 100.149 $)
3
4  Output format: Intermediate File Format
5  Output filename: report_ADS_external.iff
6
7  Reading options file .\iff.opt
8
9  Export Options:
10    HierarchyOption: 2
11    (All Levels - no public library parts)
12    The following projects will be
13    included in the export:
14        D:\ADS_prj\DC_variations_prj
15    DefaultLibraryPath: hpeesoflib
16    The IFF file will be overwritten.
17    Export IFF Version 2
18
19 Reading design
20 D:\ADS_prj\DC_variations_prj\
21 networks\report_ADS_external
22   Reading element DC DC1
23   Reading element AgilentHBT_Model HBTM1
24   Reading element AgilentHBT_NPN_Th HBT1
25   Reading element AgilentHBT_NPN_Th HBT2
26   Reading element VAR VAR1
27   Reading element V_DC SRC2
```

```
28   Reading element I_DC SRC1
29 Finished reading design
30 D:\ADS_prj\DC_variations_prj\
31 networks\report_ADS_external
32
33 IFF file report_ADS_external.iff created.
```

Listing 3.5: IFF export log.

After the *iff export* procedure, if not selected differently, an *iff* file is stored in the current project directory.

GUI Example: IFF schematic file & DSN layout file

Step 1: Initial window: select the method, design kit & project dir path After opening the program, the window shown in Fig. 3.44 appears. The user has following options to select:

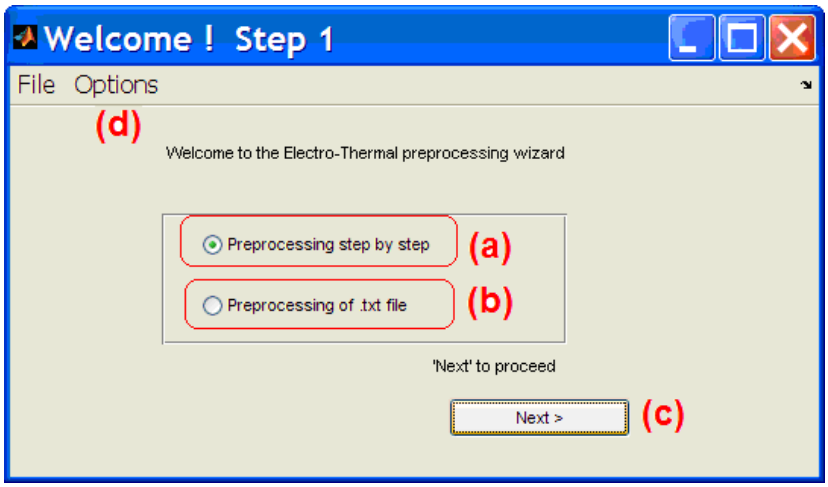


Figure 3.44: GUI electrothermal simulator initial window. Preprocessing using (a)IFF and DSN file (b)IFF and TXT configuration file. (c) Next button (d) Options menu.

(a) "Preprocessing step by step" (Fig. 3.44a).

(b) "Preprocessing of .txt file" (Fig. 3.44b).

As mentioned in previous section, this example will be performed using the option "Preprocessing step by step".

Before selecting the "Next button" (Fig. 3.44c), the "Design Kit Path" must be selected in the Options menu (Fig. 3.44d & Fig. 3.46a). It is not possible to pass to another step without selecting the “Design Kit Path” - the program will return error,

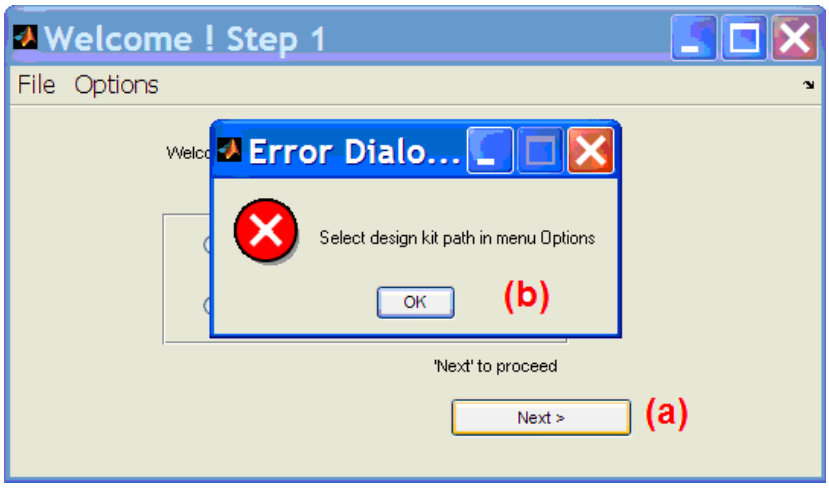


Figure 3.45: The user must set the Design Kit Path in Options menu, otherwise (a) after clicking next button, (b) error window appears.

as shown in Fig. 3.45. The options menu (Fig. 3.46), lets the user to select (a) the Design Kit Path (Fig. 3.46a) and (b) Project dir path (Fig. 3.46b). (a) "Design Kit Path" value is necessary to correctly import the file after preprocessing, otherwise devices and electrothermal feedback blocks can not be recognized by the simulator. (b) "Project Dir Path" selection is optional - lets the user to decide where access and store files. The selection of the "Design Kit Path" is made with Browse button (Fig. 3.47a) and the final selection is made with Accept button (Fig. 3.47b). It is possible to set

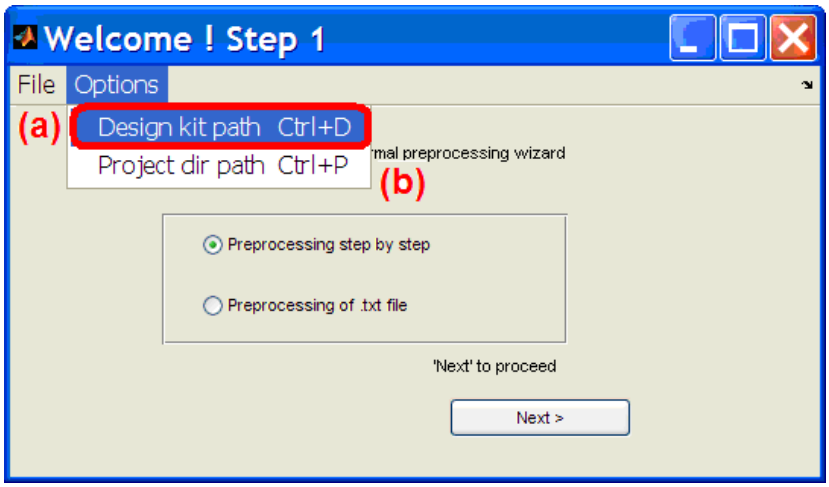


Figure 3.46: Options menu: (a) Design Kit Path option (b) Current dir option.

the current directory by clicking the "Project dir path" button (Fig. 3.46b). "Project Dir Path" can be selected using Browse accept buttons (Fig. 3.48). If "Project Dir Path" option is not selected, the warning will appear (Fig. 3.49). however the pass to the next stage is performed. The warning is visible only in the pass between first and second stage. It is just a matter of convenience to set "Project Dir Path" where the

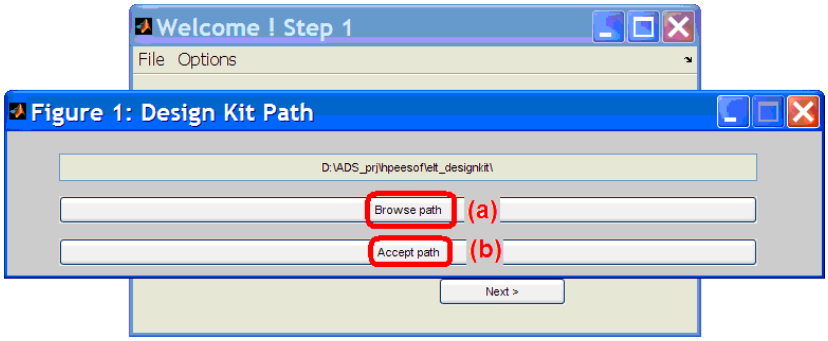


Figure 3.47: Design kit path selection window with (a)browse button and (b)accept path button.

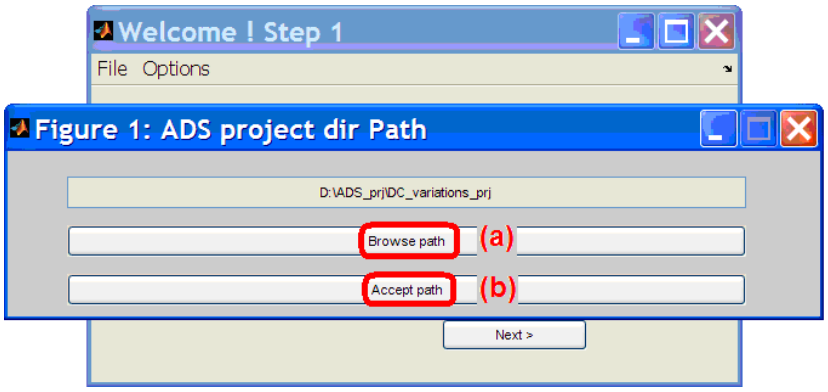


Figure 3.48: Project dir path selection window with (a)browse button and (b)accept path button.

Figure 3.49: Warning dialog, shown passing from first step to the second step if "Project Dir Path" is not set.

output files are stored.
Summarizing the first step, in order to process the IFF schematic file and DSN layout file, following options have been selected:

1. "Processing step by step"(Fig. 3.44a).
2. "Design Kit Path" has been set to "D:\ADS_prj\hpeesof\elt_designkit\" (Fig. 3.47a).
3. "Project Dir Path" has been set to "D:\ADS_prj\DC_variations_prj\" (Fig. 3.48b).

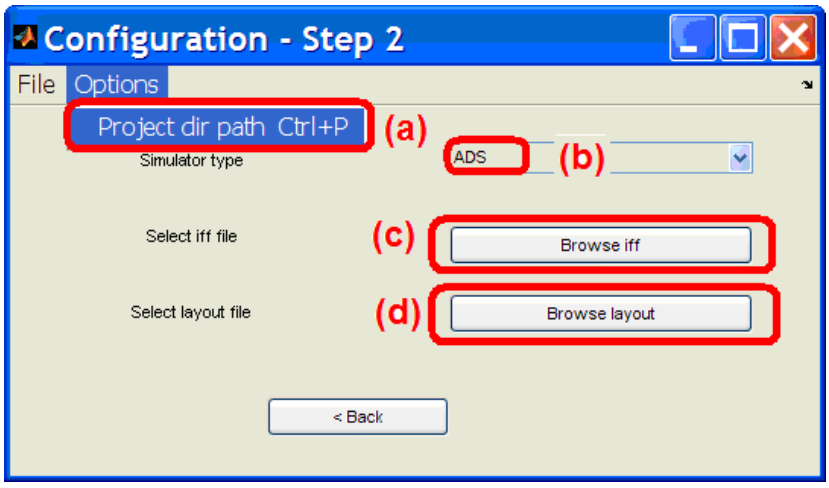


Figure 3.50: GUI electrothermal simulator window for the second step (a) options menu (b) simulator selection menu (c)IFF file browse button (d) layout browse button.

Step 2: Selecting the IFF schematic file & DSN layout After passing to the second step, the "Design Kit Path" option disappears in menu, however the "Project dir path" option is still available (Fig. 3.50a).

The IFF schematic file is opened using the button "Browse iff" (Fig. 3.50c), the layout DSN file using the button "Browse layout" (Fig. 3.50d). Simulator type is set using the context menu (Fig. 3.50b). Three simulator types are available: (1) ADS, (2) Eldo and (3) Cadence, however currently there is no difference between any of them.

After correctly selecting the IFF schematic file, "Browse iff" button disappears and on its place, the name of the file shows (Fig. 3.51). The same happens when a DSN layout file is correctly selected, however the procedure of the layout file selection, is restrictive and may produce warnings if the layout file does not contain heat sources. In particular, if a selected DSN layout file does not contain heat sources, a warning is shown (Fig. 3.52). In previous section it was described that DSN layout files have the same extension as schematic files. It might happen also, that both schematic and layout are in the same file, so the code is searching for the heat sources in the layout

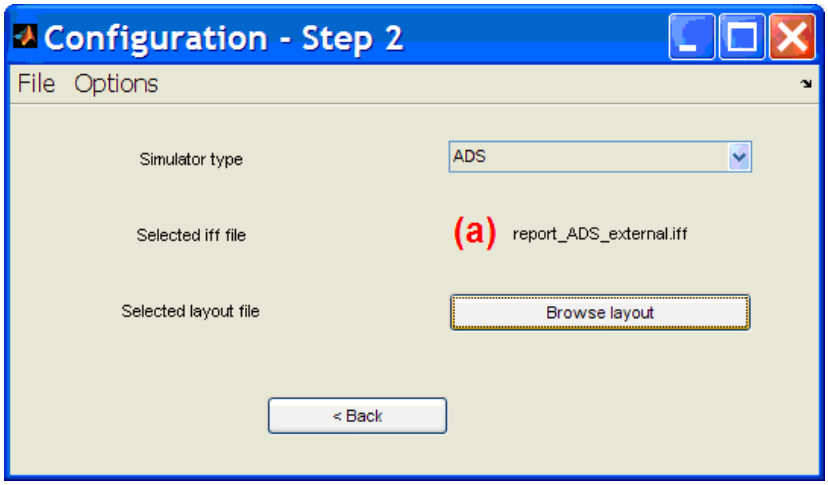


Figure 3.51: Second preprocessing step: (a) after selecting an IFF schematic file, its name appears.

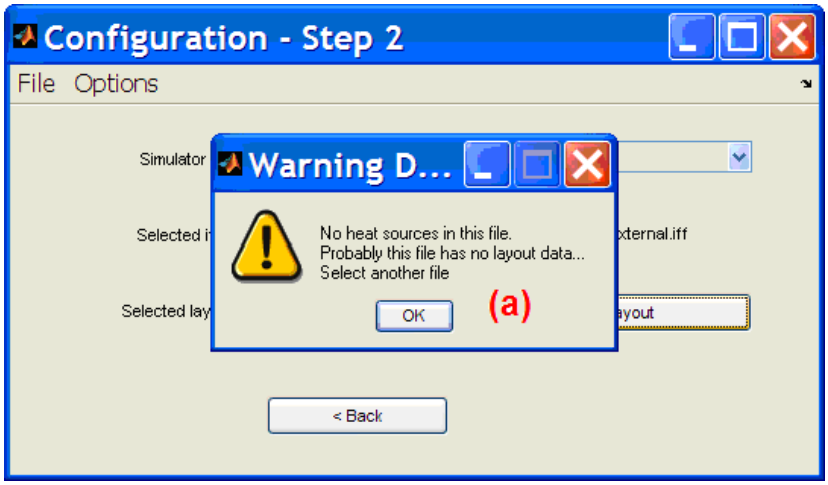


Figure 3.52: Second preprocessing step: (a) selected DSN layout file that does not contain heat sources produces warning.

part of the file. If heat sources are not found, the warning appears (Fig. 3.52a).
After the correct selection of the DSN layout file, exactly as in the case with IFF schematic file, the “Browse layout” button disappears (Fig. 3.53a) and the name of the file is visible. In order to pass to the next step, the "Next button" must be visible (Fig.

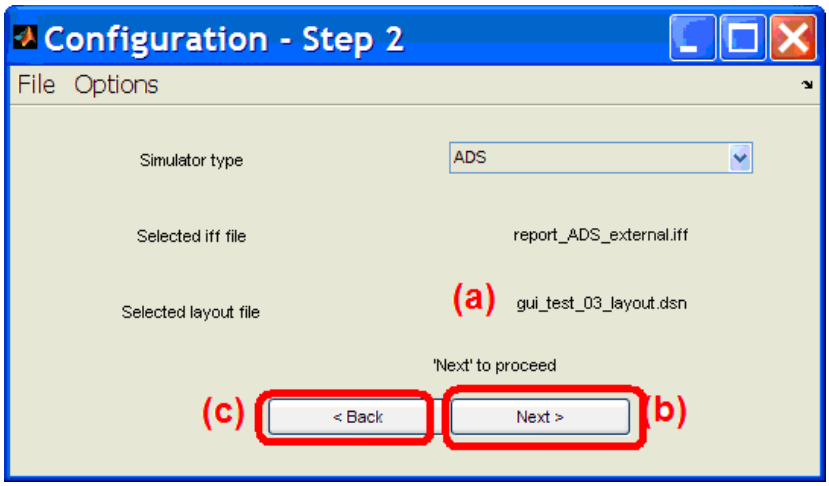


Figure 3.53: Second preprocessing step: (a) after selecting a DSN layout file, its name appears. (b) “Next” button is visible.

3.53b), an IFF schematic file and DSN layout file must be selected correctly, otherwise the "Next button" will not appear.
The "Back" button (Fig. 3.53c) lets to turn back to the previous preprocessing step. It may happen in case of mistake or when a user wants to select another data.

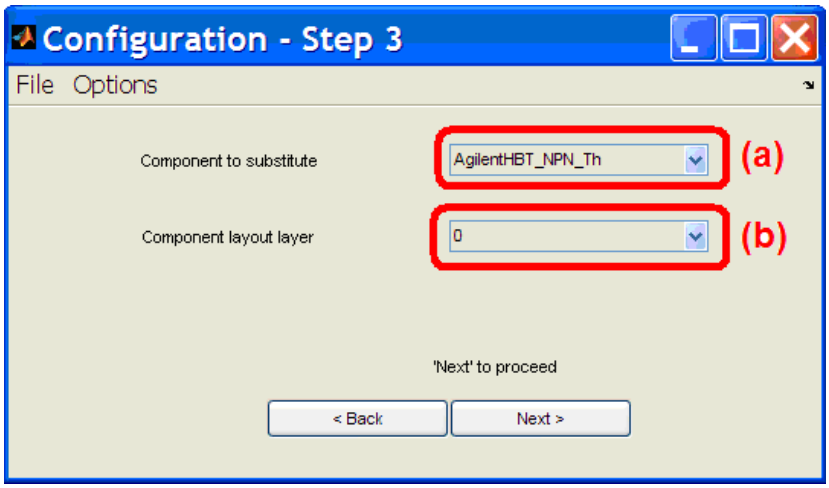


Figure 3.54: Third preprocessing step: (a) selecting the component to substitute, here "Agilent_NPN_Th" transistor (b) Selecting the heat source layer.

Step 3: Selecting the IFF schematic file & DSN layout In the third preprocessing step, the component which in preprocessing will be substituted can be selected from the pop-up menu (Fig. 3.54a and Fig. 3.55a). The heat source layer is selected in another pop-up menu (Fig. 3.54b). In comparison with external command-line approach, the layers are not named, because the "layers file" is not processed. Therefore assignment "layer name" to "layer number" is not performed – a user must know on which layer number, the heat source is stored.

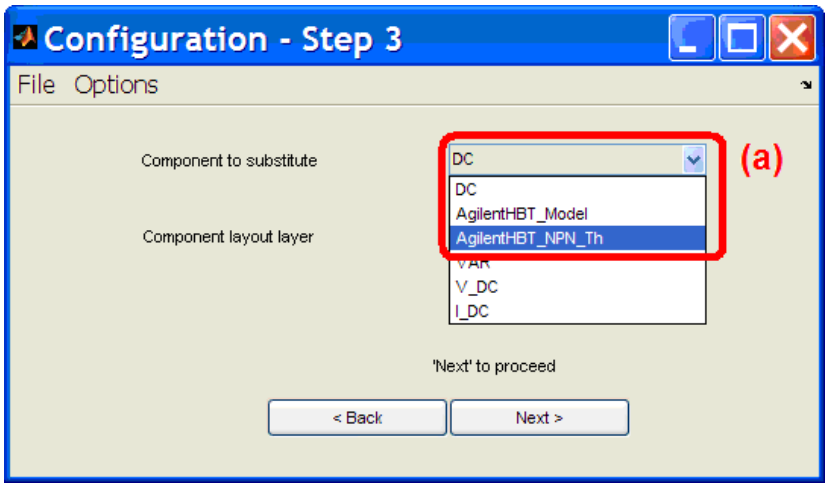


Figure 3.55: Third preprocessing step: (a) from the pop-up menu all components available in IFF schematic.

After the component to substitute has been selected (Fig. 3.55a) ; in this example the "AgilentHBT_NPN_Th", it is possible to pass to the next, optional fourth step (Fig. 3.56a). In this example the fourth preprocessing step will be performed and shown in the next paragraph.

Step 4 (Optional): Setting the model parameters In this step, the thermal resistance Rth of the selected model is set to a very large value, 1e100 in order to at-tach the external thermal network correctly. In other words, the self-heating resistance value. This process should be done with attention, because the model can have more than one thermal resistances, as shown in Fig. 3.9, so the code searches for non-zero resistances and changes them for a high value, here 1e100 Ohms. It might happen, that the device does not require the external model, like in the case with devices developed in Verilog-A behavioral language or via SDD components. In this case this step should be omitted. In the next preprocessing step, the heat sources will be assigned to the devices selected in third step.

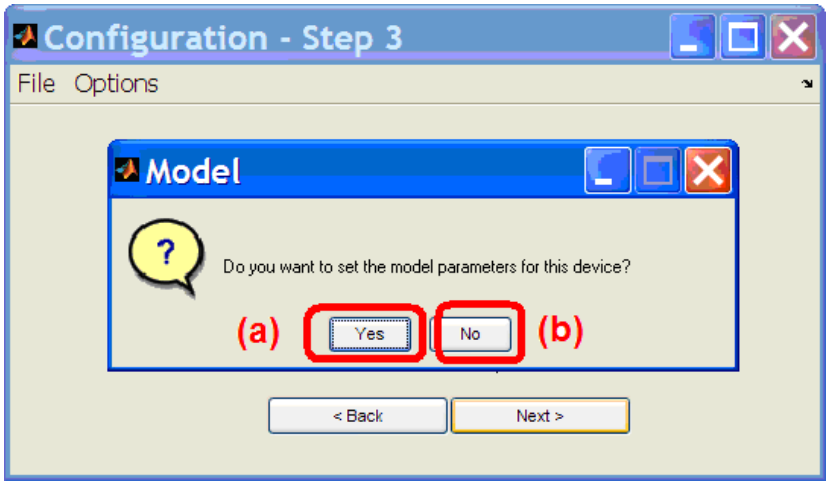


Figure 3.56: Third preprocessing step, where (a) selecting "Yes" the passing to the fourth preprocessing step is performed (b) selecting "No" fourth preprocessing step is omitted.

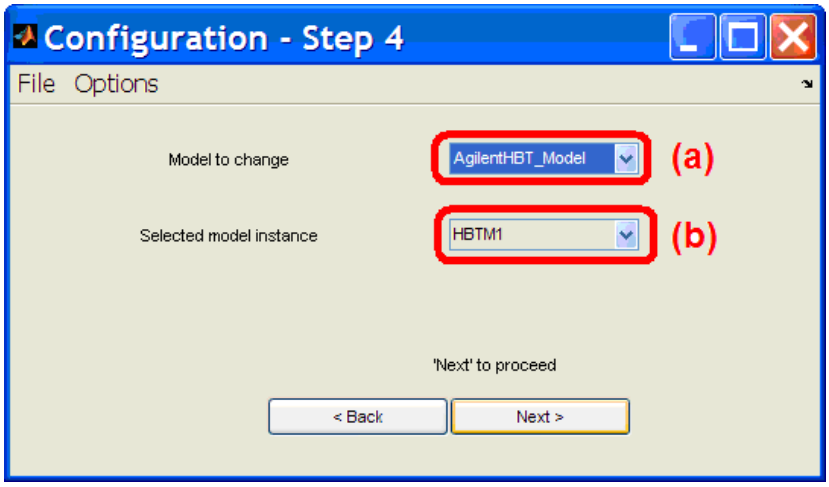


Figure 3.57: Fourth preprocessing step: (a) pop-up menu in order to select the name of the model (b) pop-up menu with the model instance name, selected in (a).

Step 5:Assigning heat sources to devices. Calculating a thermal resistance matrix. In the fifth preprocessing step the heat sources are assigned to the particular devices, according to the device type selected in the third preprocessing step. As shown, the HBT1 device (Fig. 3.58a) is assigned to the device with coordinates shown in the pop-up menu (Fig. 3.58b). The heat sources represented using coordinates x1, y1, x2, y2. In order to explain the significance of "x1, y1, x2, y2" the detailed information is shown in Fig. 3.59. As shown in Fig. 3.60, the *HBT1* device has been assigned,

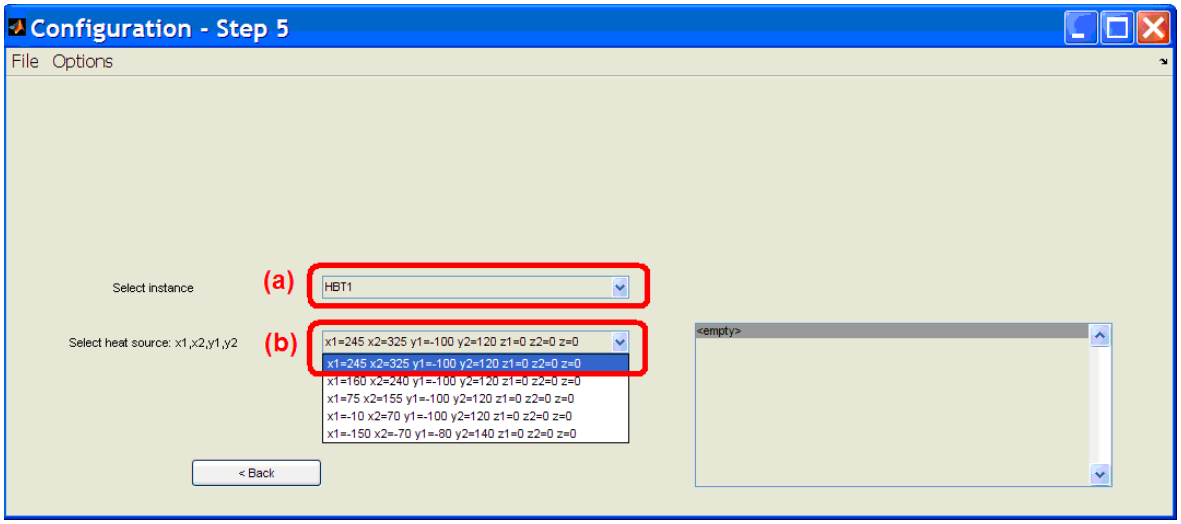


Figure 3.58: Fifth preprocessing step (a) assigning device instances to (b) particular heat sources coordinates.

Figure 3.59: Heat source coordinates are represented by two pairs of coordinates. x1,y1 x2, y2.

the *HBT2* will be assigned to the most distant one, in order to obtain small mutual thermal resistance. The final assignment heat-source-devices is shown in Fig. 3.61. It is important to note that the final list is available in the table shown in Fig. 3.62. As it is visible, all thermal parameters can be easily changed, the table rows (which are thermal matrix rows) can be deleted, moved and arranged in order to obtained the desired data. If any of the parameters has been changed, the "Assign data" button changes the colour for red, and it is not possible to continue the preprocessing, until it is clicked by the user (Fig. 3.63). Empty space is treated also as the change and the button changes to red. As soon as the changes are updated and final thermal matrix

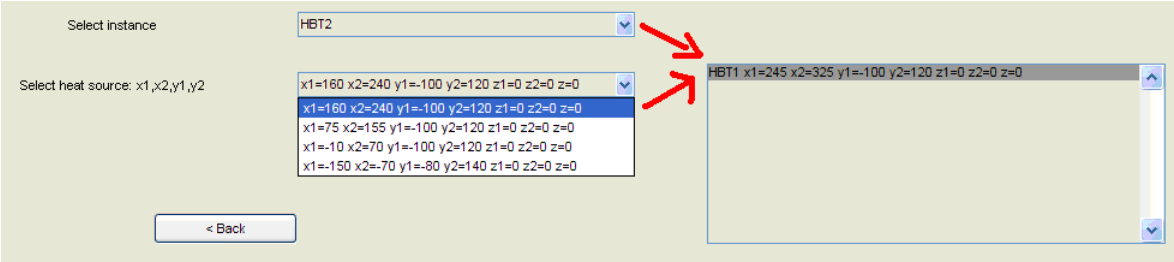


Figure 3.60: Fifth preprocessing step: after assignement heat source-device, the table on the right side is increasing.

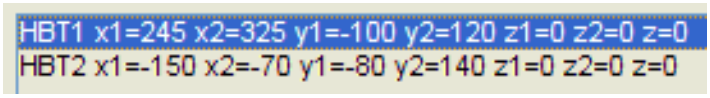


Figure 3.61: Fifth preprocessing step: final assignment: devices-heat-sources.

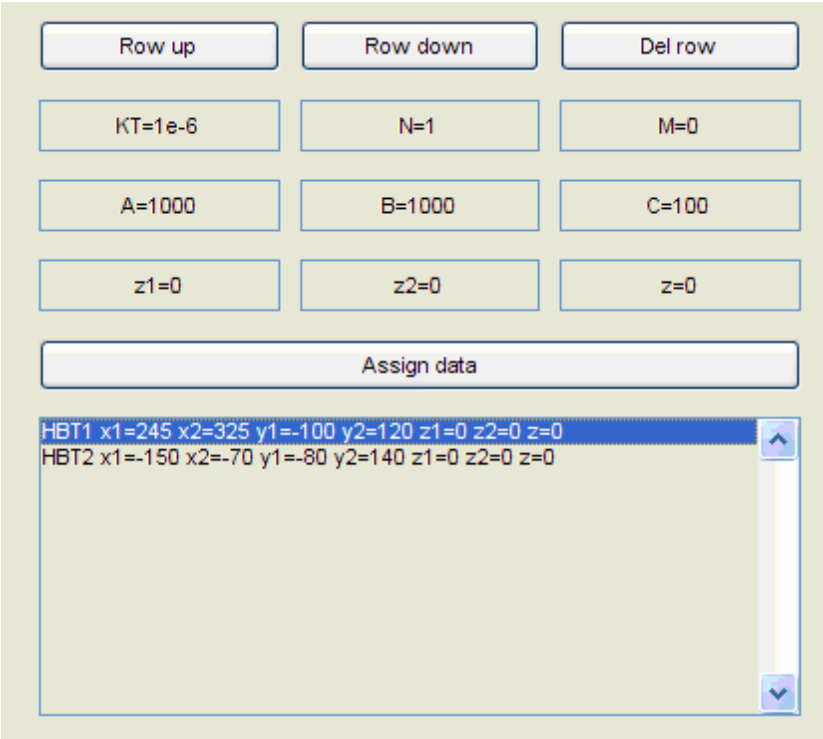


Figure 3.62: Fifth preprocessing step: Thermal table can be easily arranged and changed according to the user's desire.

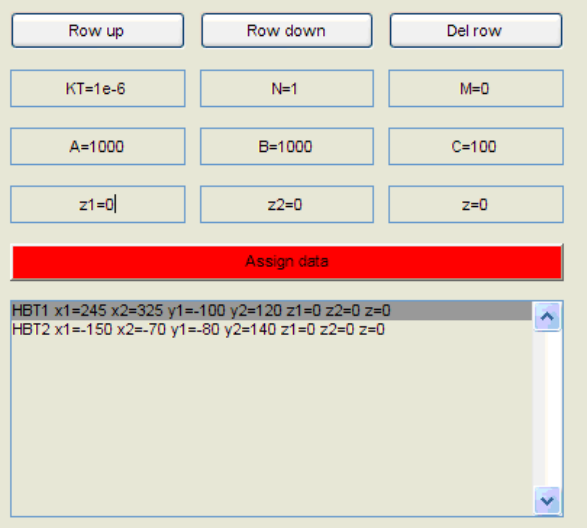


Figure 3.63: Fifth preprocessing step: After any change in the thermal matrix table, the "assign data" button must be pressed to continue.

is to be preprocessed, the next button should be clicked. The window for saving the configuration file appears and now it is possible to save the configuration to the text file (Fig. 3.64). In case of problems with the configuration file path, the application returns an error (Fig. 3.65). If the configuration file is saved, clicking on the "Next

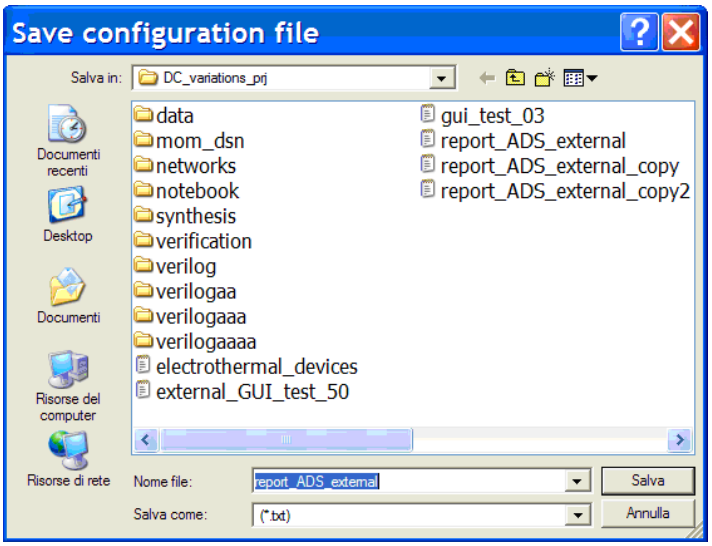


Figure 3.64: Fifth preprocessing step; Configuration file save window.

button" the calculation process is to be started (Fig. 3.66). If the thermal resistance matrix size is less than ten rows, the calculated values are printed in the application console window. In this example is used a 2x2 matrix, because we have two devices HBT1 and HBT2. The output from the console is shown in the Listing 3.6.

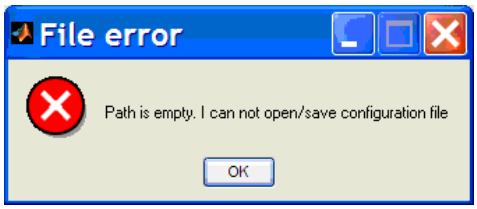


Figure 3.65: Fifth preprocessing step: Configuration path is empty, application returns an error. The preprocessing is stopped.

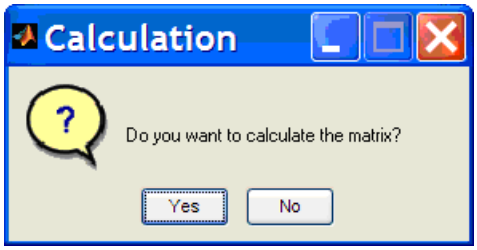


Figure 3.66: Fifth preprocessing step, thermal matrix computation is performed after clicking yes.

```
1 Counting 1 of 4 ...
2 Counting 2 of 4 ...
3 Counting 3 of 4 ...
4 Counting 4 of 4 ...
5 ans =
6      3.173167296273480e+003     4.253564633496586e+001
7      4.253564633497200e+001     3.173167296273480e+003
```

Listing 3.6: Fifth preprocessingh step: preprocessing output; thermal matrix

Now it is possible to pass to the next preprocessing step.

Step 6: Selecting substituting device In this preprocessing step the isothermal devices are substituted with electrothermal ones, as shown in Fig. 3.67. After the electrothermal model has been selected, it is possible to pass to the next preprocessing step where electrothermal feedback block is selected and inserted.

Step 7: Selecting electrothermal feedback block & its position, saving the output file In this preprocessing step, a type of electrothermal feedback block is selected (Fig. 3.68a). There are three types of electrothermal feedback blocks: (a) SDD (b) SPICE-like (c) Verilog-A.
The coordinates of the placement of the ETFB's are selected by clicking the red "Accept coordinates" button (Fig. 3.68b). A user can accept coordinates suggested by the program (Fig. 3.69) or can input his own in order to avoid possible overlapping

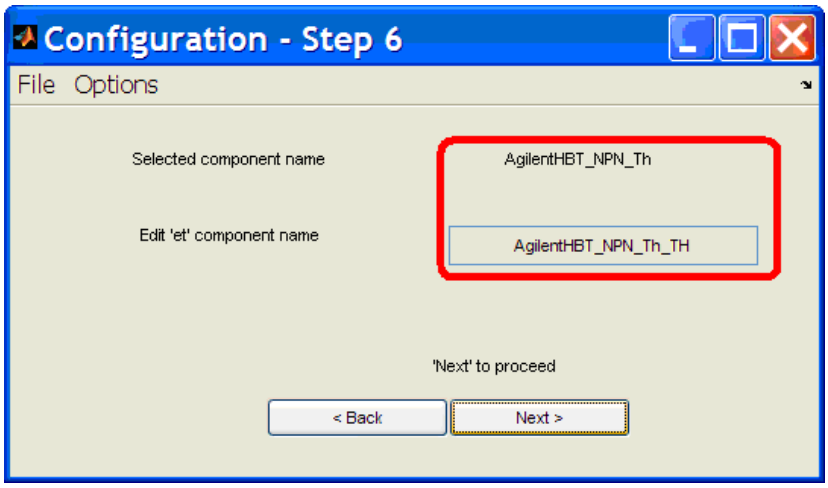


Figure 3.67: Sixth preprocessing step: Selecting the substituting model.

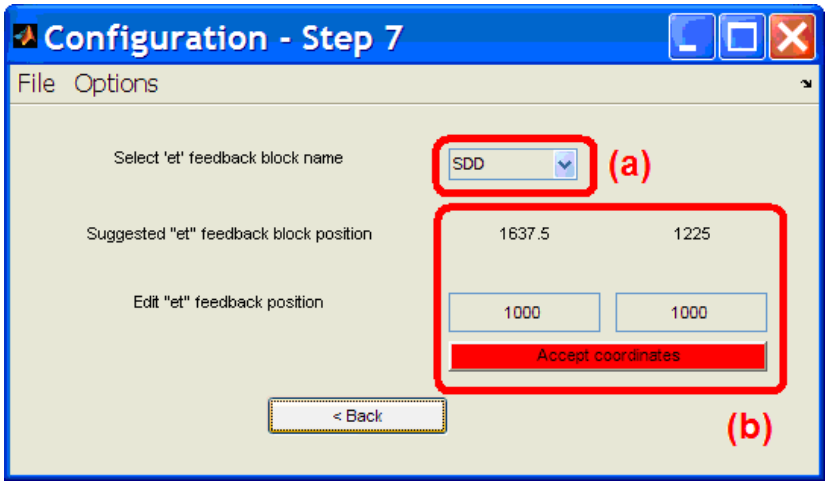


Figure 3.68: Seventh preprocessing step: Selecting (a)the electrothermal feedback block type (b) coordinates of the placement in the schematic.

of the ETFB with other components. After this step the preprocessing is finished and

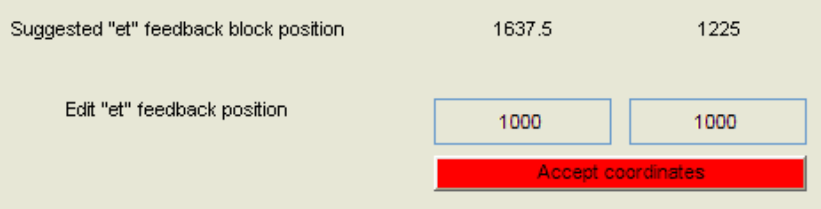


Figure 3.69: Seventh preprocessing step: electrothermal feedback block coordinates can be selected by the user.

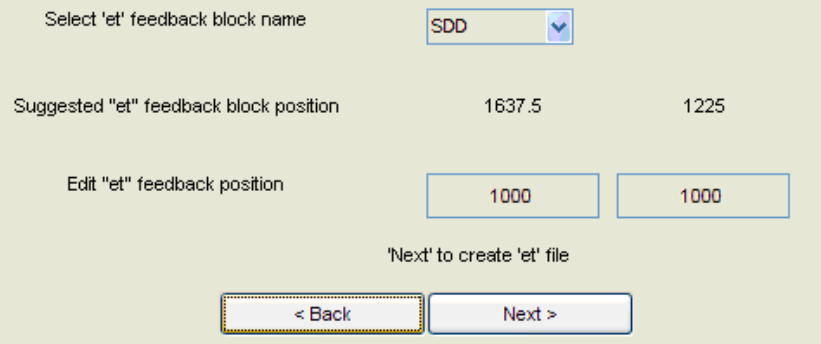


Figure 3.70: Seventh preprocessing step: The electrothermal feedback block coordinates have been accepted by a user.

it is possible to save the file (Fig. 3.71 and Fig. 3.72). The file is saved to the directory

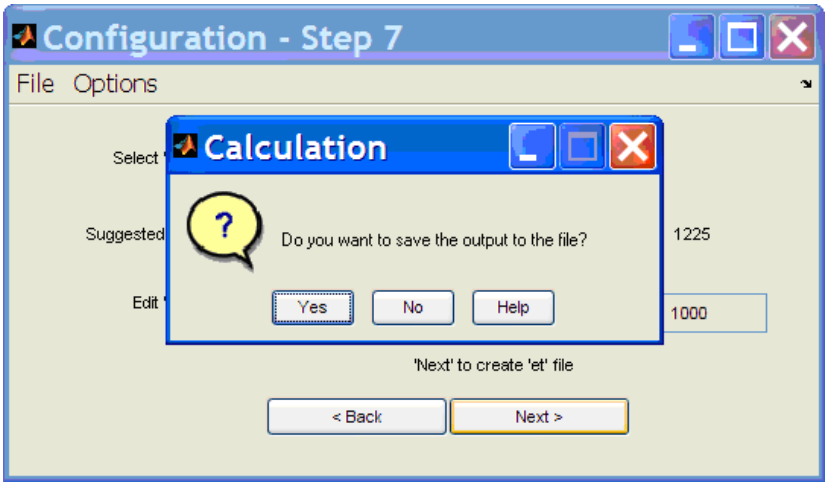


Figure 3.71: Seventh preprocessing step: Saving the output IFF schematic file.

specified in the "Project dir path" option in the Option menu. Note that the internal name of the schematic is changed, that is a string is added to the design name in order

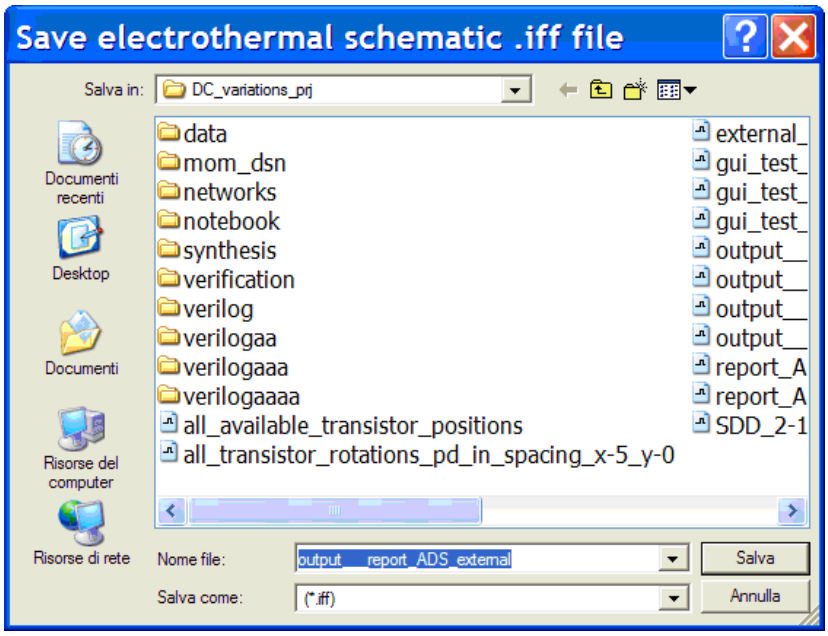


Figure 3.72: Seventh preprocessing step: saving the output file

to avoid the deletion of the original schematic during the import procedure. Final preprocessing window is shown in Fig. 3.73.

Appendix: What was not shown during the example... In the following example, several GUI features have been not shown. In particular:

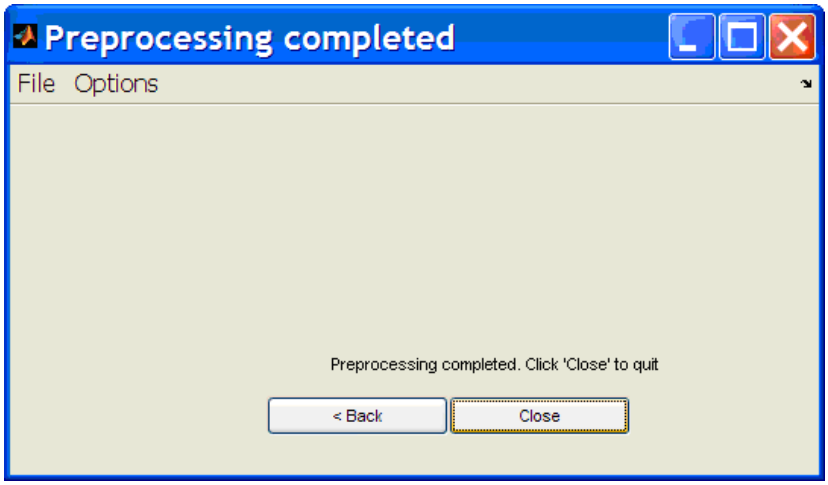


Figure 3.73: Final preprocessing window.

⋆ It is always possible to use "Back button" in order to select another data

- ⋆ It is possible to use "txt configuration file" after the preprocessing in order to avoid assignment of the heat sources to the devices.
- ⋆ Preprocessor is still under construction, so many errors may occur, however the author tried to prepare it as best as possible.
- ⋆ As visible in Fig. 3.74, there are some additional routines, that preserve the user from closing the application.

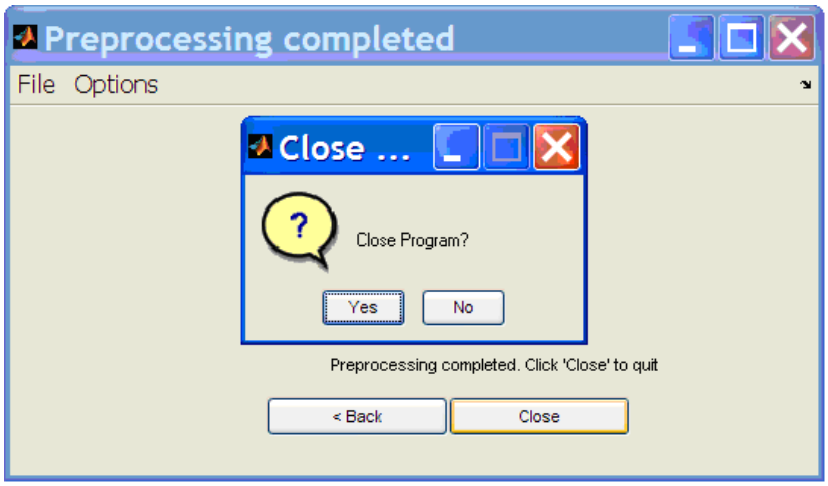


Figure 3.74: Anti-close program routine.

3.4.2. Simulation results

Simulation 1 — Case of two paralleled BJTs

Lets consider two parallelled BJTs (i.e., with common base, emitter, and collector terminals) biased in common-emitter mode at a constant total base current I_{BTOT}. The structure can be regarded as a 2-finger device. It is clear that, due to the inherent system symmetry, in principle the fingers should share the same amount of current regardless of the applied V_{CE}. However, due to unavoidable small differences (e.g., in layout, doping profiles, contact resistances), once critical basing conditions are reached, one elementary transistor starts bearing more current, whilst the other one tends to become dry [55], [56], [20], that is, a "bifurcation" phenomenon arises. A slight discrepancy between the fingers is taken into account by introducing a small difference between the "internal" parasitic resistances of the two transistors (namely, 0.05 Ω). Subsequently, the thermal resistance matrix is inserted "directly" by considering values typical of silicon-on-glass (SOG) transistors developed for RF applications, which have been shown to be extremely prone to electro-thermal effects due to the poor thermal conductivity of the materials surrounding the active silicon area [14], [57]. In particular,

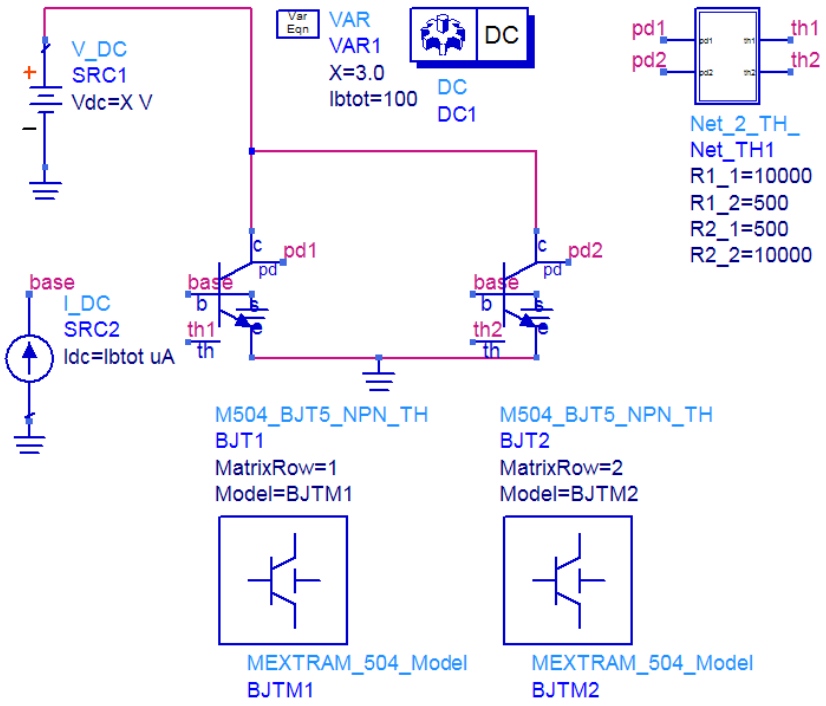


Figure 3.75: Schematic to Simulation 1

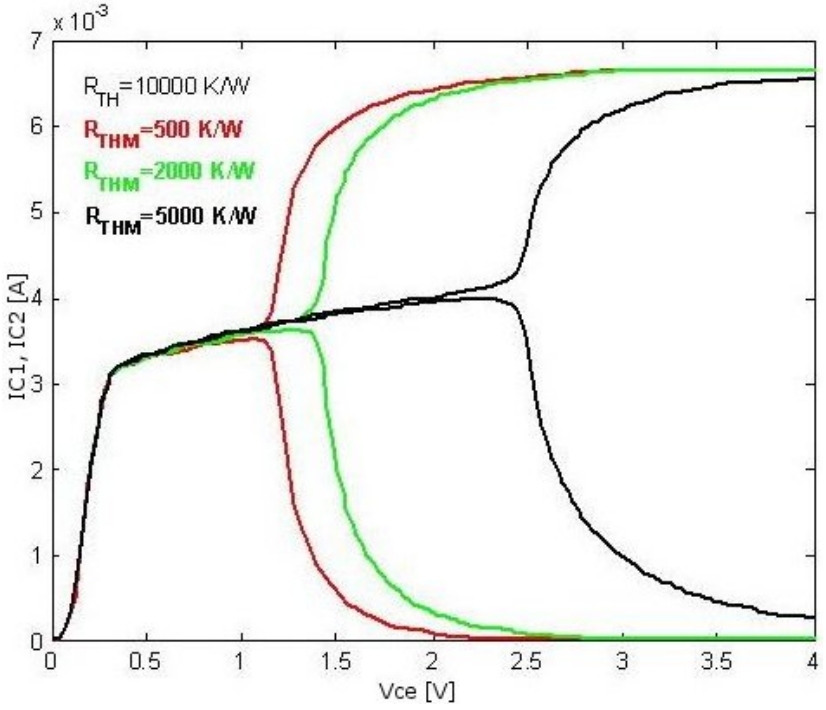


Figure 3.76: Graphical output of Simulation 1

the value of the self-heating thermal resistance R_{TH} of both fingers is assigned (i.e., 10000 K/W and analysed the effect of varying the mutual thermal resistance R_{THM}. Fig. 3.76 details the individual currents of both fingers vs. the collector-to-emitter voltage V_{CE}. As can be seen the bifurcation phenomenon occurs at larger power levels when the thermal coupling (i.e., the R_{THM} value) is higher [56], [20].

Simulation 2 — Case of three paralleled BJTs

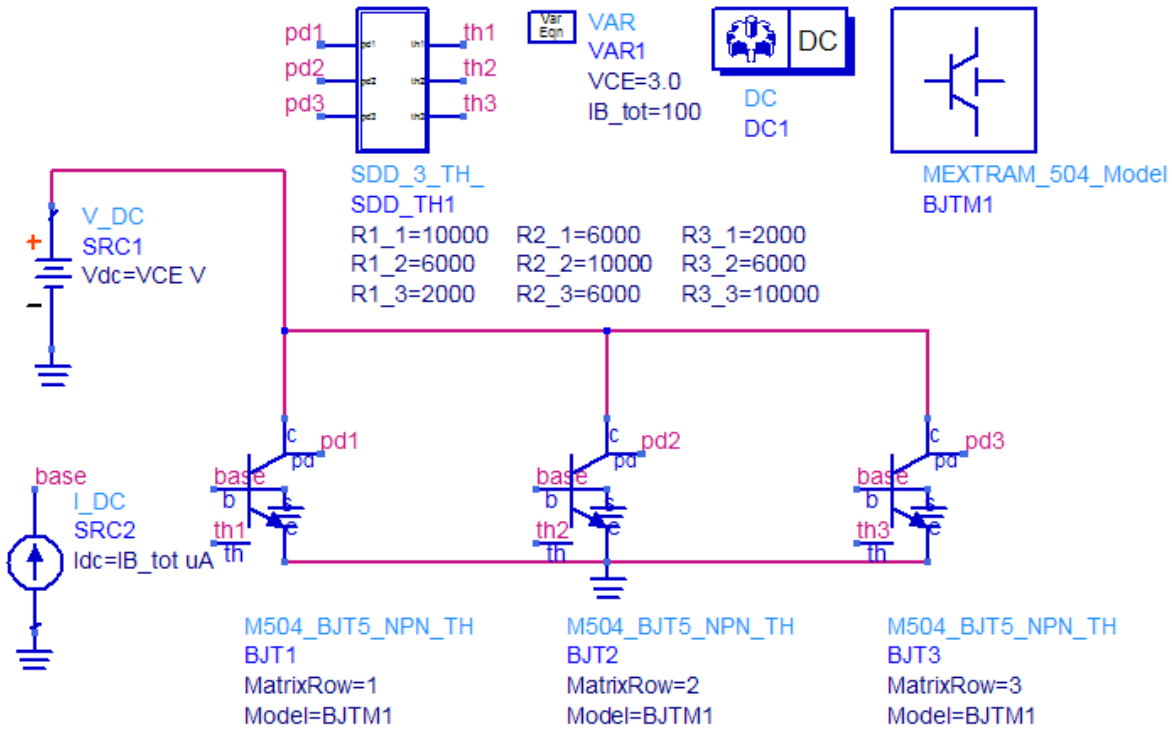


Figure 3.77: Schematic to Simulation 2.

Lets consider now the more complex case of three parallelled BJTs (or, equivalently, a 3-finger device). The thermal resistance matrix incorporated in the Verilog-A component has been inserted "directly" (and not calculated from the layout); again, the values considered are typical of SOG transistors. In this case, no "electrical" discrepancies have been introduced between transistors, that is, all fingers are assumed ideally identical[7].
The aim of this analysis is to evaluate the effect of the spacing between fingers on the electro-thermal behavior of the 3-finger device. The first thermal resistance matrix accounted for describes the case of a "poor" thermal coupling between transistors (that is, a high spacing). The matrix is in table on page 102. All simulations below have been

[7]In the following, it will be clarified that this is a reasonable assumption for an odd number of uniformly spaced devices.

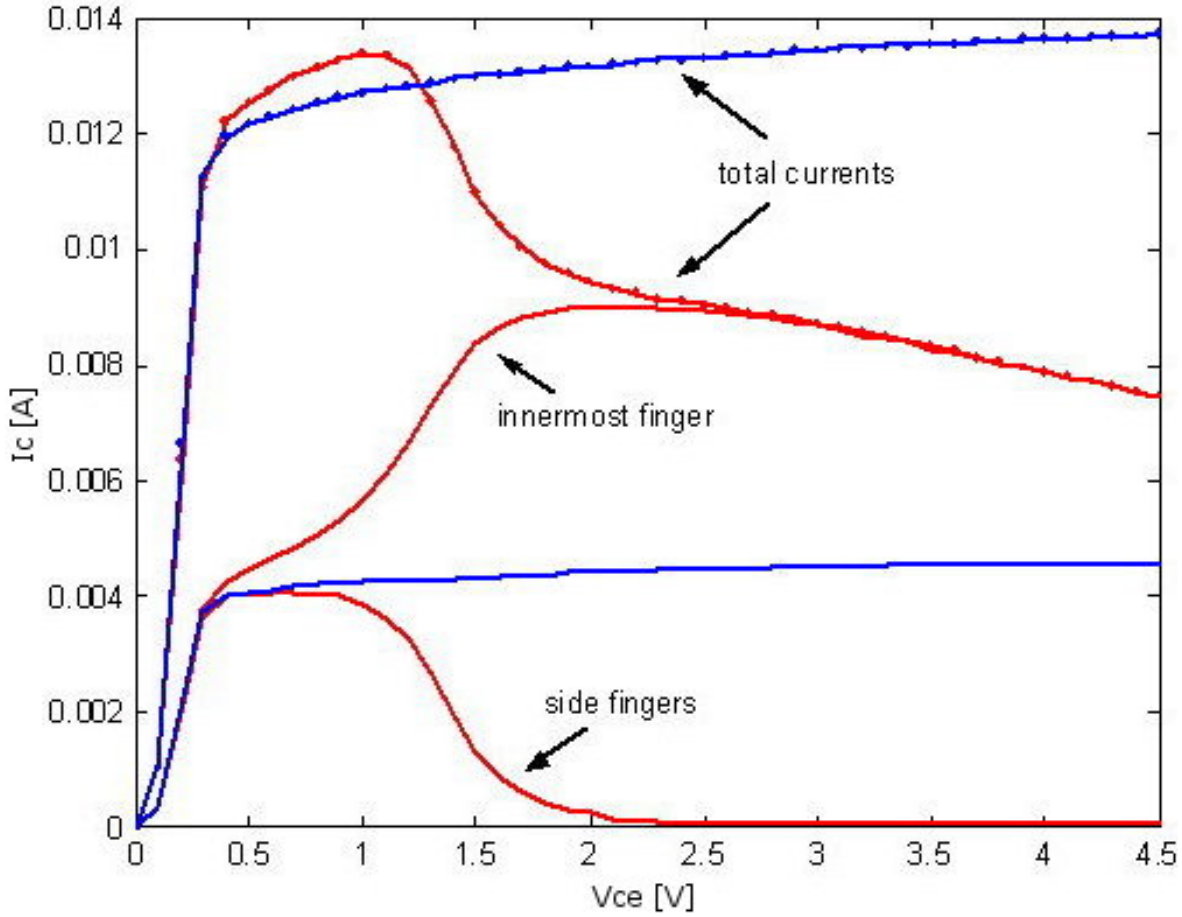


Figure 3.78: Results of Simulation 2 (1).

Finger	1	2	3
1	10000 K/W	2000 K/W	500 K/W
2	2000 K/W	1000 K/W	2000 K/W
3	500 K/W	2000 K/W	10000 K/W

Table 3.1: Values (1) of thermal resistances for Simulation 2.

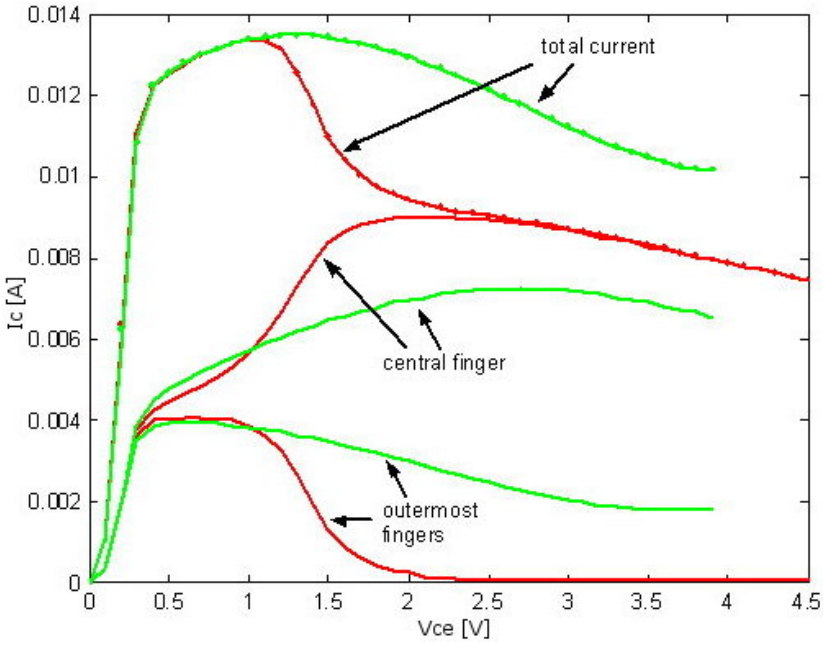


Figure 3.79: Results of Simulation 2 (2).

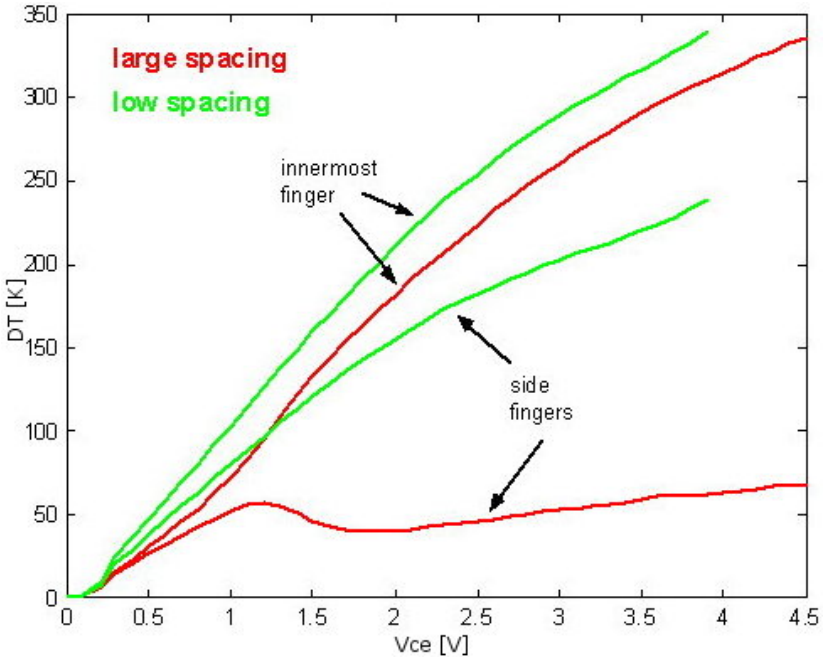


Figure 3.80: Results of Simulation 2 (3).

performed by applying a total base current I_{BTOT}= 100μA. Fig. 3.78 reports the collector currents of the individual fingers vs. V_{CE} (solid lines), along with the ttal collector current I_{CTOT} = 100μA (solid lines with dots). Red curves have been evaluated from the "electro-thermal" schematic, whereas blue curves refer to the "standard" schematic (i.e., the one without electrothermal effects). As can be seen, a purely electrical circuit simulator would predict the same behavior for all fingers (which would handle an identical current amounting 4 mA. Conversely, the electrothermal ADS-based tool allows evidencing that the innermost finger suddenly starts conducting more current than the side ones (which handle the same amount of current due to the inherent system symmetry); for V_{CE} values larger than 2V, the outermost fingers are dry and the central fingers bears all the current. This translates into a total collector current much lower (8mA @ V_{CE}=4V) than the one that would be conducted under isothermal conditions at ambient temperature (13 mA @ V_{CE}=4V), that is, electrothermal effects strongly limit the current handling capability of this 3-finger device.

The above analysis demonstrates that the adoption of a commercially available circuit simulator that does not enable electro-thermal effects might lead to significantly inaccurate results, especially when the thermal resistances are high, like in new-generation SOI or SOG structures.

Inspection of Fig. 3.78 allows demonstrating that, for biasing conditions here the current distribution over the 3-finger device is still almost uniform (i.e., V_{CE}<1V), the total current I_{CTOT} increases with V_{CE}. This is the obvious consequence of the PTC (Positive Temperature Coefficient) of the current gain in silicon bipolar transistors [58] due to the band-gap narrowing in highly-doped emitters [59]. In the following, it will be shown that the "analogous" 3-finger HBTs exhibit an opposite behavior in this "thermally stable" region. Conversely, when the current (and temperature) distribution becomes uneven ("thermally unstable" region), the total current "collapses" with respect to the T=300K case (as experimentally verified in [60]), similarly to what happens in GaAs-based HBTs, thus confining the current handling capability of the device well below the purely electrical limitations.

The aim of this analysis is to evaluate the effect of the spacing between fingers on the electro-thermal behavior of the 3-finger device. The first thermal resistance matrix accounted for describes the case of a "poor" thermal coupling between transistors (that is, a high spacing). The matrix is given by R_{THij} values. Again, the thermal resistance matrix is provided "directly" to the code, and is given in table on page 105. Fig. 3.79 details the comparison between the case of reduced spacing (i.e., more "thermally coupled" fingers, green curves) and the previous case of large spacing (red characteristics). Again, due to the heat flow coming from the side fingers, the central transistor begins bearing more current than that handled by the others. It has to be noted that for V_{CE} values within the range 0.5-1V this effect is even enhanced with respect to the case of larger spacing. However, due to the larger thermal coupling in the analysed structure, the central finger does not conduct the overall current and the side fingers do not become dry at higher V_{CE} values (see green curves). Indeed the side fingers strongly suffer from the heat flow coming from the center. Hence, when the

Finger	1	2	3
1	10000 K/W	6000 K/W	2000 K/W
2	6000 K/W	10000 K/W	6000 K/W
3	2000 K/W	6000 K/W	10000 K/W

Table 3.2: Values (2) of thermal reistances for Simulation 2.

spacing is reduced, the 3-finger device exhibits a more evenly-distributed current (and temperature) and is less prone to thermally-induced current crowding phenomena. As a consequence, the overall collector current (green solid line with dots) does not "collapse" like in the "strongly uncoupled" structure above, that is, the 3-finger device is not much "confined" below its electrical boundaries.

Fig. 3.79 shows the calculated temperature increases above ambient corresponding to Fig. 3.80 As can be seen, in the "low spacing" case, the temperature distribution is almost uniform, whilst in the more "thermally uncoupled" structure, a strongly uneven temperature distribution arises for V_{CE}>1V. It has to be noted, however that, in the "low spacing" device, the temperature peak in the structure is higher than the "high spacing" counterpart at the same biasing point. This is the obvious consequence of the close proximity between fingers.

Simulation 3 — Case of two paralleled HBTs: spacing effect

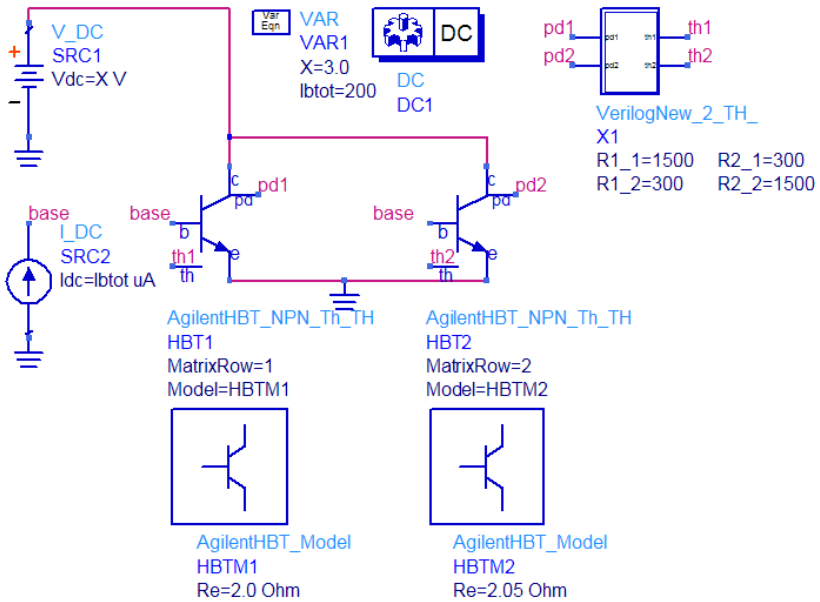


Figure 3.81: Schematic to Simulation 3

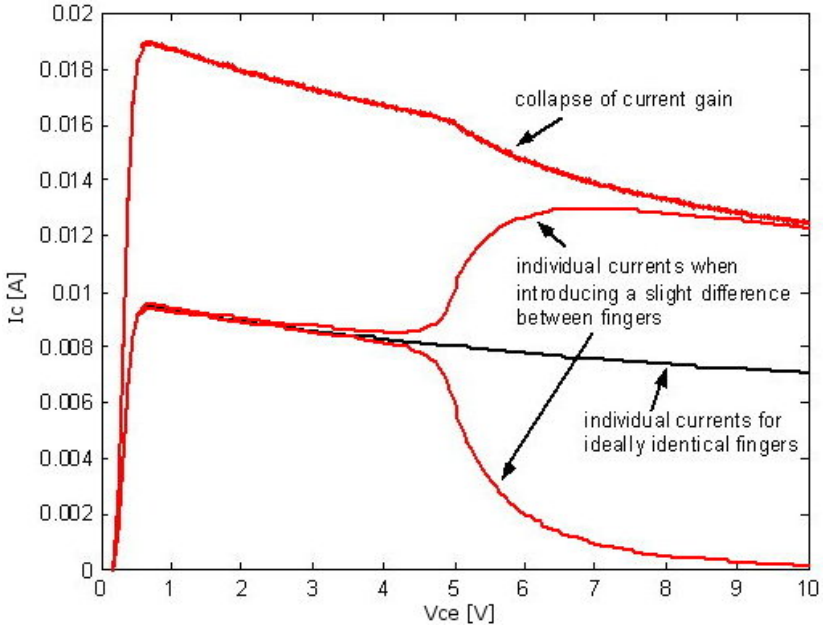


Figure 3.82: Results of Simulation 3.

The case of two parallelled GaAs-based HBTs has been lucidly discussed in [61]. In this section, is considered a 2-finger HBT operated in common-emitter configuration. The matrix of thermal resistances inserted is inserted "directly", values shown in table 3.3 on page 106. The device has been biased with a total base current given

Table 3.3: Values of thermal resistances for Simulation 3.

Finger	1	2
1	1500 K/W	300 K/W
2	300 K/W	1500 K/W

by I_{BTOT}= 200μA. Again, considering ideally identical fingers would lead to a collector current exactly shared between them. Since such a situation does not exist in practice, again a small discrepancy is forced between the "internal" parasitic emitter resistances (0.05 Ω), in order to be as close as possible to reality. Fig. 3.82 depicts the simulation results. Due to the imposed "unbalancing" condition, a "bifurcation" phenomenon occurs at a certain bias point, that is, and one finger starts suddenly bearing more current. As demonstrated in [61], the uneven current distribution manifests itself as a sudden I_{CTOT} "drop" (current crush).

Simulation 4 — Case of three parallelled HBTs: spacing effect

In this case, the thermal resistance matrix to be included into the Verilog-A block has been calculated at the pre-processing stage from a) the layout file, which contains

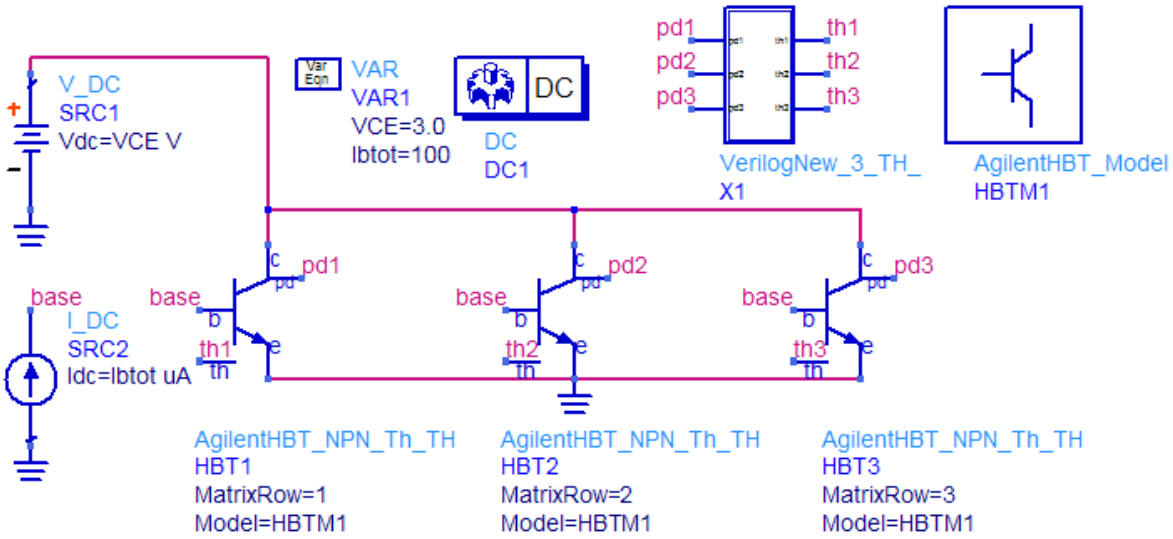


Figure 3.83: Schematic to Simulation 4.

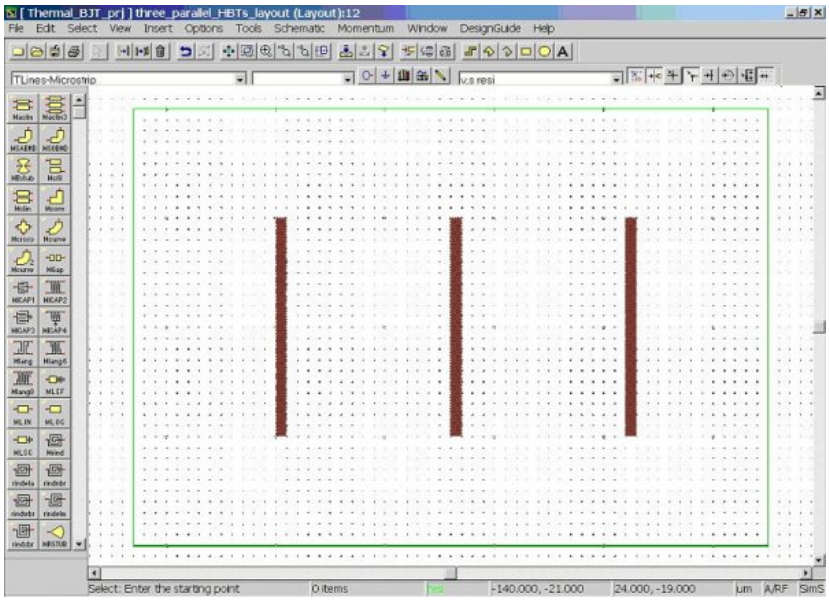


Figure 3.84: Layout for simulation 4.

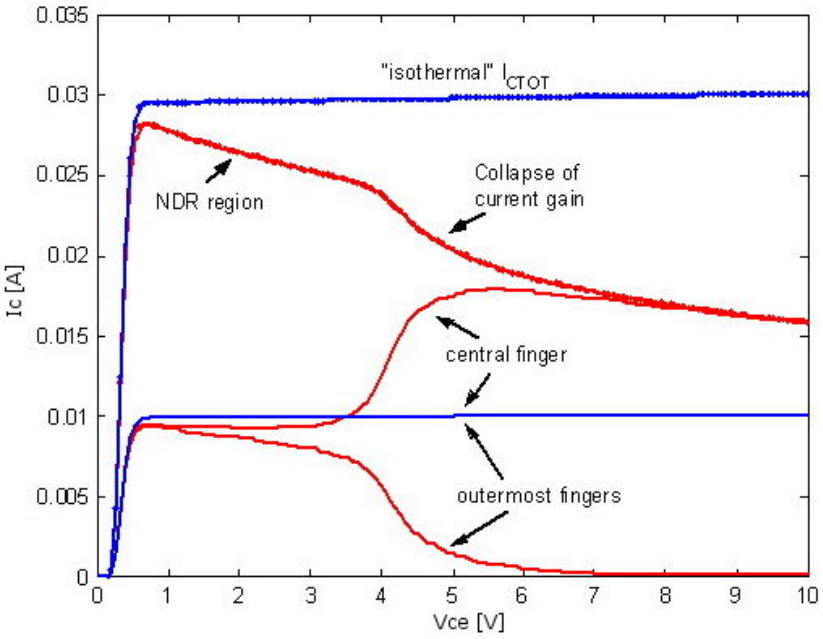


Figure 3.85: Simulation 4.(1).

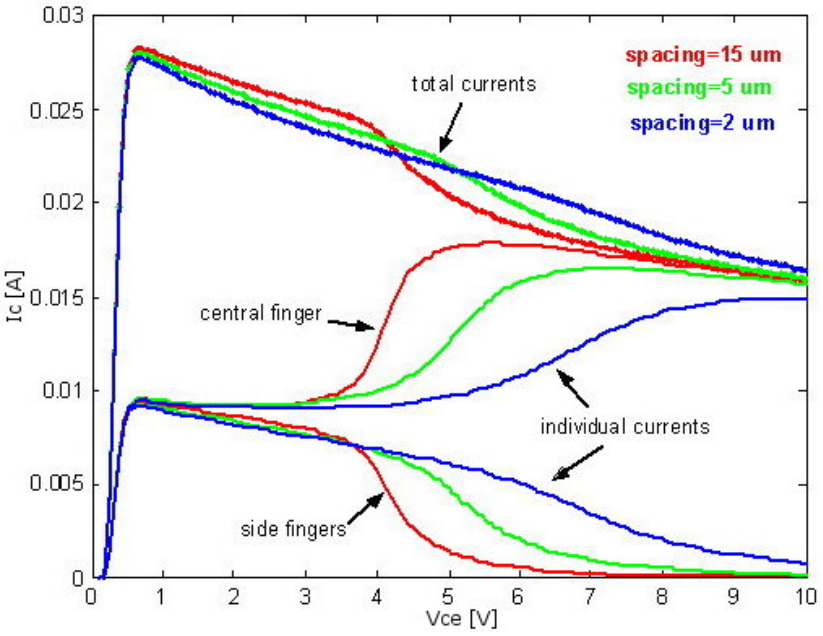


Figure 3.86: Simulation 4 (2).

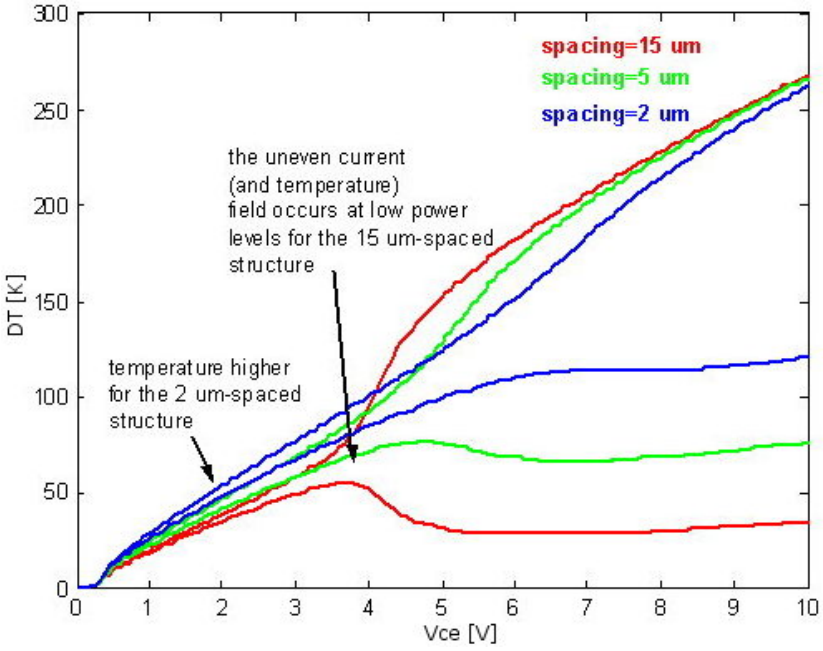


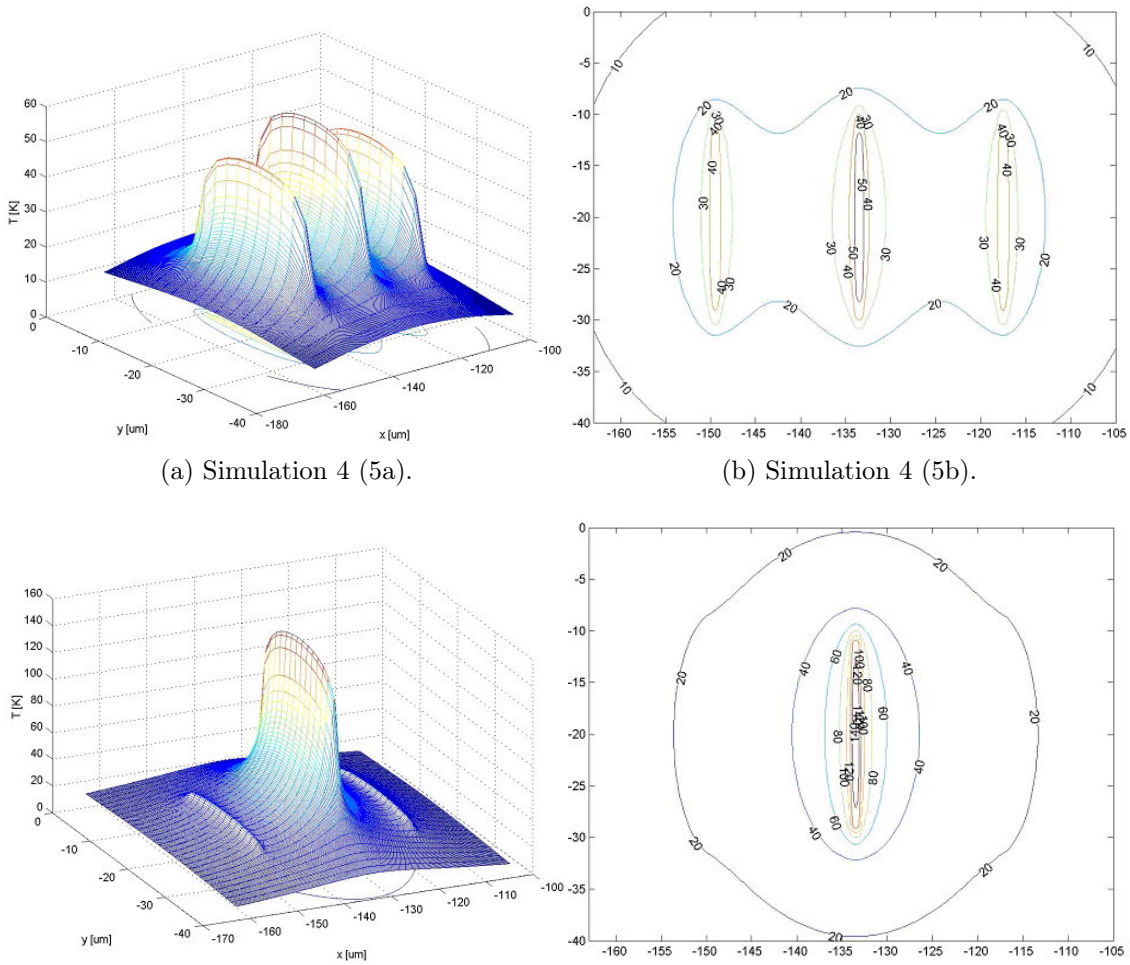
Figure 3.87: Simulation 4 (3).

geometric data about the emitter windows (x and y coordinates of the heat sources) and b) information provided by the user about the depth and thickness of the heat sources and the thermal properties of the medium. As mentioned before, the closed-form analytical expressions proposed in [35] have been exploited to this purpose. The typical value of $0.44 \cdot 10^{-4}\mathrm{W}/\mu\mathrm{m} \cdot \mathrm{K}$ has been chosen for the thermal conductivity of the GaAs substrate; nonlinear thermal effects (i.e., the thermal conductivity dependence on temperature) have been neglected, although they can be easily accounted for through the Kirchhoff transform approach (as in e.g., [62]). This section analyses the spacing influence upon the electro-thermal behavior of a 3-finger structure. The layout corresponding to the schematic is shown in Fig. 3.84.

The first device under test is characterized by $1 \cdot 20\mu\mathrm{m}^2$ emitter fingers and spacing between fingers amounting 15 μm (that is, the elementary transistors are almost "thermally uncoupled"). The total base current equates 300 μA for all the results shown in this section.

Fig. 3.85 depicts the individual collector current of each finger and the overall collector current vs. the collector-to-emitter voltage V_{CE} as calculated by the proposed tool. Red curves represent the results of the electro-thermal simulation, while blue curves are the "isothermal" (T=300K)) characteristics, which would be obtained from a commercial "purely electrical" circuit simulation tool like SPICE.

The behavior is somehow analogous to the 3-finger BJT one with some discrepancies that may be explained as follows. For low V_{CE} values (that is, in the range 0.5-4V the current (and temperature) distribution is still (almost) uniform. The individual



(a) Simulation 4 (5a).

(b) Simulation 4 (5b).

(c) Simulation 4 (6a).

(d) Simulation 4 (6b).

Figure 3.88: Thermal maps for simulation 4.

currents of the three elementary transistors decrease due to the negative temperature coefficient (NTC) of the current gain in HBTs [63]. This obviously reflects upon an I_{CTOT} lowering with V_{CE} (Negative Differential Resistance region, or NDR [64]). Note such an effect is somehow opposite with respect to the homojunction transistor case (see Fig. 3.79), where the NDR region is replaced with a PDR (P=Positive) one. This is due to the different temperature coefficient of the current gain, which dominates the whole electro-thermal behavior when the current distribution is uniform ("thermally stable" region). Indeed, as mentioned before, the temperature coefficient of current gain is positive in bipolar transistors.

In conclusion, when the current (and temperature) distribution is (almost) uniform over the 3-finger device, the electro-thermal behavior depends on the device nature (NDR in HBTs and PDR in BJTs). When a "critical" biasing condition is reached, the fingers do not share the same amount of current any more ("thermally unstable" region); in the 3-finger device analysed, the central transistor starts handling more current, while the side fingers tend towards switching off. As can be seen from Fig. 3.85 the innermost transistor bears the whole current at about V_{CE}=6V. The nonuniform current field translates into a sudden slope lowering of the I_{CTOT}–V_{CE} curve (the so-called "current crush" or "collapse of current gain" [64], [65]), similarly to what happens for the 3-finger BJT (see Fig. 3.79). Hence, contrarily to the "even" region, the features of the "uneven" operating region are somewhat independent of the device nature when considering more than two fingers.

In Fig. 3.85 the characteristics at T = 300 K are also represented (blue curves). It is shown that, in the absence of electro-thermal effects, the I_{CTOT}–V_{CE} curve slightly increases due to the Early effect. Interestingly, we note that at V_{CE}=10V a standard simulator would have predicted a current (30 mA) twice the "electro-thermal" one (15 mA), which is "pulled down" (or "collapsed") due to the thermally-induced current distribution among fingers.

Before analyzing 3-finger HBT structures characterized by smaller spacing between fingers, let us study the temperature distribution concerning the structure with spacing of 15 μm (i.e. Fig. 3.85), in the NDR region V_{CE}=3V, Figs. 3.88a and 3.88b) and well inside the instability "collapse" region (V_{CE}=5V), Figs. 3.88c and 3.88d).

Figs. 3.88c and 3.88d clearly illustrates that in the "collapse" region only the innermost transistor is bearing the overall current.

Lets now analyze the effect of the spacing between fingers. To the purpose, besides the above device with spacing equal to 15 μm, we will consider two more structures with spacing of 5 and 2 μm, we will consider two more structures with spacing of 5 and 2 μm. Fig. 3.86 details the comparison between all these cases. Both the individual collector currents of the three fingers and the total collector current versus V_{CE} are reported. It is to be remarked that all fingers are assumed ideally identical, that is, no discrepancies in the model parameters have been introduced.

Inspection of Fig. 3.86 plainly evidences the stabilizing effect of a spacing reduction (which corresponds to an increase in mutual thermal resistances). As can be seen, the

uneven current distribution (and, therefore, the "collapse of current gain") occurs at higher V_{CE} values for the structures with reduced spacing. It has to be noted, that in the NDR region the current I_{CTOT} is lower for the structures with lower spacing. Indeed the "thermal coupling" between fingers is stronger, that is, every finger suffers more from the power dissipated by other fingers; this leads to a higher temperature for each finger, and, therefore, to a reduced current gain. This behavior is clearly illustrated in Fig. 3.87, which reports the temperature increases above ambient for the three "differently-spaced" structures.

Simulation 5 — Case of three paralleled HBTs: ballasting resistances

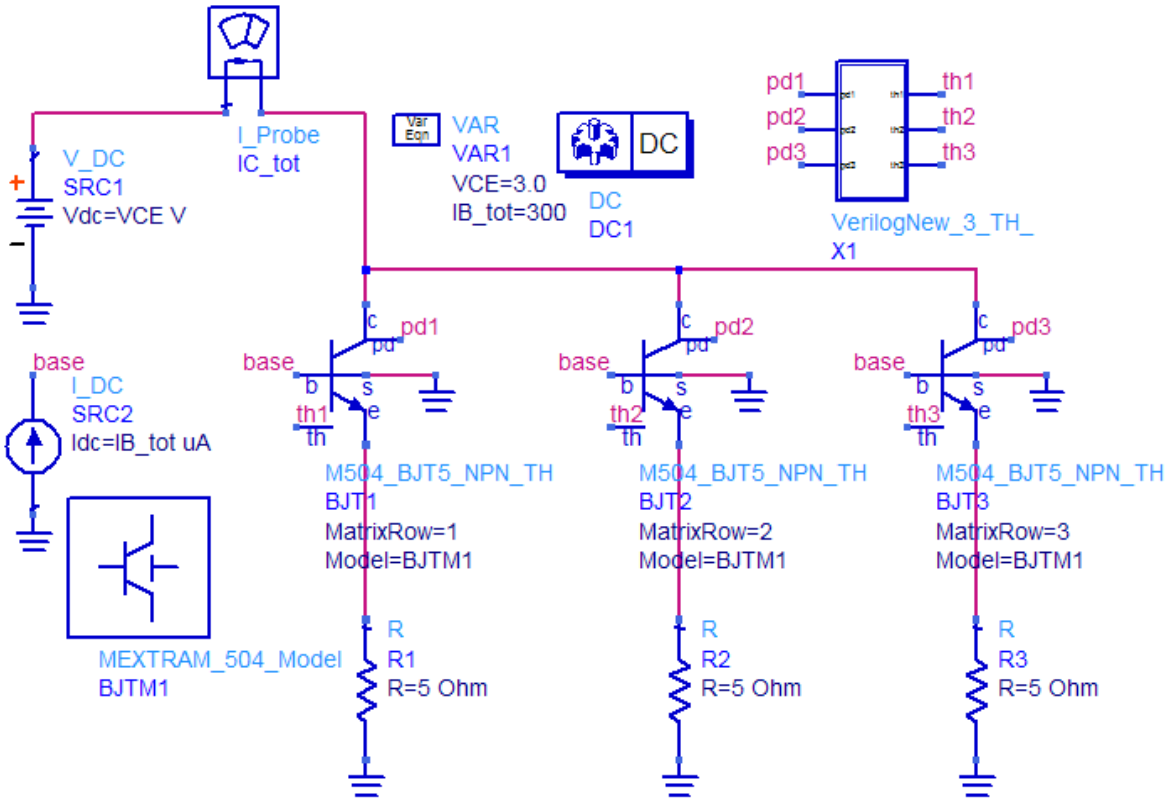


Figure 3.89: Schematic to Simulation 5.

The previous analyses clarify that the occurrence of the "current crush" due to a nonuniform current distribution in multi finger HBTs imposes a serious limitation to the power handling capability of these devices. Among other strategies, a widely diffused approach to "push" towards higher power levels the "critical" biasing conditions leading to the "collapse" is the adoption of ballasting resistors [66].

In this section, is analysed the effect of "external" ballasting resistors tied on the emitter terminals of the three fingers. In first-order analysis, the resistors will

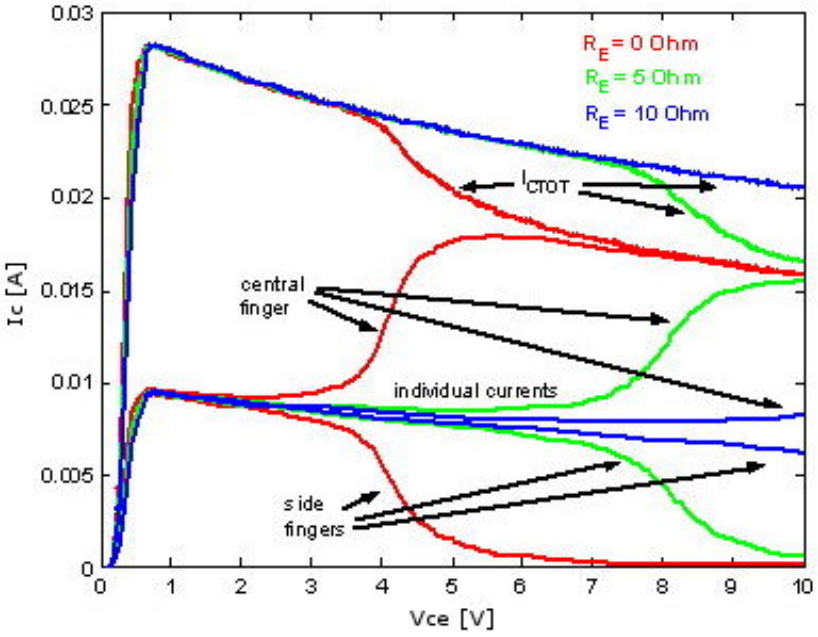


Figure 3.90: Simulation 5 (1).

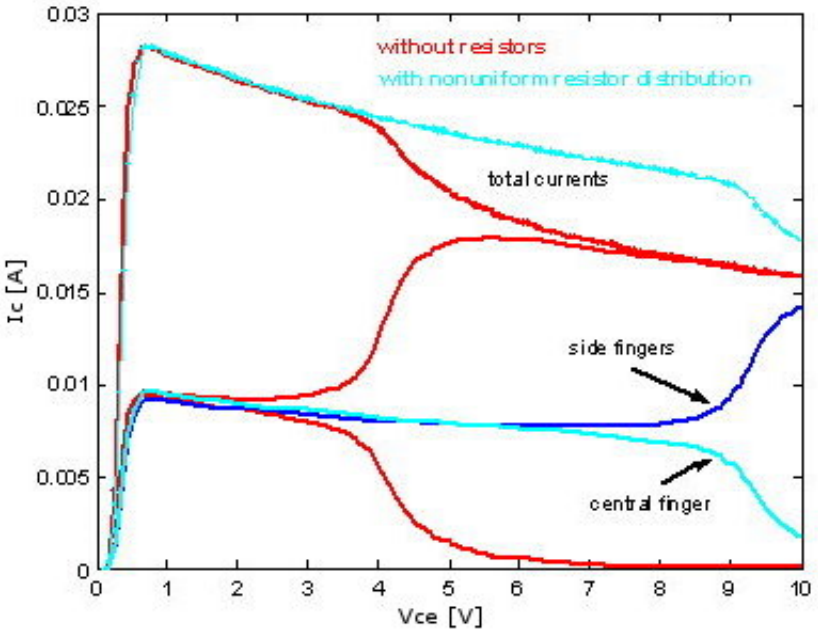


Figure 3.91: Simulation 5 (2).

be assumed as temperature-insensitive, although in practical applications they are integrated in the chip. All simulation results shown refer to an applied overall base current I_{BTOT}=300 μA. First, we consider situations where the distribution of ballasting resistors is uniform, that is, all resistors connected to the fingers are identical. The investigated cases correspond to R_E=5 and 10 Ω, respectively. In Fig. 3.90 is represented the individual currents along with the total collector currents for the case of spacing equal to 15 μm and R_E=0 (red), 5 (green) and 10 Ω (blue). As can be seen, the stabilizing effect due to the presence of ballasting resistors is apparent (the higher the R_E value, the higher the power corresponding to the "critical" biasing conditions). Of course, a noticeable increase in ballasting resistor values is undesirable, since it will limit the current densities, and, therefore, the device speed.

It should be noted that, since the heat dissipation in not uniform in a multi finger device, often the usage of a uniform R_E array may not be the "optimal" solution. As a consequence, nonuniform resistor distributions can be used in order to minimise electro-thermal effects (as in e.g., [67]). Fig. 3.91 illustrates the results corresponding to the term $R_{E,SIDE}$=6Ω (left and right), and $R_{E,CENTER}$=6.5Ω.

Simulation 6 — Case of four parallelled HBTs

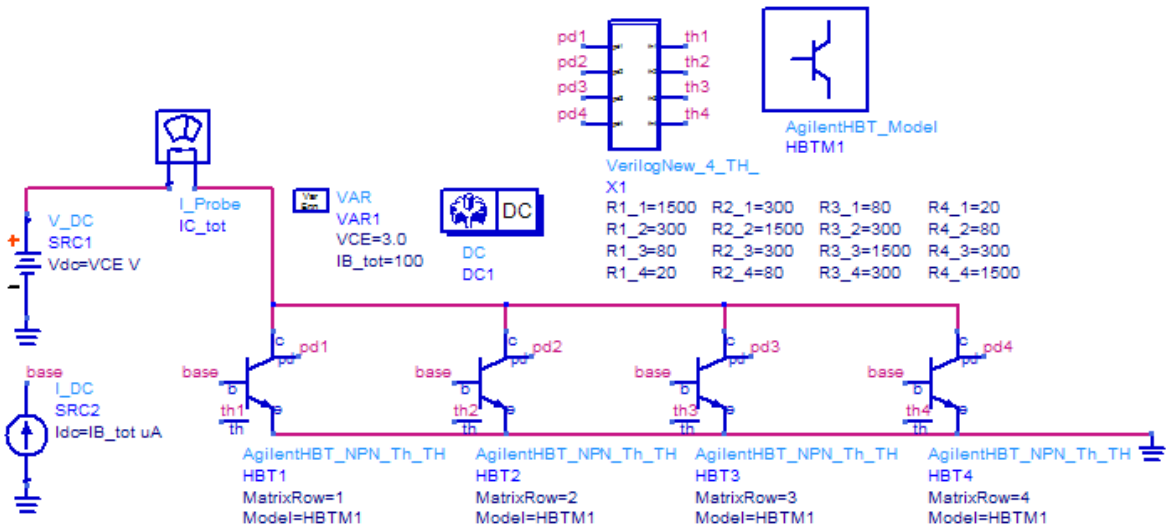


Figure 3.92: Schematic to Simulation 6.

Like in the case of a two-finger device, the assumption of perfectly identical transistors would lead to meaningless simulation results in the case of four uniformly-spaced fingers. Hence, in order to correctly describe the electro-thermal behavior of a "real" structure, is introduced a small discrepancy between the innermost fingers (namely, a difference in the parasitic emitter resistances). In this case, the matrix of thermal resistances is not calculated from information on layout and geometry of heat sources,

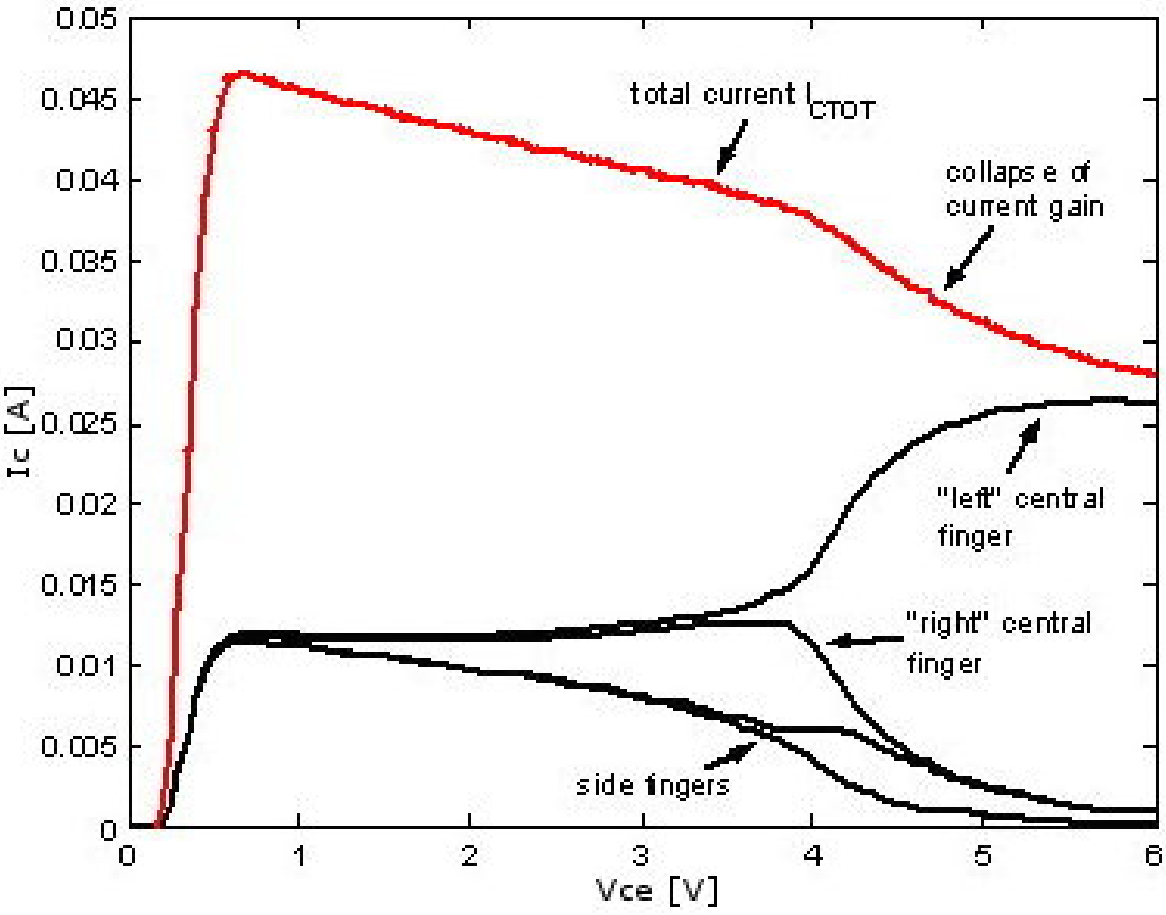


Figure 3.93: Simulation 6.

Table 3.4: Values of thermal resistances for Simulation 6.

Finger	1	2	3	4
1	1500 K/W	300 K/W	80 K/W	20 K/W
2	300 K/W	1500 K/W	300 K/W	80 K/W
3	80 K/W	300 K/W	1500 K/W	300 K/W
4	20 K/W	80 K/W	300 K/W	1500 K/W

yet the thermal resistance values has been assigned "directly". The chosen matrix is in table 3.4 page 115:

Simulation results are shown in Fig. 3.93. Black curves represent the individual collector current of each finger, while red characteristic corresponds to the total collector current I_{CTOT} handled by the 4-finger device. As can be seen, since the innermost fingers suffer more from the heat flowing from other fingers, they start conducting more current than the outermost counterparts for V_{CE}>2V. As V_{CE} is in the range 2V-3V, the central fingers handle the same amount of current (like the "external" fingers). However, when the "critical" condition V_{CE}=3.5V is reached, a bifurcation concerning the "internal" fingers occurs due to the "unbalancing" - and unavoidable in reality - discrepancy imposed. For higher V_{CE} values, it can be seen that only one finger tends to bear most of the current, while the other "central" finger behaves like a side one. In conclusion, it is clear that, even if the innermost fingers are perfectly symmetrical with respect to the barycenter of the structure, in real cases they do not conduct the same current.

Simulation 7 — Case of five parallelled HBTs

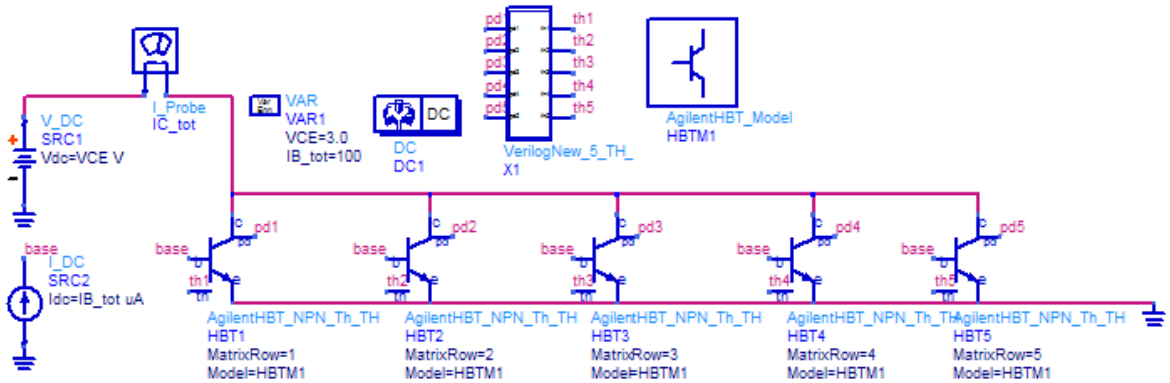


Figure 3.94: Schematic to Simulation 7.

Lets consider the case of a 5-finger device. In this case, the thermal resistance matrix has been evaluated at the pre-processing stage through the formulae proposed in [35]. A total base current amounting 1 mA has been applied. The spacing between fingers is 15 μm, that is, the elementary transistors are almost thermally uncoupled. Fig. 3.95 shows the ADS results: the individual collector currents (green, blue, and black) are depicted along with the total current (solid red with dots). Once again, the typical behavior of multi finger HBTs is detected. When the currents are still evenly-distributed, I_{CTOT} slightly lowers with V_{CE} (NDR region). At V_{CE}=3V, the electrothermal interactions give rise to a nonuniform current (and temperature) distribution among fingers, which reflects on the I_{CTOT} crush. For V_{CE}=4V, the central finger bears the whole current, and the 5-finger device is noticeably thermally limited.

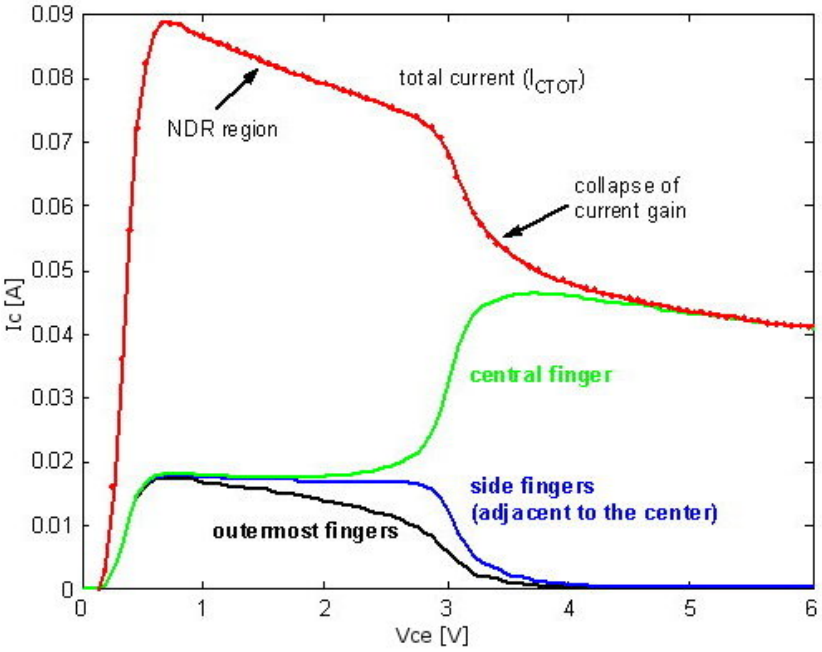


Figure 3.95: Simulation 7 (1).

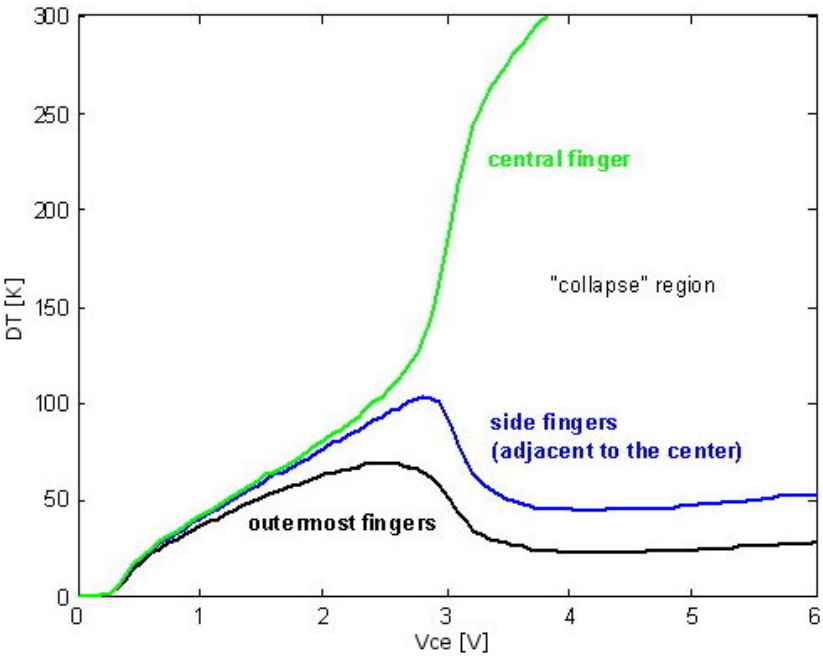
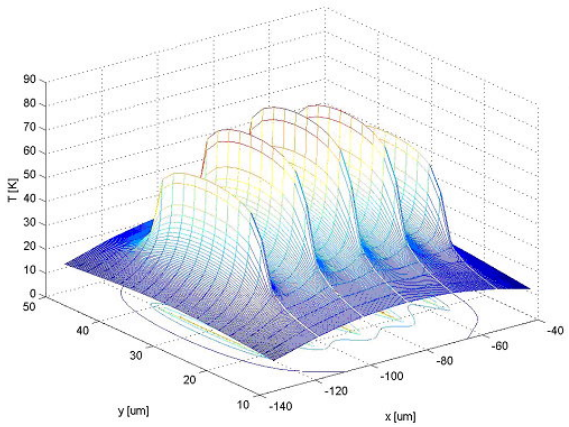
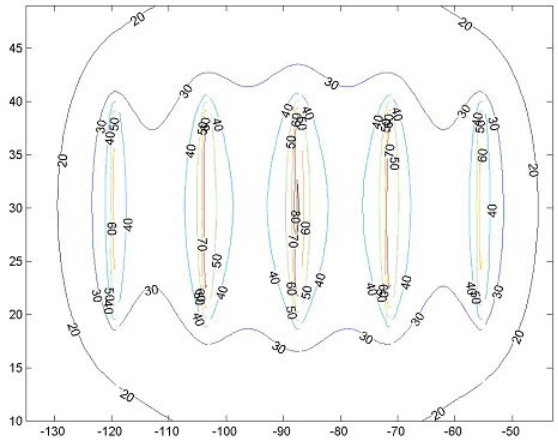


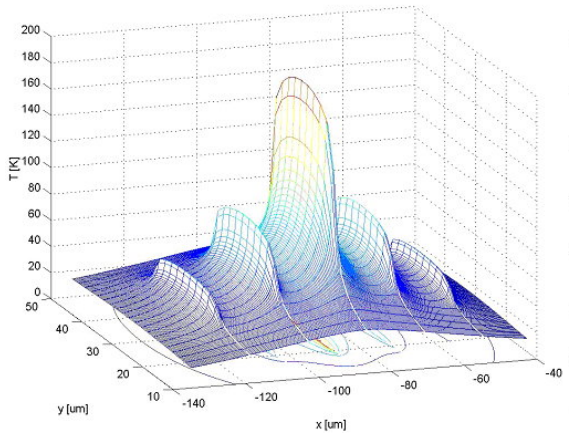
Figure 3.96: Simulation 7 (2).



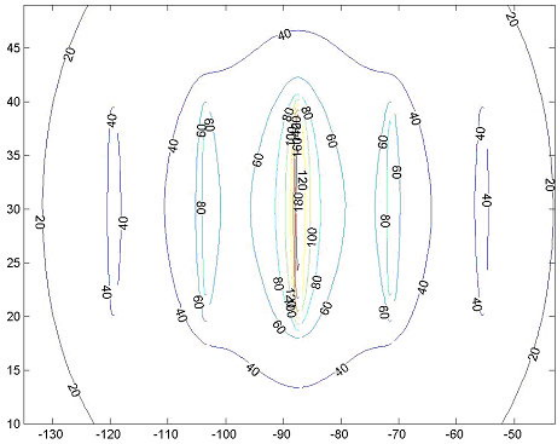
(a) Simulation 7 (3).



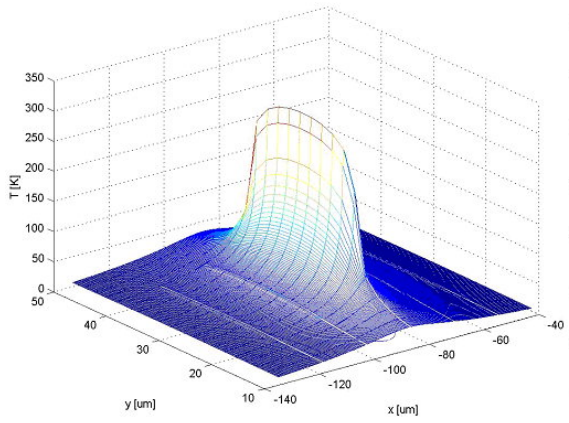
(b) Simulation 7 (4).



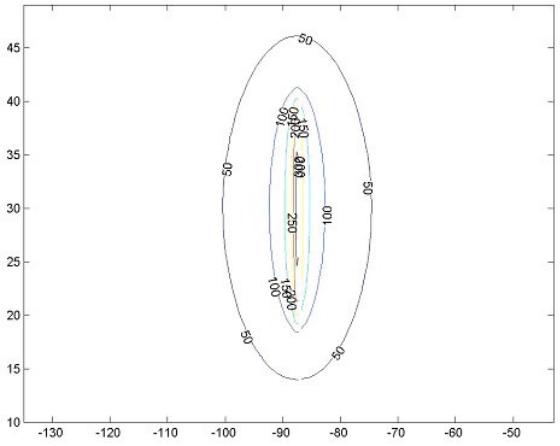
(c) Simulation 7 (5).



(d) Simulation 7 (6).



(e) Simulation 7 (7).



(f) Simulation 7 (8).

Figure 3.97: Thermal maps for Simulation 7.

It has to be noted that, since all fingers are assumed identical in the simulation, the currents of the two fingers adjacent to the center (blue curves) are identical due to the system inherent symmetry; analogously, the currents handled by the two outermost fingers (black curves) are equal.

Important is, that introducing small differences between transistors do not play an important role when considering an odd number of fingers with uniform spacing; indeed it can be demonstrated that, in this case, the behavior is dominated by the thermal interactions more than by the electrical discrepancies. Conversely, when accounting for an even number of equally-spaced elementary transistors, the overall behavior is significantly influenced by small electrical unbalancing conditions (see e.g., Fig. 3.82 (for a 2-finger device) and Fig. 3.93 (for a 4-finger device)).

For the sake of completeness, in Fig. 3.96 is shown the temperature increases above ambient corresponding to Fig. 3.95.

In order to plainly illustrate that the collapse of current gain is a phenomenon related to the nonuniform temperature distribution over the device, shows the temperature maps as obtained by the in-house post processor. Figs. 3.97a and 3.97b refer to the case V_{CE}=2V (NDR region). As can be seen, all transistors are still conducting (almost) the same amount of current, and the overall behavior is dominated by the NTC of the current gain. Figs. 3.97c and 3.97d represent the case V_{CE}=3V, which is the onset of the thermal instability, namely the occurrence of a thermally-induced uneven current distribution, which basically coincides with the I_{CTOT} crush. The case V_{CE}=4V is illustrated in Figs. 3.97e and 3.97f: the lateral devices are off due to the electro-thermal interactions, and only the central device is conducting the current I_{CTOT}, which - as previously clarified - is usually much lower than the total isothermal current at T=300K.

Simulation 8 — Case of fifteen parallelled HBTs

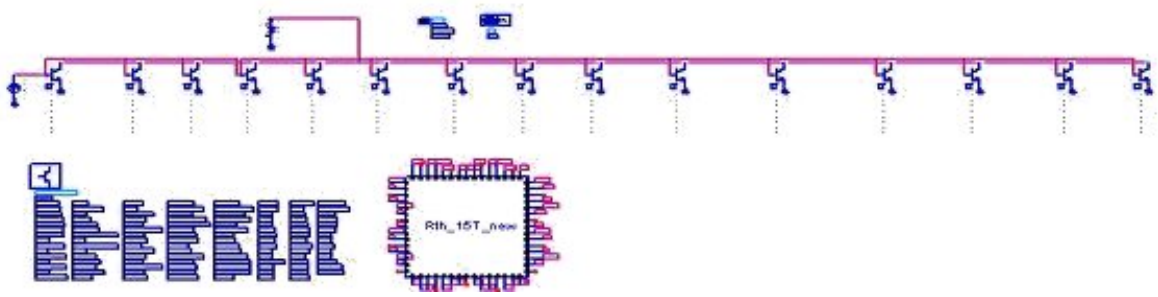


Figure 3.98: Simulation 8 (1)

Finally, is considered the more complex case of a 15 μm spaced 15 finger device. The ADS electro-thermal schematic is represented in Fig. 3.98. The dedicated Verilog-A block acting as the electro-thermal feedback component is clearly visible. The simula-

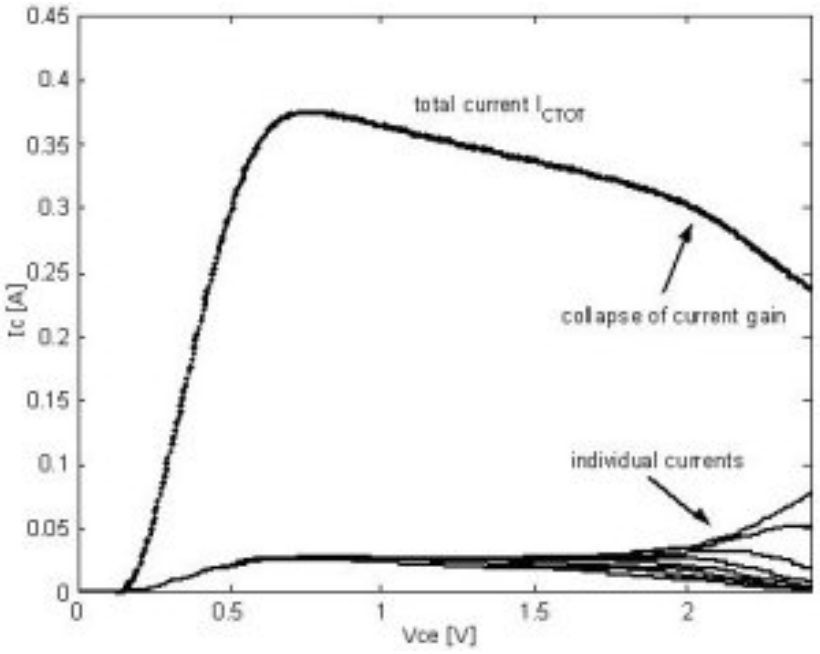
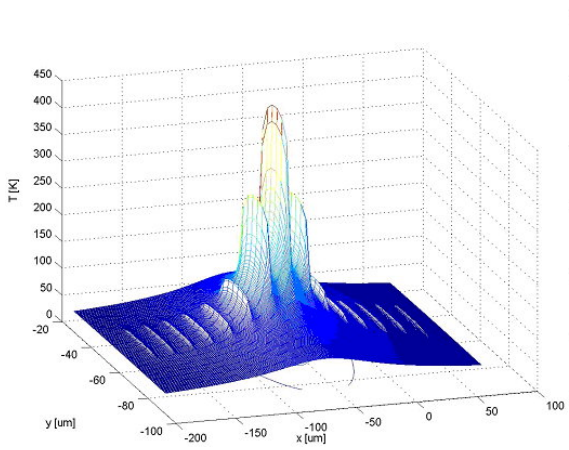
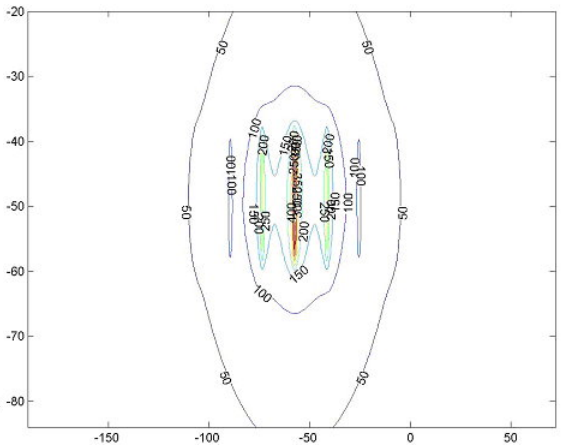


Figure 3.99: Simulation 8 (2)



(a) Simulation 8 (3).



(b) Simulation 8 (4).

Figure 3.100: Thermal maps for the simulation 8.

tion has been performed by applying I_{BTOT}=5mA. The thermal resistance matrix has been computed in the preprocessing stage.

The processing stage is quite fast: the whole electro-thermal simulation (more than 100 V_{CE} values) takes only a few seconds on a old-generation 500 MHz CPU PC. As can be seen in Fig. 3.99, the I_{CTOT} collapse occurs when V_{CE}=2V.

Figs. 3.100a and 3.100b represent the temperature maps at V_{CE}=2.5V, i.e., well beyond the instability condition. As can be seen, the central finger is handling most of the current.

Simulation 9 – Transient simulation of 1 HBT

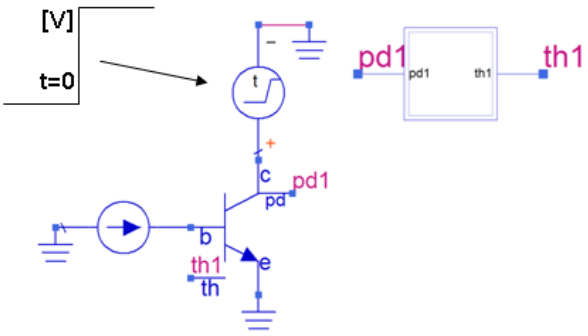


Figure 3.101: Circuit for transient simulation.

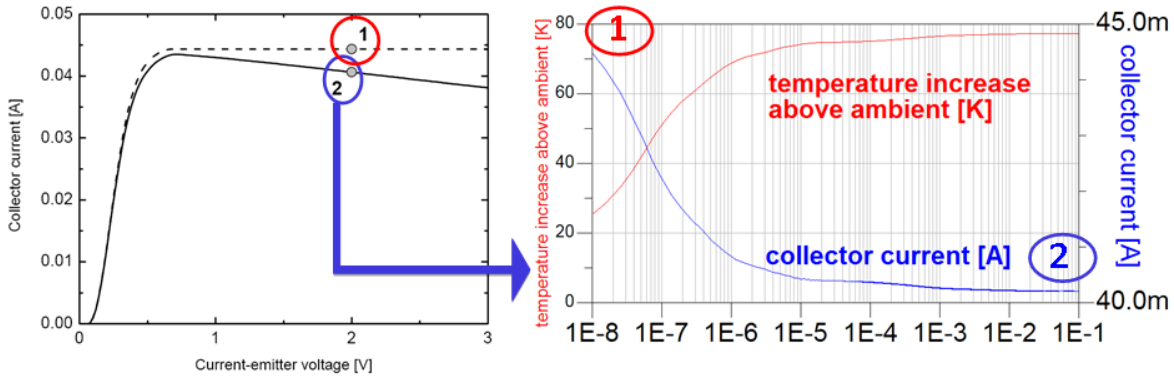


Figure 3.102: The collector current reduces with increasing temperature due to the negative temperature coefficient of the current gain.

Starting at time=0 (Fig. 3.101) a collector voltage is applied, so the device starts dissipating power, starts to heats up, so the simulation shows the I-V curves for the isothermal and non-isothermal cases. At the beginning of the simulation, the device is at an ambient temperature, and then as the device heats up, the temperature reaches

the steady-state value, which corresponds to point 2. The current gradually decreases form point 1 to point 2 and this is shown in Fig. 3.102. The transient thermal networks is created using an automated routines described in the appendix A and libraries from the Design-Kit.

APPENDIX A

Design of efficient optimisation algorithm

The issues described in this chapter are related to an automated identification of transient thermal curves.
As described in the chapter 2, the thermal impedance may be approximated using an arbitrary length RC networks of Foster or Cauer type. However the automatic identification of positive RC pairs performed by a software is a complex task. For the identification process, following assumptions were made:

1. The amount of N desired RC pairs, which approximate the curve behaviour is given as an input. The shorter RC series, the better for a simulation time for a price of accuracy.

2. Identified resistances R and capacitances C must have positive values, since the circuit simulators may not accept negative ones.

3. The thermal impedance *Zth* values (samples) can come from both measurement or simulation. The vectors must be the same length, and should be noise free.

4. Identification is made using Foster network. After an identification a translation to a Cauer network is possible using an approach in [13].

In [68], a time constants identification method for transient thermal responses has been proposed. The method is based on NID[1], and is briefly described in [69]. Shortly, the NID method is as follows:

1. A noise-free transient thermal response *Zth* must be either measured or simulated.

2. A pass from linear t to the logarithmic time domain z is done by the operation *z=ln(t)*.

[1]Network Identification by Deconvolution

3. A derivative *d(Zth)/d(z)* has to be calculated. As a result a signal is obtained, which is a convolution between τ density function, and a weighting function ω.

4. In order to obtain the τ density spectrum, the deconvolution is performed in a Fourier domain using a simple division, and finally the desired τ spectrum is obtained.

The NID approach was under investigation, in order to obtain the initial conditions for the transient response identification method, however has not been found attractive for the purpose of automated identification of transient thermal responses. The difficulties has been found as follows:

Resolution loss as described in [68] - if the time constant are too near each other, it is impossible to distinguish them.

Filtering in the Fourier domain using in order to avoid the noise enhancement due to convolution with weighting function, performed by e.g. Gauss filter. The filtering changes shape and amplitude of τ spectrum. It is important further, during integration of the spectrum. An integration is performed in order to obtain the values of resistors/capacitors for the preliminary Foster network.

Integration of the τ density function is performed in order to obtain the final Foster network, which approximates the thermal transient response. The smaller integration ranges, the longer RC Foster network is created, typically around 100 to 150 RC cells. The usual amount of RC cells which approximates the thermal response is made up to 10 cells. In fact the very long network is created for the purpose of structure functions, which later can be used to create the final RC network

Negative spectrum may occur in τ spectrum data, which are contributing to transfer (mutual) thermal impedances.

With respect to the identification criteria enumerated at the beginning of the chapter, an algorithmic implementation of NID method proposed by [68] for an automated identification of transient thermal responses has been found too complex. Additionally requires higher computational effort and more intermediate steps with respect to the proposed and developed approaches.

In the proposed approaches described in sections A.1 – A.2, a successful identification solution strongly depends on initial conditions. These methods offer a strategy for obtaining initial conditions, which enable the identification convergence.
After initial conditions are calculated, the identification is performed using the *Simplex* method [70] in one or two passes. The 2^{nd} pass uses an output from the 1^{st} pass and produces a final values of resistors and capacitors for N length RC cells. According to the experimental experience, the Simplex method performs the identification for family of thermal impedance curves correctly and with a small computational effort.

In section A.1 is described the first method based on experimental observations. Further, in section A.2, the method based on a gradient identification is proposed. The identification is performed for set of curves, created using the method described in [12], according to the parameters shown in the Tab. A.1. The curves are presented in Fig. A.1

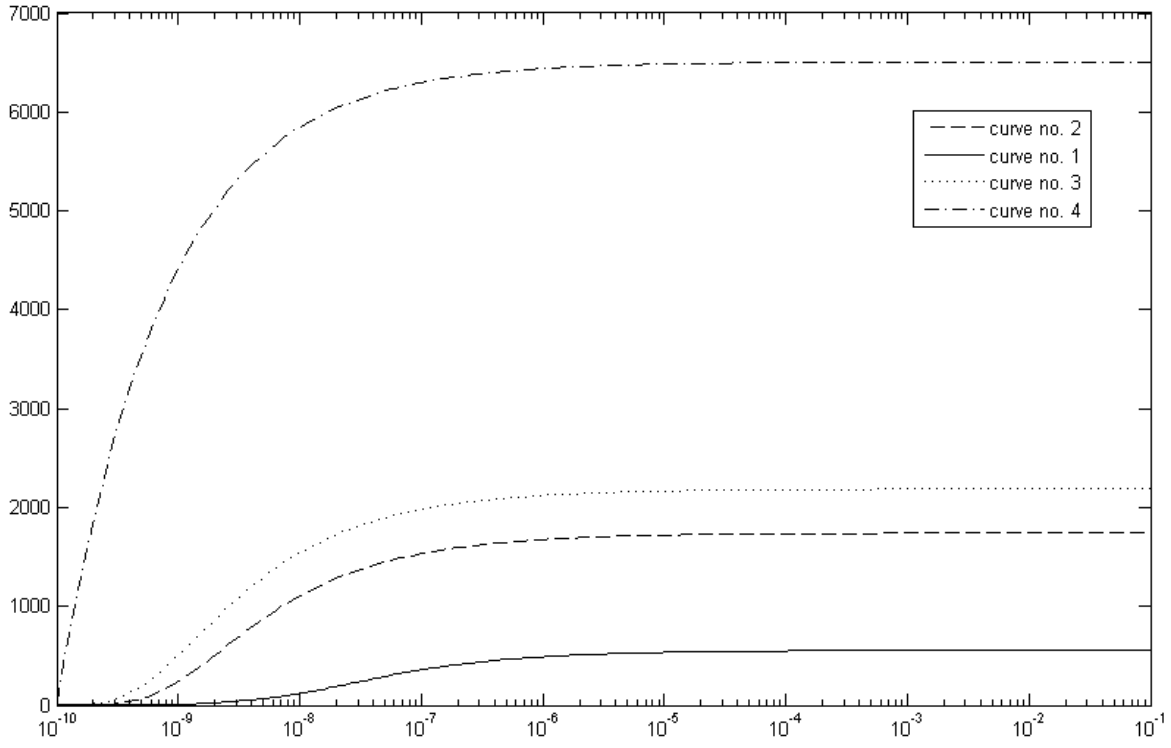


Figure A.1: Reference thermal impedance responses, created using [12] according to parameters in the Tab. A.1.

Table A.1: Parameters of reference thermal impedance curves, created using [12].

curve n°	width W [μm]	length L [μm]	P depth [μm]
1	6	6	0.5
2	1	1	0.5
3	0.1	0.1	0.5
4	0.1	0.1	0.01

A.1. Difference method

The method takes the initial conditions as follows:

Resistances initial values are taken from the final value of a thermal impedance. In a noise-free thermal impedance vector, the last value is the steady-state one. The initial condition is set dividing the steady-state value by the desired amount of resistances N.

Capacitors initial values are taken from the time vector. The time vector of length L is divided by the value of N, which is the length of the desired RC networks. As a result of division, the time vector is split into N time ranges. The first point of each range is taken as an initial time constant τ_N value. Finally, the capacitances are obtained dividing τ_N by appropriate resistance value R_N.

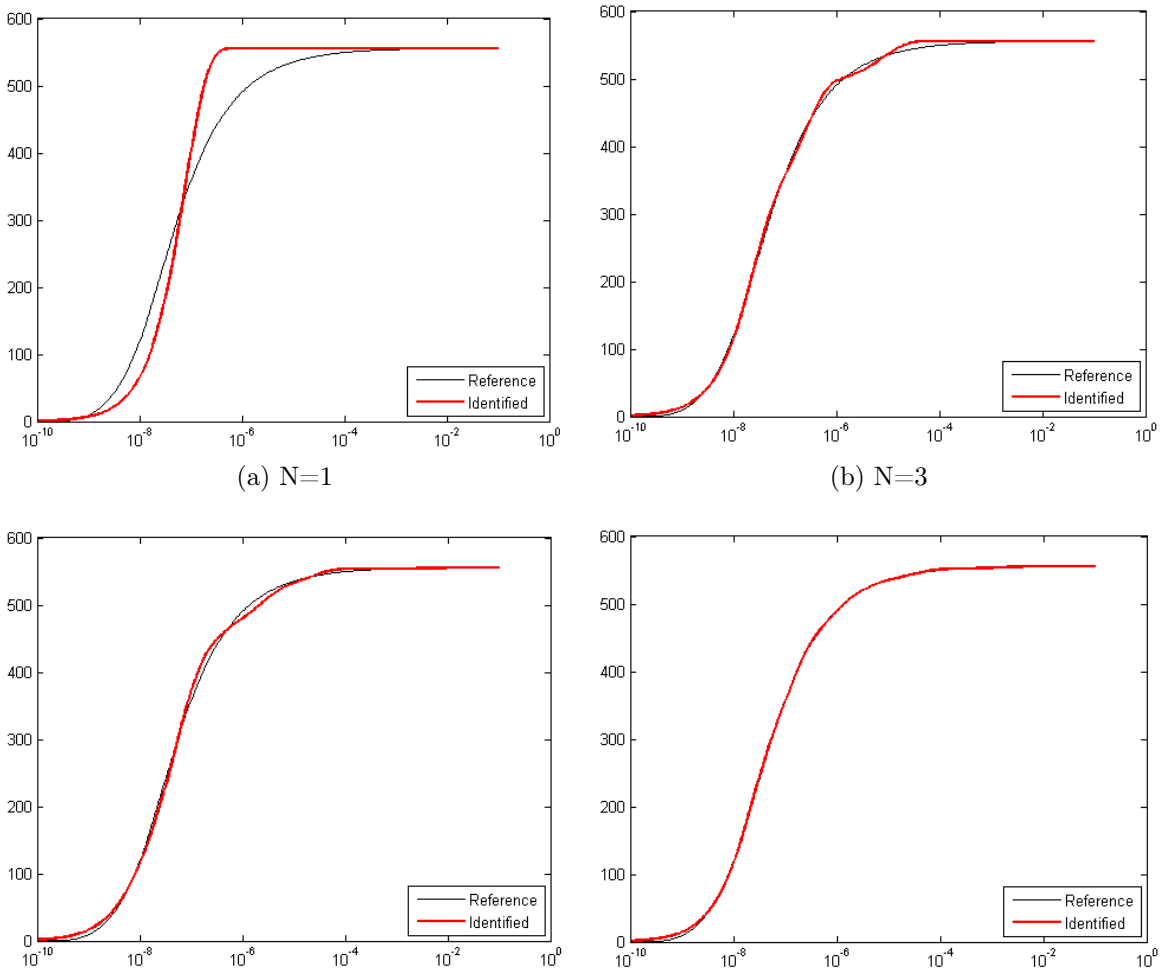
With aforementioned initial conditions for Resistors and Capacitors values, the 1st pass of Simplex method starts. Further the results from the 1st pass are used as initial conditions during the 2nd pass of the Simplex method. The 2-pass strategy reduces significantly an identification error, with respect to the one pass approach. It is clearly visible for RC series lengths higher than 5, as shown further in examples.
The identification examples of the "Difference method" are shown in Fig. A.2 for the curve n°1, and in the Tab. A.2 for all curves shown in Fig. A.1. For the clarity, differences between 1st and 2nd are shown using bold font in the Tab. A.2.

Table A.2a: Identified resistances for the curve n°1 with one Simplex pass.

N=1	N=2	N=3	N=4	N=5	N=6	N=7	N=8	N=9	N=10
556.09	379.84	290.43	240.1	230.69	234.31	306.77	256.9	240.19	220.98
-	176.26	202.82	204.1	20.25	162.74	70.6	203.77	158.32	190.24
-	-	62.84	86.27	90.51	55.68	67.7	21.65	86.59	72.47
-	-	-	25.64	29.47	70.02	33	59.42	42.76	44.99
-	-	-	-	2.95	5.5	75.92	8.86	6.9e-1	8.05e-1
-	-	-	-	-	27.83	5.95e-3	3.65	11.33	21.57
-	-	-	-	-	-	2.18	1.8	14.65	5.24e-1
-	-	-	-	-	-	-	4.29e-2	1.58	2.95
-	-	-	-	-	-	-	-	1.84e-4	7.8e-1
-	-	-	-	-	-	-	-	-	7.93e-1

As shown in the Tab. A.2, the 2nd pass can reduce the fitting error, defined by Eq. A.1

$$error = \frac{|Z_{TH\_Identified} - Z_{TH\_Reference}|}{Z_{TH\_FINAL}} \cdot 100[\%] \tag{A.1}$$



(a) N=1 (b) N=3

(c) N=7 (d) N=10

Figure A.2: Identification results for the curve n°1, with only one Simplex pass.

Table A.2b: Identified capacitances for the curve n°1 with one Simplex pass.

N=1	N=2	N=3	N=4	N=5	N=6	N=7	N=8	N=9	N=10
1.4e-10	8e-11	7.6e-11	7.7e-11	7.7e-11	7.8e-11	1.7e-10	7.6e-11	2.8e-10	7.8e-11
-	6e-9	1.2e-9	6.2e-10	5.6e-10	6.3e-10	2.8e-9	7.7e-10	8.6e-11	5.1e-10
-	-	1.4e-7	1.7e-8	1.2e-8	5.1e-9	3.1e-8	5.2e-8	7.5e-9	7.3e-9
-	-	-	2e-6	8.6e-7	1.9e-8	6.4e-7	5.6e-8	1.3e-7	5.3e-8
-	-	-	-	5e-3	2.6e-4	9.5e-11	1.9e-5	1.6e-6	7e-7
-	-	-	-	-	6.9e-7	3.6e-4	5e-5	1.7e-5	1.4e-6
-	-	-	-	-	-	4.6e-3	3e-3	2.6e-8	1.6e-5
-	-	-	-	-	-	-	3.9e-3	3e-3	5.2e-4
-	-	-	-	-	-	-	-	6.7e-3	2e-3
-	-	-	-	-	-	-	-	-	1.3e-2

Table A.2c: Relative error for N amount of RC pairs for the curve n°1. 1st and 2nd Simplex pass.

Pass	N=1	N=2	N=3	N=4	N=5	N=6	N=7	N=8	N=9	N=10
1st	108.5	32.8	13.2	7.2	**6.4**	**5.7**	**17.8**	**7.5**	**6.9**	**5.4**
2nd	108.5	32.8	13.2	7.2	**5.6**	**5.2**	**5.2**	**5.2**	**5.1**	**5.1**

Table A.2d: Relative error for N amount of RC pairs for the curve n°2. 1st and 2nd Simplex pass.

Pass	N=1	N=2	N=3	N=4	N=5	N=6	N=7	N=8	N=9	N=10
1st	97.7	31.6	21.3	**20.7**	**20.6**	**20.6**	**21.8**	**162.8**	**32.8**	**169**
2nd	97.7	31.6	21.3	**19.9**	**19.9**	**19.7**	**19.7**	**20**	**19.7**	**19.8**

Table A.2e: Relative error for N amount of RC pairs for the curve n°3. 1st and 2nd Simplex pass.

Pass	N=1	N=2	N=3	N=4	N=5	N=6	N=7	N=8	N=9	N=10
1st	97	31.8	22	**22.67**	**37**	**21.2**	**22.9**	**21.9**	**102.2**	**342.8**
2nd	97	31.8	22	**20.8**	**20.8**	**20.6**	**20.7**	**20.6**	**20.8**	**20.7**

Table A.2f: Relative error for N amount of RC pairs for the curve n°4. 1st and 2nd Simplex pass.

Pass	N=1	N=2	N=3	N=4	N=5	N=6	N=7	N=8	N=9	N=10
1st	87.5	34.4	**33.1**	**32.5**	**67**	**51.6**	**88**	**44.2**	**565.8**	**129.7**
2nd	87.5	34.4	**29**	**28.8**	**28.7**	**29**	**28.4**	**28.4**	**28.6**	**28.4**

The error obviously depends on the amount of Z_{TH} samples, since the Euclidean norm in Eq. A.1 is taken into account.

A.2. Gradient-based method

The method has following assumption for initial conditions:

Resistances initial values are extracted in the same way as in the "Difference method" described in section A.1. In particular, initial values are taken from the final value of a thermal impedance. In a noise-free thermal impedance vector, the last value is the steady-state one. The initial condition is set dividing the steady-state value by the desired amount of resistances *N*.

Capacitances initial values are extracted based on a 1st derivative of a transient thermal response in a time domain $d(Z_{th})/dt$. Further, derivative $d(Z_{th})/dt$ time vector t of size L is split into N time ranges. The N is the length of the desired RC networks. The first point of each N^{th} range is taken as an initial time constant value τ_N. Finally, the initial capacitances are obtained dividing τ_N by appropriate resistance value R_N.
In addition, the size L of derivative $d(Z_{th})/dt$ can be manipulated with parameter *gradient boundary (gb)*, which restricts it to a smaller, reduced derivative value (rdv) of size *L'*. The rdv value is obtained in the following way:

1. The maximum value of derivative $d(Z_{th})/dt$ is calculated, defined as $max = maximum(d(Z_{th})/dt)$
2. Values of derivative $d(Z_{th})/dt$ above $gb \cdot max$ are found, and are defined as rdv. The rdv vector is of the length *L'*.
3. The rdv vector is split into N ranges. The first point of each range is taken as an initial τ_N value. Initial capacitances are obtained as described above, that is by dividing τ_N by appropriate resistance value R_N

The gb parameter varies in the range of 0 to 1. If gb is set to 0, it means that the $d(Z_{th})/dt$ value is not reduced to rdv, and the whole time t range is considered. On the other side, if gb is set to 1, only max value is taken into account during the initial conditions assignment for the capacitances. The default value of gb parameter is set to 0.01. The graphical illustration of the gb parameter influence on a time range is shown in Fig. A.3.

As in the method described in section A.1, with aforementioned initial conditions for Resistors and Capacitors values, the 1st pass of Simplex method starts. Further results

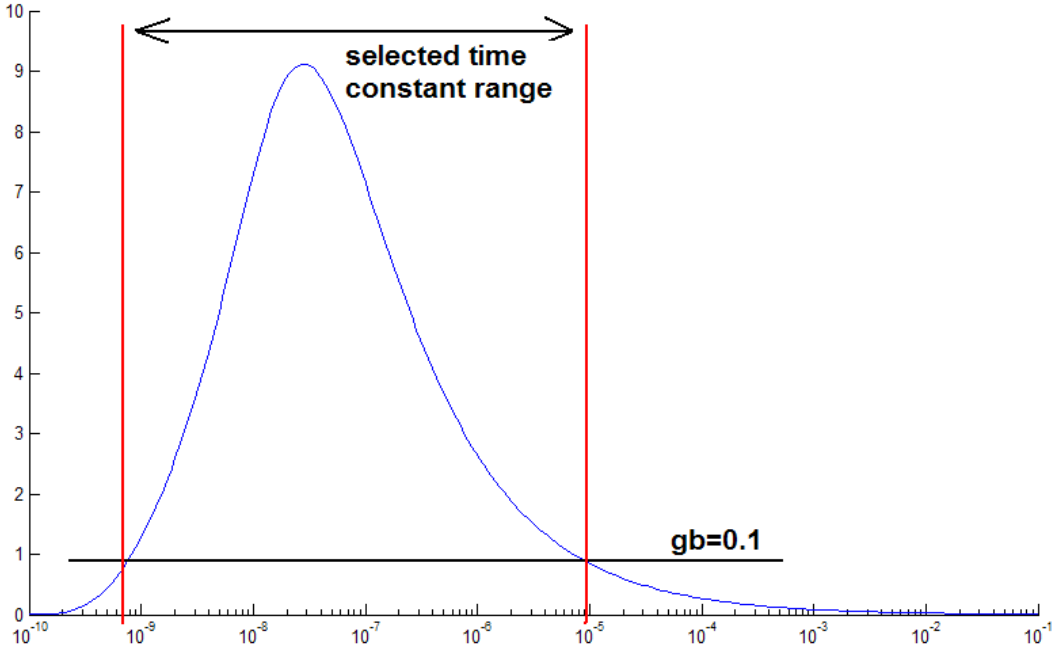


Figure A.3: Derivative of the transient thermal response in time $d(Z_{th})/dt$. The selected time range (red) is restricted by the gb parameter.

from the 1st pass are used as initial conditions during the 2nd pass of the Simplex method. The 2-pass strategy reduces significantly an identification error, with respect to the one pass approach.

The identification relative error defined by Eq. A.1 for the curves shown in Fig. A.1 and in the Tab. A.1 is shown in the Tab. A.3

Table A.3a: Relative error for N amount of RC pairs for the curve n°1. 1st and 2nd Simplex pass.

Pass	N=1	N=2	N=3	N=4	N=5	N=6	N=7	N=8	N=9	N=10
1st	108.5	32.8	13.2	13.2	**9**	**24.8**	**9.4**	**16.5**	**32.7**	**6.7**
2nd	108.5	32.8	13.2	13.2	**7.3**	**5.4**	**8**	**6.4**	**7.7**	**5.3**

Table A.3b: Relative error for N amount of RC pairs for the curve n°2. 1st and 2nd Simplex pass.

Pass	N=1	N=2	N=3	N=4	N=5	N=6	N=7	N=8	N=9	N=10
1st	97.7	31.6	31.6	21.3	**20.9**	21.3	**20.9**	20.3	**20.4**	**24**
2nd	97.7	31.6	31.6	21.3	**20.8**	21.3	**20**	20.3	**19.8**	**20.65**

Table A.3c: Relative error for N amount of RC pairs for the curve n°3. 1st and 2nd Simplex pass.

Pass	N=1	N=2	N=3	N=4	N=5	N=6	N=7	N=8	N=9	N=10
1st	97	31.8	31.8	22	22.2	**28.2**	**22.8**	**20.7**	**21.2**	**21.6**
2nd	97	31.8	31.8	22	22.2	**20.8**	**20.9**	**20.6**	**21**	**20.7**

Table A.3d: Relative error for N amount of RC pairs for the curve n°4. 1st and 2nd Simplex pass.

Pass	N=1	N=2	N=3	N=4	N=5	N=6	N=7	N=8	N=9	N=10
1st	174.6	66.1	54.5	54.5	**54**	**54.4**	**55**	**54.2**	**54**	53.6
2nd	174.6	66.1	54.5	54.5	**53.3**	**53.3**	**53.2**	**53.3**	**53.3**	53.6

A.3. Both method Comparison

The curves used in sections A.1 and A.2 do not encounter rapid increase of thermal impedance, like presented in Fig. A.4, so called "stairs". "Stairs" may happen when

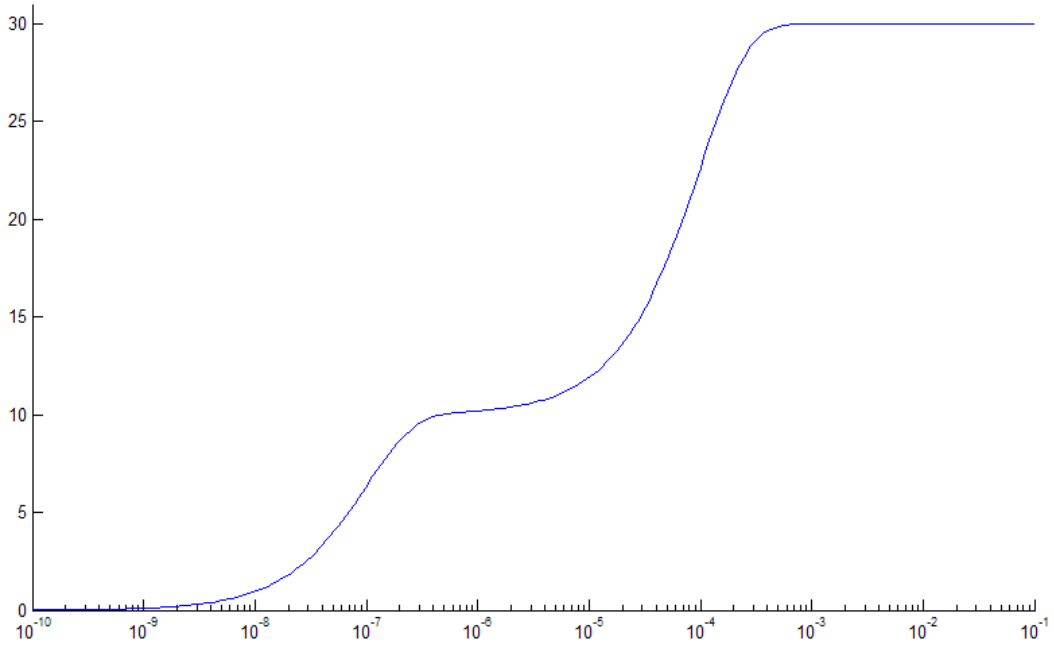


Figure A.4: Possible influence of the package

a transient thermal response is characterised for a device with many layers (many heat flow paths), as described in the chapter 2, where material/package thermal time

constant may have distant values.
The identification algorithm has was developed in order to deal with arbitrary curves. To show its capability the test transient thermal response has been created using Eq. A.2, which is as follows:

$$Z_{TH}(t) = 10 \cdot (1 - exp(\frac{-t}{1e-7})) + 20 \cdot (1 - exp(\frac{-t}{1e-4})); \tag{A.2}$$

As visible time constants are distant between each other ($\tau_1 = 1e-7$, $\tau_2 = 1e-4$). The curve has been already demonstrated in Fig. A.4. The identification results are presented in the Tab. A.4.

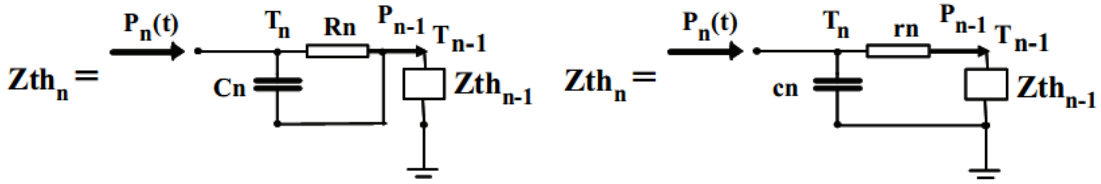
Table A.4: Comparison between identification methods for the curve in Fig. A.4 with distant thermal time constants.

N (RC cells)	Difference method	Gradient method
1	232.8	232.8
2	3.4e-14	3.4e-14
3	3.3e-12	**2.3e-12**
4	3.1e-12	**7.7e-13**
5	3.6e-10	**3.4e-13**
6	1.5e-6	**1.5e-11**
7	7.6e-6	**9.9e-11**
8	0.6	**5.7e-12**
9	0.01	**2.1e-12**
10	0.08	**2.7e-6**

Conclusions The identification approaches proposed in this chapter, successfully can identify an arbitrary thermal transient response, with a desired amount of RC pairs N. Produced values for resistors and capacitors are always positive. Both identification methods use Simplex algorithm proposed by [70] in two passes, that is the output from the 1st pass is an input for the 2nd.

A.4. Cauer network transformation

The transformation between Foster and Cauer network is performed using the algorithm described in [13]. The impedance of Foster network of length N is described by the recursive Eq. A.3.



(a) Recursive Foster. (b) Recursive Cauer.

Figure A.5: Description of thermal networks. Source [13].

The algorithm is as follows:

$$ZthF_N(s) = \frac{1}{s \cdot C_N + 1/R_N} + ZthF_{N-1}(s) \tag{A.3}$$

The equivalent Cauer network of length N is described by the recursive Eq. A.4

$$ZthC_N(s) = \frac{1}{s \cdot c_N + \frac{1}{r_N + ZthC_{N-1}(s)}} \tag{A.4}$$

Further, if Eq. A.4 inverses, one obtains Eq. A.5.

$$\frac{1}{ZthC_N(s)} = s \cdot c_N + \frac{1}{r_N + ZthC_{N-1}(s)} \tag{A.5}$$

Now, if the Foster network is described by a fraction of two polynomials in the Laplace domain, that is by Eq. A.6, the only one problem is to find the coefficients of such a division.

$$ZthF_N(s) = \frac{p_N(s)}{q_N(s)} \tag{A.6}$$

In particular, Eq. A.4 can be transformed to a polynomial as well (Eq. A.7).

$$\frac{1}{ZthC_N(s)} = \frac{q_N(s)}{p_N(s)} = s \cdot c_N^{'} + k_N + \frac{reminder_N(s)}{p_n(s)} \tag{A.7}$$

where, the degree of the $reminder_N(s)$ is smaller than the $p_n(s)$.

Comparing both Eqs. A.5 & A.7, following conclusions can be done:

Capacitors. N^{th} capacitance c_N in the Cauer network is equal to $c_N^{'}$.

$$c_N = c_N^{'} \tag{A.8}$$

Resistors. N^{th} resistance r_N in the Cauer network is equal to $1/k_N$.

$$r_N = \frac{1}{k_N} \tag{A.9}$$

N-1th impedance. The impedance of the N^{th}-*1* Cauer network block $ZthC_{N-1}$ is equal to a fraction: $p_{N-1}(s)/q_{N-1}(s)$, where $q_{N-1}(s) = k_N \cdot p_n(s) + reminder_N(s)$ and $p_{n-1} = -reminder_N(s)/k_N$.

Conclusions The Cauer network can be easily obtained from the Foster one. A Foster network must be described in a polynomial form (Eq. A.6). Division operations are performed using standard Euclidean algorithm, according to mathematical derivations described in this section.

A.5. Cauer network identification

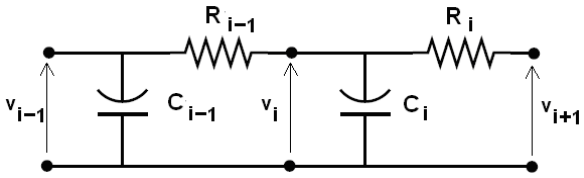


Figure A.6: Cauer type ladder network.

Finally it is possible to identify the Cauer network, using two schemes:

1. The 1st one identifies the circuit voltage at the node n, when adjacent voltages on *n-1* and *n+1* are known.

$$\frac{dv_i}{dt} = \frac{1}{C_i}\left(\frac{v_{i-1} - v_i}{R_{i-1}} - \frac{v_i - v_{i+1}}{R_i}\right) \tag{A.10}$$

   The disadvantage is that the adjacent voltages must be known:

2. The 2nd one identifies the circuit using the well-known *Crank-Nicholson scheme for 1D problem with 2-pass Simplex method*. The advantage is that the problem can be easily described in a matrix form of:

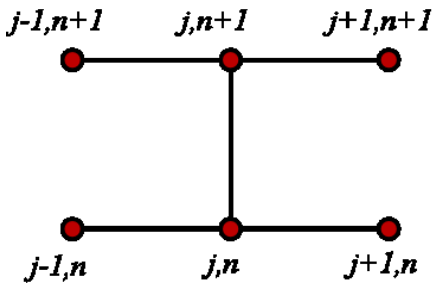


Figure A.7: The Crank-Nicolson stencil for 1D problem.

$$\mathbf{A}(\mathbf{R}, \mathbf{C}) \cdot v_{i,n+1} = \mathbf{B}(\mathbf{R}, \mathbf{C}) \cdot v_i + \text{Initial Condition} \tag{A.11}$$

Knowing the (1) *Initial Condition* (setting the voltages to zero); (2) setting the initial values for resistors R and capacitors C to obtain the matrices $\mathbf{A}$ and $\mathbf{B}$, the voltages of the next step are known in form:

$$v_{i,n+1} = \mathbf{A}(\mathbf{R}, \mathbf{C})^{-\mathbf{1}} \cdot (\mathbf{B}(\mathbf{R}, \mathbf{C}) \cdot v_i + \text{Initial Condition}) \tag{A.12}$$

Acknowledgements

First, I want to express my sincere gratitude to my supervisor prof. Niccolò Rinaldi for leading me during the PhD stage. I would like to thank also to Francesco de Paola for kind help during my beginnings in Naples. Special thanks to prof. Vincenzo d'Alessandro for helping me with publications. Prof. Rinaldi, Francesco and Vincenzo were my best teachers during my PhD.
I would like to thank also to Grazia Sasso, Salvatore Russo and Ilaria Marano for a cooperation. We were together in our RF/Electrothermal group.

Being abroad for almost four years, I have met a lot of people from all over the world. I would like to thank: my polish roommate Dominik Laskowski; Raffaele Gaetano for kind help with Italian and French bureaucracy; "prof" Giuseppe Scarpa and prof. Giacinto Gelli for interesting conversations.
I would like to greet and thank all PhD students of the 19^{th}, 20^{th}, 21^{st}, 22^{nd} cycle from the DIBET Department. Special thanks to Valeria Garofalo, Marino Coppola - my roommates at the department; Edén Bojorquez from Mexico; Ricardo Alzate Castaño, Julian Londono and Naile Aguirre from Colombia.
I would like to thank also the DIBET administration team, which helped me with bureaucracy: Annamaria Postiglione, Franco de Luca, Valentina Carrera and especially Roberto De Martino for cakes and joking all the time.

I really enjoyed my PhD stage in LAAS-CNRS, and I would like to thank to prof. Patrick Tounsi for the stage possibility, finished with several publications I made with the ISGE group. Of course I would like to appreciate the intensive and friendly cooperation with Francesc Madrid during my stay at LAAS. Special thanks to my friends: Piotrek, Cezary, Michał, Francesc, Ignacio, Javier, Olesya, Aloña, Hussein, Gaetan and Hakim. They helped me during the PhD stage in LAAS-CNRS as well.

However, I would like to thank to my close family. Without their help and support my PhD in Naples/Toulouse would not be possible. I dedicate this PhD thesis for them ....

List of publications

- ⋆ F. M. De Paola, J. P. Nowakowski, V. d'Alessandro, and N. Rinaldi, "Fully automated electrothermal simulation using standard CAD tools," in Proc. IEEE MIEL, vol. 2, pp. 483-486, 2006.
- ⋆ J.P. Nowakowski, V. d'Alessandro, F. M. De Paola, M. Spirito, and N. Rinaldi "Advances in electrothermal simulation of solid-state devices and circuits using commercial CAD tools," in Proc. IEEE MICROTHERM, pp. 65-72, 2007.
- ⋆ N. Rinaldi, V. d'Alessandro, I. Marano, J. P. Nowakowski, and M. Spirito "Electrothermal Phenomena in Solid-State Devices and Circuits: A Review and Progress Report (invited)," in Proc. IEEE MICROTHERM, pp. 11-22, 2007.
- ⋆ P. Tounsi, F. Madrid, W. Habra, and J. P. Nowakowski: Boundary Condition Adaptive Thermal Compact Models for Multi-cooling Surfaces and Multi-heat Sources Power Packages in Proc. ISPS08 2008
- ⋆ P. Tounsi, F. Madrid, and J. P. Nowakowski: "Nonlinear thermal resistance control equations for adaptive multiple cooling surface CTM, and boundary condition independent multiple heat sources CTM" in Proc. IEEE ThETA 2, 17th -20th December 2008. (accepted)
- ⋆ F. Madrid, P. Tounsi, J.P. Nowakowski, and W. Habra: "Nonlinear parameter control equations for boundary condition self-adaptable Compact Thermal Models MATCOM – "Mathematics, and Computers in Simulation" special issue of Electrimacs 2008, Quebec Canada. ELSEVIER, Journal of Mathematics and Computing http://ees.elsevier.com/matcom (accepted)

Bibliography

[1] E. Pop, S. Sinha, and K.E. Goodson. Heat generation and transport in nanometer-scale transistors. *Proceedings of the IEEE*, 94(8):1587–1601, Aug. 2006.

[2] M. Pfost, P. Brenner, and R. Lachner. Investigation of advanced sige heterojunction bipolar transistors at high power densities. *Bipolar/BiCMOS Circuits and Technology, 2004. Proceedings of the 2004 Meeting*, pages 100–103, Sept. 2004.

[3] J. Bock, H. Schafer, H. Knapp, K. Aufinger, M. Wurzer, S. Boguth, T. Bottner, R. Stengl, W. Perndl, and T.F. Meister. 3.3 ps sige bipolar technology. *Electron Devices Meeting, 2004. IEDM Technical Digest. IEEE International*, pages 255–258, Dec. 2004.

[4] M. Pfost, V. Kubrak, and P. Brenner. A practical method to extract the thermal resistance for heterojunction bipolar transistors. *European Solid-State Device Research, 2003. ESSDERC '03. 33rd Conference on*, pages 335–338, Sept. 2003.

[5] T. Hashimoto, Y. Nonaka, T. Tominari, H. Fujiwara, K. Tokunaga, M. Arai, S. Wada, T. Udo, M. Seto, M. Miura, H. Shimamoto, K. Washio, and H. Tomioka. Direction to improve sige bicmos technology featuring 200-ghz sige hbt and 80-nm gate cmos. *Electron Devices Meeting, 2003. IEDM '03 Technical Digest. IEEE International*, pages 5.5.1–5.5.4, Dec. 2003.

[6] M. Khater, J.-S. Rieh, T. Adam, A. Chinthakindi, J. Johnson, R. Krishnasamy, M. Meghelli, F. Pagette, D. Sanderson, C. Schnabel, K.T. Schonenberg, P. Smith, K. Stein, A. Strieker, S.-J. Jeng, D. Ahlgren, and G. Freeman. Sige hbt technology with fmax/ft=350/300 ghz and gate delay below 3.3 ps. *Electron Devices Meeting, 2004. IEDM Technical Digest. IEEE International*, pages 247–250, Dec. 2004.

[7] B.A. Orner, Q.Z. Liu, B. Rainey, A. Stricker, P. Geiss, P. Gray, M. Zierak, M. Gordon, D. Collins, V. Ramachandran, W. Hodge, C. Willets, A. Joseph, J. Dunn, J.-S. Rieh, S.-J. Jeng, E. Eld, G. Freeman, and D. Ahlgren. A 0.13 um bicmos technology featuring a 200/280 ghz (ft/fmax) sige hbt. *Bipolar/BiCMOS Circuits and Technology Meeting, 2003. Proceedings of the*, pages 203–206, Sept. 2003.

[8] B. Jagannathan, M. Khater, F. Pagette, J.-S. Rieh, D. Angell, H. Chen, J. Florkey, F. Golan, D.R. Greenberg, R. Groves, S.J. Jeng, J. Johnson, E. Mengistu, K.T. Schonenberg, C.M. Schnabel, P. Smith, A. Stricker, D. Ahlgren, G. Freeman, K. Stein, and S. Subbanna. Self-aligned sige npn transistors with 285 ghz fmax and 207 ghz ft in a manufacturable technology. *Electron Device Letters, IEEE*, 23(5):258–260, May 2002.

[9] A.J. Joseph, D.L. Harame, B. Jagannathan, D. Coolbaugh, D. Ahlgren, J. Magerlein, L. Lanzerotti, N. Feilchenfeld, S. St Onge, J. Dunn, and E. Nowak. Status and direction of communication technologies - sige bicmos and rfcmos. *Proceedings of the IEEE*, 93(9):1539–1558, Sept. 2005.

[10] A.J. Joseph, J. Dunn, G. Freeman, D.L. Harame, D. Coolbaugh, R. Groves, K.J. Stein, R. Volant, S. Subbanna, V.S. Marangos, S.S. Onge, E. Eshun, P. Cooper, J.B. Johnson, Jae-Sung Rieh, B. Jagannathan, V. Ramachandran, D. Ahlgren, Dawn Wang, and X. Wang. Product applications and technology directions with sige bicmos. *Solid-State Circuits, IEEE Journal of*, 38(9):1471–1478, Sept. 2003.

[11] B.G. Malm, E. Haralson, T. Johansson, and M. Ostling. Self-heating effects in a bicmos on soi technology for rfic applications. *Electron Devices, IEEE Transactions on*, 52(7):1423–1428, July 2005.

[12] R.C. Joy and E.S. Schlig. Thermal properties of very fast transistors. *Electron Devices, IEEE Transactions on*, 17(8):586–594, Aug 1970.

[13] Y.C. Gerstenmaier, W. Kiffe, and G. Wachutka. Combination of thermal subsystems modeled by rapid circuit transformation. *Thermal Investigation of ICs and Systems, 2007. THERMINIC 2007. 13th International Workshop on*, pages 115–120, Sept. 2007.

[14] L.K. Nanver, N. Nenadovic, V. d'Alessandro, H. Schellevis, H.W. van Zeijl, R. Dekker, D.B. de Mooij, V. Zieren, and J.W. Slotboom. A back-wafer contacted silicon-on-glass integrated bipolar process. part i. the conflict electrical versus thermal isolation. *Electron Devices, IEEE Transactions on*, 51(1):42–50, Jan. 2004.

[15] G.E. Moore. Cramming more components onto integrated circuits. *Electronics*, 38, Apr 1965.

[16] S. Borkar. Low power design challenges for the decade. *Design Automation Conference, 2001. Proceedings of the ASP-DAC 2001. Asia and South Pacific*, pages 293–296, 2001.

[17] O. Semenov, A. Vassighi, M. Sachdev, A. Keshavarzi, and C.F. Hawkins. Effect of cmos technology scaling on thermal management during burn-in. *Semiconductor Manufacturing, IEEE Transactions on*, 16(4):686–695, Nov. 2003.

[18] H.A. Schafft. Second breakdown—a comprehensive review. *Proceedings of the IEEE*, 55(8):1272–1288, Aug. 1967.

[19] N. Rinaldi and V. d'Alessandro. Theory of electrothermal behavior of bipolar transistors: Part i -single-finger devices. *Electron Devices, IEEE Transactions on*, 52(9):2009–2021, Sept. 2005.

[20] N. Rinaldi and V. d'Alessandro. Theory of electrothermal behavior of bipolar transistors: Part ii-two-finger devices. *Electron Devices, IEEE Transactions on*, 52(9):2022–2033, Sept. 2005.

[21] N. Rinaldi and V. d'Alessandro. Theory of electrothermal behavior of bipolar transistors: part iii-impact ionization. *Electron Devices, IEEE Transactions on*, 53(7):1683–1697, July 2006.

[22] Michaël Bairanzade. Understanding power transistors breakdown parameters. Technical report, ON semiconductor, http://onsemi.com, Aug. 2003. AN1628-D.

[23] John Chi-Hung Hui, Tzu-Yin Chiu, S.-W.S. Wong, and W.G. Oldham. Sealed-interface local oxidation technology. *Solid-State Circuits, IEEE Journal of*, 17(2):184–191, Apr 1982.

[24] K.Y. Chiu, R. Fang, J. Lin, J.L. Moll, C. Lage, S. Angelos, and R. Tillman. The swami - a defect free and near-zero bird's-beak local oxidation process and its application in vlsi technology. *Electron Devices Meeting, 1982 International*, 28:224–227, 1982.

[25] Wang Qingping, Zhang Zhengfan, Li Kaicheng, Guo Lin, Xiangdong Wu, and Wang Zihang. A deep trench isolation for silicon. *Solid-State and Integrated Circuit Technology, 1998. Proceedings. 1998 5th International Conference on*, pages 172–175, 1998.

[26] B. El-Kareh, Bomy Chen, and T. Stanley. Silicon on insulator-an emerging high-leverage technology. *Components, Packaging, and Manufacturing Technology, Part A, IEEE Transactions on*, 18(1):187–194, Mar 1995.

[27] E. Pop, Chi On Chui, R. Dutton, S. Sinha, and K. Goodson. Electro-thermal comparison and performance optimization of thin-body soi and goi mosfets. *Electron Devices Meeting, 2004. IEDM Technical Digest. IEEE International*, pages 411–414, Dec. 2004.

[28] N. Nenadovic, L.K. Nanver, and J.W. Slotboom. Electrothermal limitations on the current density of high-frequency bipolar transistors. *Electron Devices, IEEE Transactions on*, 51(12):2175–2180, Dec. 2004.

[29] N. Nenadovic, V. d'Alessandro, L.K. Nanver, F. Tamigi, N. Rinaldi, and J.W. Slotboom. A back-wafer contacted silicon-on-glass integrated bipolar process. part

ii. a novel analysis of thermal breakdown. *Electron Devices, IEEE Transactions on*, 51(1):51–62, Jan. 2004.

[30] J.N. Burghartz, J.D. Cressler, J. Warnock, R.C. McIntosh, K.A. Jenkins, J.Y.-C. Sun, J.H. Comfort, J.M.C. Stork, C.L. Stanis, W. Lee, and D.D. Danner. Partial-soi isolation structure for reduced bipolar transistor parasitics. *Electron Device Letters, IEEE*, 13(8):424–426, Aug 1992.

[31] J.N. Burghartz, R.C. McIntosh, and C.L. Stanis. A low-capacitance bipolar/bicmos isolation technology. i. concept, fabrication process, and characterization. *Electron Devices, IEEE Transactions on*, 41(8):1379–1387, Aug 1994.

[32] J.N. Burghartz, A.O. Cifuentes, and J.D. Warnock. A low-capacitance bipolar/bicmos isolation technology. ii. circuit performance and device self-heating. *Electron Devices, IEEE Transactions on*, 41(8):1388–1395, Aug 1994.

[33] T. Iwamatsu, T. Ipposhi, T. Uchida, S. Maegawa, and M. Inuishi. Suppression of stress induced drain leakage current of soi mosfets by using partial trench isolation technology. *SOI Conference, 2000 IEEE International*, pages 80–81, 2000.

[34] W. B. Joyce. On the application of the kirchhoff transformation to the steady-state thermal analysis of semiconductor devices with temperature-dependent and piecewise inhomogeneous thermal conductivity. *Solid-State Electronics*, 38(7):1409–1412, 1995.

[35] N. Rinaldi. Thermal analysis of solid-state devices and circuits: ananalytical approach. *Solid-State Electronics*, 44:1789–1798, May 2000.

[36] V. Szekely. On the representation of infinite-length distributed rc one-ports. *Circuits and Systems, IEEE Transactions on*, 38(7):711–719, Jul 1991.

[37] A. Poppe. Measurement, modeling and simulation of thermal dynamics in microelectronic structures. Technical report, REASON Tutorial "Half-day on Thermal Issues in Microelectronics", Lodz, http://reason.imio.pw.edu.pl, 2003.

[38] Y.C. Gerstenmaier, H. A Pape, and G. Wachutka. Rigorous model and network for static thermal problems. *Microelectronics Journal*, 33(9):711–718, September 1 2002.

[39] H.I. Rosten, J.D. Parry, C.J.M. Lasance, H. Vinke, W. Temmerman, W. Nelemans, Y. Assouad, T. Gautier, O. Slattery, C. Cahill, M. O'Flattery, C. Lacaze, and P. Zemlianoy. Final report to semitherm xiii on the european-funded project delphi-the development of libraries and physical models for an integrated design environment. *Semiconductor Thermal Measurement and Management Symposium, 1997. SEMI-THERM XIII., Thirteenth Annual IEEE*, pages 73–91, Jan 1997.

[40] H.I. Rosten, C.J.M. Lasance, and J.D. Parry. The world of thermal characterization according to delphi-part i: Background to delphi. *Components, Packaging, and Manufacturing Technology, Part A, IEEE Transactions on*, 20(4):384–391, Dec 1997.

[41] C.J.M. Lasance, H.I. Rosten, and J.D. Parry. The world of thermal characterization according to delphi-part ii: Experimental and numerical methods. *Components, Packaging, and Manufacturing Technology, Part A, IEEE Transactions on*, 20(4):392–398, Dec 1997.

[42] H.I. Rosten. Delphi-a status report on the european-funded project for the development of libraries and physical models for an integrated design environment. *Electronic Components and Technology Conference, 1996. Proceedings., 46th*, pages 172–185, May 1996.

[43] H.M. Gutierrez, C.E. Christoffersen, and M.B. Steer. An integrated environment for the simulation of electrical, thermal and electromagnetic interactions in high-performance integrated circuits. *Electrical Performance of Electronic Packaging, 1999*, pages 217–220, 1999.

[44] C.J.M. Lasance, H. Vinke, and H. Rosten. Thermal characterization of electronic devices with boundary condition independent compact models. *Components, Packaging, and Manufacturing Technology, Part A, IEEE Transactions on*, 18(4):723–731, Dec 1995.

[45] M. Rencz, G. Farkas, A. Poppe, V. Szekely, and B. Courtois. A methodology for the generation of dynamic compact models of packages and heat sinks from thermal transient measurements. *Electronics Manufacturing Technology Symposium, 2003. IEMT 2003. IEEE/CPMT/SEMI 28th International*, pages 117–123, July 2003.

[46] M. Rencz, V. Szekely, A. Poppe, and B. Courtois. An algorithm for the direct co-simulation of dynamic compact models of packages with the detailed thermal models of boards. *Electronics Packaging Technology Conference, 2002. 4th*, pages 293–298, Dec. 2002.

[47] W. Habra, P. Tounsi, and J.-M. Dorkel. Improved 3d-nonlinear compact modelling for power components. *Thermal, Mechanical and Multi-Physics Simulation and Experiments in Micro-Electronics and Micro-Systems, 2005. EuroSimE 2005. Proceedings of the 6th International Conference on*, pages 390–393, April 2005.

[48] W. Habra, P. Tounsi, and J.M. Dorkel. Transient compact modeling for multi chips components. *Thermal Investigation of ICs and Systems, 2005. THERMINIC 2005. 13th International Workshop on*, pages 390–393, Sept. 2005.

[49] W. Habra, P. Tounsi, and J.M. Dorkel. Advanced compact thermal modeling using vhdl-ams. *Thermal Investigation of ICs and Systems, 2006. THERMINIC 2006. 14th International Workshop on*, pages 390–393, Sept. 2006.

[50] C. C. McAndrew. A complete and consistent electrical/thermal hbt model. In *Proc. of the IEEE Bipolar / BiCMOS Circuits and Technology Meeting*, pages 200–203, 1992.

[51] F. M. De Paola, J. P. Nowakowski, V. d'Alessandro, and N. Rinaldi. Fully automated electrothermal simulation using standard cad tools. *Microelectronics, 2006. MIEL 2006. 25th International Conference on*, pages 483–486, 2006.

[52] Agilent Technologies. http://eesof.viewmark.com/docs/. ADS Online Documentation.

[53] M. Macchiaroli, V. d'Alessandro, and N. Rinaldi. Nasdac - a new simulation tool for the electro-thermal analysis of bipolar devices: application to multi-finger algaas/gaas hbt's. *Microelectronics, 2002. MIEL 2002. 23rd International Conference on*, 2:455–458, 2002.

[54] M. Macchiaroli, N. Rinaldi, V. d'Alessandro, G. Breglio, and P. Spirito. A new electro-thermal simulation tool for the analysis of bipolar devices and circuitas. In *THERMINIC*, 2001.

[55] R. H. Winkler. Thermal properties of high-power transistors. *IEEE Transactions on Electron Devices*, ED-14(5):260–263, May 1967.

[56] N. Nenadovic, V. d'Alessandro, F. Tamigi, A. Rossi, A. Griffo, L. K. Nanver, and J. W. Slotboom. Thermal instability in two-finger bipolar transistors. *IEEE ESSDERC, in Proc*, pages 203–206, 2003.

[57] N. Nenadovic, S. Mijalkovic, L. K. Nanver, L. J. K. Vandamme, V. d'Alessandro, H. SSchellevis, and Slotboom J. W. Extraction and modelling of self-heating and mutual thermal coupling impedance of bipolar transistors. *IEEE Journal of Solid-State Circuits*, 39(10):1764–1772, Oct. 2004.

[58] D. Buhanan. Investigation of current-gain temperature dependence in silicon transistors. *IEEE Transactions on Electron Devices*, ED-16(1):117–124, Jan. 1969.

[59] J. W. Slotboom and H. C. De Graaff. Bandgap narrowing in silicon bipolar transistors. *IEEE Transactions on Electron Devices*, ED-24(8):1123–1125, Aug. 1977.

[60] O. Müller and J. Pest. Thermal feedback in power semiconductor devices. *IEEE Transactions on Electron Devices*, ED-17(9):770–782, Sep. 1970.

[61] W. Liu. Thermal coupling in 2-finger heterojunction bipolar transistors. *IEEE Transactions on Electron Devices*, 42(6):1033–1038, Jun. 1995.

[62] K. J. Negus, R. W. Franklin, and Yovanovich M. M. Thermal modeling and experimental techniques for microwave bipolar devices. *IEEE Transactions on Components, Hybrids, and Manufacturing Technology*, 12(4):680–689, Dec. 1989.

[63] W. Liu, S.-K. Fan, T. Henderson, and Davito D. Temperature dependences of current gains in gainp/gaas and algaas/gaas heterojunction bipolar transistors. *IEEE Transactions on Electron Devices*, 40(7):1351–1353, Jul. 1993.

[64] W. Liu, S. Nelson, D. G. Hill, and A. Khatibzadeh. Current gain collapse in microwave multifinger heterojunction bipolar transistors operated at very high current densities. *IEEE Transactions on Electron Devices*, 40(11), Nov. 1993.

[65] L. L. Liou and B. Bayraktaroglu. Thermal stability analysis of algaas/gaas heterojunction bipolar transistors with multiple emitter fingers. *IEEE Transactions on Electron Devices*, 41(5):629–636, May 1994.

[66] G. Gao, M. S. Ünlü, H. Morko¸c, and D. L. Blackburn. Emitter ballasting resistor design for, and current handling capability of algaas/gaas power heterojunction bipolar transistors. *IEEE Transactions on Electron Devices*, 38(2):185–196, 1991.

[67] C.-H. Liao, C.-P. Lee, N. L. Wang, and B. Lin. Optimum design for a thermally stable multifinger power transistor. *IEEE Transactions on Electron Devices*, 49(5):902–908, May 2002.

[68] V. Szekely. Identification of rc networks by deconvolution: chances and limits. *Circuits and Systems I: Fundamental Theory and Applications, IEEE Transactions on*, 45(3):244–258, Mar 1998.

[69] M. Rencz, V. Székely, Zs. Kohári, and Courtois B. A method for thermal model generation of mems packages. In *Technical Proceedings of the 2000 International Conference on Modeling and Simulation of Microsystems*, pages 209 – 212. Micred Ltd., Hungary, 2000.

[70] J. A. Nelder and R. Mead. A simplex method for function minimization. *The Computer Journal*, 7(4):308–313, 1965.