



HAL
open science

A very high speed method to simulate quartz crystal oscillator

Nicolas Ratier, Mickaël Bruniaux, Serge Galliou, Rémi Brendel

► **To cite this version:**

Nicolas Ratier, Mickaël Bruniaux, Serge Galliou, Rémi Brendel. A very high speed method to simulate quartz crystal oscillator. 19th European Frequency and Time Forum (EFTF), Mar 2005, Besançon, France. pp.413-418. hal-00345214

HAL Id: hal-00345214

<https://hal.science/hal-00345214>

Submitted on 8 Dec 2008

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

A very high speed method to simulate quartz crystal oscillator

Nicolas Ratier, Mickaël Bruniaux, Serge Galliou, Rémi Brendel

Institut FEMTO-ST, département LPMO, CNRS UMR 6174,
32 avenue de l'Observatoire, F-25044 Besançon, France
nicolas.ratier@lpmo.edu

and Jérôme Delporte

Centre National d'Etudes Spatiales, Toulouse, France

Abstract— In this paper, we present the SHA method, a Symbolic Harmonic Analysis method to simulate the behaviour of ultrastable quartz crystal oscillators. This nonlinear method is aimed to compute very quickly the steady state as well as amplitude and frequency transients. The ultimate goal is to see instantaneously the influence of a parameter change on the oscillator's features thank to the computation speed. The method proposed here is a mixing of the nonlinear dipolar method previously developed in our team and the harmonic method. It allows to replace the set of algebro-differential equation of the circuit by a nonlinear system of the Fourier coefficients of the circuit unknowns.

1. INTRODUCTION

We develop a method to analyse the behaviour of quartz crystal oscillators with very high quality factor. The ultimate aim is to perform a real time analysis of this type of oscillators. The method consists in replacing the whole circuit connected to the quartz by a nonlinear dipole whose impedance depends on the current amplitude evaluated at the resonator frequency.

The nonlinear dipolar method [1][2][3] shows that the knowledge of the nonlinear dipole is sufficient to determine the oscillator behaviour. The nonlinear dipolar impedance is computed by replacing the motional part of the quartz by a sinusoidal current source at the resonator frequency. Using an external current source overcomes the drawback of the large resonator time constant thus allowing an efficient harmonic analysis.

The originality of the present approach is the computation of the dipolar impedance from the circuit equations by using a symbolic calculation method. Our symbolic harmonic analysis replaces the original set of differential equations describing the circuit by a set of nonlinear equations in the Fourier coefficients of the circuit unknowns (Node voltages, ...).

Nonlinear dipolar method is first recalled, then, the new method that replaces the dipolar impedance calculation by a symbolic harmonic analysis of the equations is developed. This method is faced with two major difficulties: The processing of the nonlinear components and the management of a very large number of coefficients. The solution proposed

to solve these two problems is based on the tree parsing. The method is detailed on a basic oscillator to ease the understanding.

2. PRINCIPLE OF THE NONLINEAR DIPOLAR METHOD

The quartz crystal oscillator is split into two parts: the motional part of the resonator (in short “resonator” by misuse of language) and the amplifier part (including the parallel capacitance C_p), Fig. 1

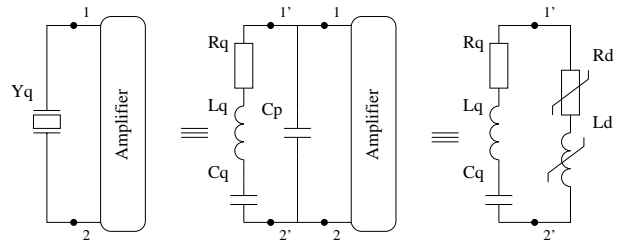


Fig. 1. Dipolar representation

The resonator is regarded as an impedance strongly varying with the frequency and slightly with the amplitude (because of the amplitude–frequency effect) while the amplifier is considered as a nonlinear dipole whose impedance strongly depends on the current amplitude and weakly varying with frequency. These two remarks permit to consider the amplifier part as an nonlinear dipole (R_d , L_d) that depends only on the current amplitude as shown in Fig. 2 right.

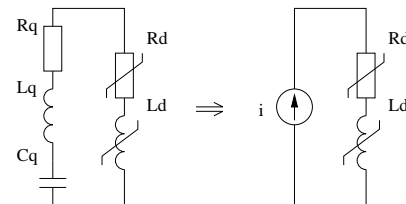


Fig. 2. Amplifier impedance determination

The determination of the equivalent single port network for the amplifier is obtained by replacing the resonator motional branch by a sinusoidal current source of amplitude a and frequency equal to the

resonator's frequency f_q as shown in Fig. 2. Then, a SPICE transient analysis is performed for different current amplitudes. Eventually, for each value of the current amplitude a Fourier transform is calculated to obtain the first harmonic amplitude and phase of the dipolar voltage. The nonlinear dipole R_d et L_d are then obtained as a function of the loop current amplitude a .

The knowledge of $R_d(a)$ and $L_d(a)$ allows us to determine the behaviour of the oscillator. The steady state amplitude a_0 is obtained by solving the implicate nonlinear equation Eq. (1).

$$R_q + R_d(a) = 0. \quad (1)$$

The steady state "frequency" ω_0 of the oscillation is given by Eq. (2).

$$\omega_0^2 = \omega_q^2 \left(1 - \frac{L_d(a_0)}{L_q} \right) \quad (2)$$

The computation time of the curves $R_d(a)$ and $L_d(a)$ is of the order of a few seconds. At each parameter change, for example a change of a resistance value, the computation must be redone. Our aim is to reduce the computation time for the curves $R_d(a)$ and $L_d(a)$ to a few milliseconds allowing to see, in real time, the influence of each circuit component on the behaviour of the oscillator.

3. PRINCIPLE OF THE SYMBOLIC HARMONIC METHOD

- Compute the set of ODE describing the circuit in symbolic form, including the nonlinear elements (diode, transistor NPN, ...).

$$\frac{dV_1}{dt}(t) = f(V_1(t), \dots, I_L(t)) \quad (3)$$

$$\frac{dI_L}{dt}(t) = f(V_1(t), \dots, I_L(t)) \quad (4)$$

- Replace the motional branch of the quartz by a sinusoidal current source.
- Replace all the unknowns $V_1(t), \dots, I_L(t)$ by their Fourier series limited to N terms.

$$V_1 = a_0^{V_1} + a_1^{V_1} \cos(\omega t) + b_1^{V_1} \sin(\omega t) + \dots \quad (5)$$

$$\dots = \dots$$

$$I_L = a_0^{I_L} + a_1^{I_L} \cos(\omega t) + b_1^{I_L} \sin(\omega t) + \dots \quad (6)$$

- Transform the set of ODE into a set of nonlinear equations.

$$f(a_0^{V_1}, a_1^{V_1}, b_1^{V_1}, \dots, a_N^{V_1}, b_N^{V_1}) = 0 \quad (7)$$

$$\dots = 0$$

$$f(a_0^{I_L}, a_1^{I_L}, b_1^{I_L}, \dots, a_N^{I_L}, b_N^{I_L}) = 0 \quad (8)$$

- Solve numerically this system.

- By using a powerful numerical algorithm, the symbolic algebraic system can be solved very quickly (ms). Because of the quite large Q factor of the resonator, higher order harmonics have no significant contribution on the dipolar current so that only the first order harmonic coefficients a_1 and b_1 are needed to calculate the dipolar impedance.

4. THREE MAIN PROBLEMS

- How to have a linear complexity? A direct and naive application of the method to build the nonlinear system leads to an exponential growth of the coefficient numbers. That leads to an untractable number of equations, even in the case of basic circuits. So, a method leading to a linear dependence of the coefficient number on the circuit component number has to be found.
- How to manage nonlinear elements? An oscillator is by nature a highly nonlinear circuit. In the case of quartz oscillators with high spectral purity, the nonlinearities usually come from the bipolar transistors. For example, the Ebers-Moll static model transistor (Eq. 9, 10, 11) [4] has an exponential non linearity.

$$I_{BE} = \frac{I_S}{\beta_F} (e^{qV_{BE}/kT} - 1) \quad (9)$$

$$I_{BC} = \frac{I_S}{\beta_R} (e^{qV_{BC}/kT} - 1) \quad (10)$$

$$I_{CT} = I_S (e^{qV_{BE}/kT} - e^{qV_{BC}/kT}) \quad (11)$$

- How to solve efficiently the nonlinear system? The resulting system is highly nonlinear with a small density. In the case of a simple oscillator, the classical Newton–Raphson method is sufficient. In the case of an oscillator with a higher number of components, one should develop specific algorithms. Efforts are currently made in this direction.

5. HOW TO ACHIEVE A LINEAR COMPLEXITY?

To ease the understanding, let's demonstrate the method on the simple transconductance oscillator shown in Fig. 3.

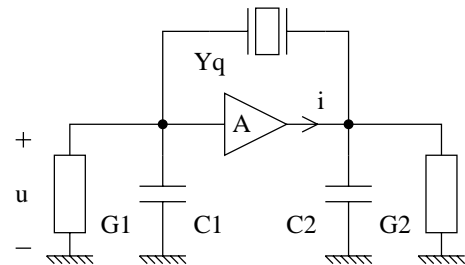


Fig. 3. Transconductance oscillator

The system of equations that models the electronic circuit (Eq. 12 and 13) is build symbolically from a circuit description by using the modified nodal

method (MMM)[5]. It consists in applying the Kirchoff Current Law (KCL) to each circuit node, and applying the constitutive or branch equations to each circuit element. The programme used is fully home-made and does not call for any external software.

$$-a \sin(\omega t) + G_1 V_1(t) + C_1 \frac{dV_1}{dt}(t) = 0 \quad (12)$$

$$a \sin(\omega t) + G_2 V_2(t) + C_2 \frac{dV_2}{dt}(t) + G_1 V_1(t)(1 - \varepsilon V_1(t)^2) = 0 \quad (13)$$

The problem is to express each equation in harmonic form, with the constraint that the coefficient number generated during this phase does not increase exponentially.

The key point of the method is to build in a binary tree relative to each equation. Each node represents an algebraic operation. The whole tree can be expressed into an harmonic form as soon as the method to express each leaf and to reduce each node into an harmonic form is known.

The tree will be progressively reduced in harmonic form from the bottom to the top: at each node new coefficients expressed in fonction of the previous one will be generated, so we will have a linear complexity of the generated coefficient. The formal definition of the binary tree is written in OCAML-like syntax [6] below. This kind of definition is generally called "abstract syntax".

```

type equTree =
  SUM of equTree * equTree
  | PROD of equTree * equTree
  | DIFF of equTree
  | EXP of equTree
  | COS of equTree
  | SIN of equTree
  | UNK of unknown
  | FCT of function

```

For example, the binary tree (Fig. 4) represents the first Eq. (12).

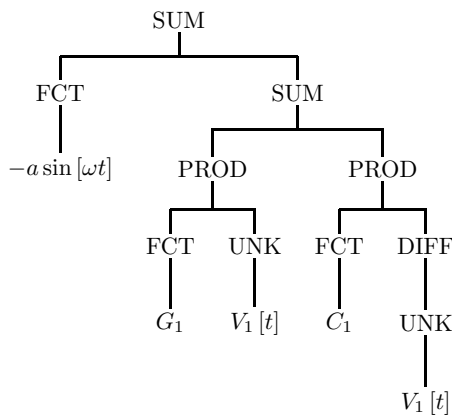


Fig. 4. Binary tree of expression

The representation has many advantages: since abstract syntax has a unique tree structure, it is not necessary to reflect binding strength using different types to avoid ambiguity. Moreover, the abstract syntax shows that the tree is binary, so each node involves only one operation (SUM, PROD, ...).

The abstract syntax processing consists in progressively parse the tree so that, at the final step, it will remain only a Fourier series with manageable coefficient. The processing of the nonlinear functions EXP, COS and SIN is performed in the next section.

- UNK(V1(t)) is replaced by its Fourier series limited to N terms, i.e. it return:

$$\begin{aligned}
 &A00V1 + A01V1 \cos wt + B01V1 \sin wt \\
 &\quad + A02V1 \cos 2wt + B02V1 \sin 2wt \\
 &\quad + A03V1 \cos 3wt + B03V1 \sin 3wt
 \end{aligned}$$

- FCT(G1) is already under the right form, so it is replaced by itself.

- SUM(S1, S2) with

$$\begin{aligned}
 S1 &= A00S1 + A01S1 \cos wt + B01S1 \sin wt \\
 S2 &= A00S2 + A01S2 \cos wt + B01S2 \sin wt
 \end{aligned}$$

is reduced by generating new coefficients

$$\begin{aligned}
 TT1 &= A00S1 + A00S2 \\
 TT2 &= A01S1 + A01S2 \\
 TT3 &= B01S1 + B01S2 \\
 \dots &= \dots
 \end{aligned}$$

and return

$$TT1 + TT2 \cos wt + TT3 \sin wt + \dots$$

- PROD(S1, S2), DIFF(S1) are reduced like SUM(S1, S2), however, the equations associated with the generated coefficients are a little bit more complex than the trivial case of SUM(S1, S2).

6. HOW TO MANAGE THE NONLINEAR ELEMENTS?

The problem is to represent the exponential, cosinus and sinus of a Fourier series into the form of a Fourier expansion. In other words, The problem is to express EXP(S1) with

$$S1 = A00S1 + A01S1 \cos(\omega t) + B01S1 \sin(\omega t)$$

Under the form

$$\begin{aligned}
 S2 &= \text{EXP}(S1) \\
 &= TT1 + TT2 \cos(\omega t) + TT3 \sin(\omega t) + \dots
 \end{aligned}$$

The solution of the problem is quite similar to the previous one. It is based on constructing a binary tree and then generating new coefficient function of the previous one during the tree parsing. In this problem, the concrete grammar to analyse is written in Fig. 5

The definition of the tree is based on the addition formulas. Of course, each transcendental function EXP, COS, SIN will have a different associated tree. The reduction of the tree will be possible thanks to their associated Bessel series.

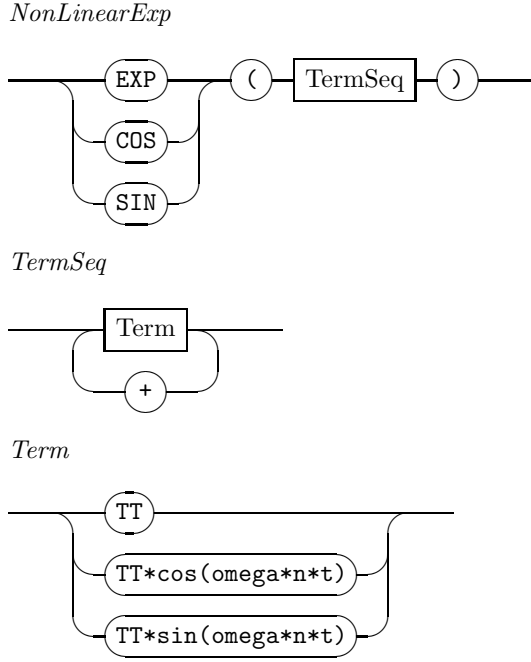


Fig. 5. Concrete grammar

7. HARMONIZATION OF EXP FUNCTION

A recursive application of the addition formula for the exponent function (Eq. 14) allows one to define an abstract grammar that represents a binary tree of EXP.

$$\exp(a + b + c + \dots) = \exp(a) \exp(b + c + \dots) \quad (14)$$

For example, the expression

$$\text{EXP}(\text{A00S1} + \text{A01S1 COS wt} + \text{B01S1 SIN wt} + \text{A02S1 COS 2wt} + \text{B02S1 SIN 2wt})$$

is represented by the following binary tree (Fig. 6)

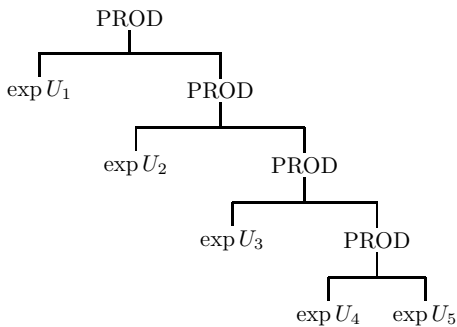


Fig. 6. Binary tree of exponential

with

$$\begin{aligned} U_1 &= \text{A00S1} \\ U_2 &= \text{A01S1 cos(wt)} \\ U_3 &= \text{B01S1 sin(wt)} \\ U_4 &= \text{A02S1 cos(2wt)} \\ U_5 &= \text{B02S1 sin(2wt)} \end{aligned}$$

The reduction of the tree is based on the associated Bessel series of the exponential function. Indeed, it is well known that the harmonic expansion of $\exp(z \cos \theta)$ and $\exp(z \sin \theta)$ can be expressed under the form Eq. (15) and Eq. (16) where $I_k(z)$ are the Bessel functions of the second kind and of integer order [7]. This allows us to harmonize the leafs $\text{EXP}(\text{A01S1 COS(wt)})$ and $\text{EXP}(\text{B01S1 SIN(wt)})$. The terms in $\text{EXP}(\text{A00S1})$ are already in the right form.

$$\exp(z \cos \theta) = I_0(z) \quad (15)$$

$$+ 2 \sum_{k=1}^{\infty} I_k(z) \cos(k\theta)$$

$$\exp(z \sin \theta) = I_0(z) \quad (16)$$

$$+ 2 \sum_{k=1}^{\infty} (-1)^k I_{2k}(z) \cos(2k\theta)$$

$$+ 2 \sum_{k=1}^{\infty} (-1)^k I_{2k+1}(z) \sin((2k+1)\theta)$$

8. HARMONIZATION OF COS AND SIN FUNCTIONS

The method used for EXP can be applied to COS and SIN, it is just a little bit more complicated because the trees for COS and SIN are mutually recursive.

A recursive application of the addition formula for cos (resp. sin), see Eq. (17) and Eq. (18), allows one to define an abstract grammar that represents a binary tree of COS (resp. SIN). Note that the construction of the tree for COS involves the tree of SIN, and conversely.

$$\cos(a + b + c + \dots) = \quad (17)$$

$$\cos(a) \cos(b + c + \dots)$$

$$- \sin(a) \sin(b + c + \dots)$$

$$\sin(a + b + c + \dots) = \quad (18)$$

$$\sin(a) \cos(b + c + \dots)$$

$$+ \cos(a) \sin(b + c + \dots)$$

For example

$$\text{COS}(\text{A00S1} + \text{A01S1 cos wt} + \text{B01S1 sin wt})$$

is represented by the binary tree shown in Fig. 7, with

$$\begin{aligned} U_1 &= \text{A00S1} \\ U_2 &= \text{A01S1 cos(wt)} \\ U_3 &= \text{B01S1 sin(wt)} \end{aligned}$$

Similarly,

$$\text{SIN}(\text{A00S1} + \text{A01S1 cos wt} + \text{B01S1 sin wt})$$

is represented by the binary tree shown in Fig. 8, with

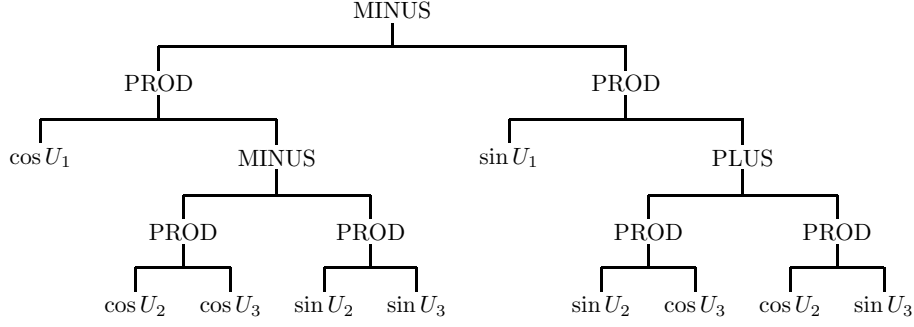


Fig. 7. Binary tree of cosinus

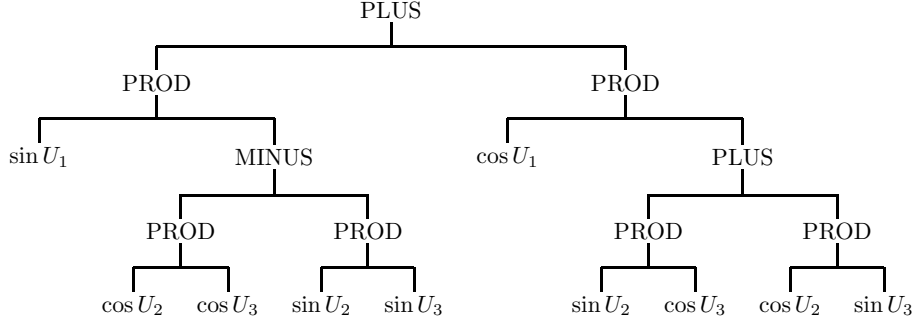


Fig. 8. Binary tree of sinus

$$\begin{aligned}
 U_1 &= \text{A00S1} \\
 U_2 &= \text{A01S1 } \cos(\text{wt}) \\
 U_3 &= \text{B01S1 } \sin(\text{wt})
 \end{aligned}$$

The reduction of the tree is based on the harmonic expansion of $\cos(z \cos \theta)$ and $\cos(z \sin \theta)$. These well known formulas are recalled here for completeness (Eq. (19) and Eq. (20)), where $J_k(z)$ are the Bessel functions of the first kind and of integer order [7]. These equations allow one to reduce the leaf $\text{COS}(\text{A01S1 } \text{COS}(\text{wt}))$ and $\text{COS}(\text{B01S1 } \text{SIN}(\text{wt}))$. The leaf $\text{COS}(\text{A00S1})$ is already in the right form.

$$\begin{aligned}
 \cos(z \cos \theta) &= J_0(z) \\
 &+ 2 \sum_{k=1}^{\infty} (-1)^k J_{2k}(z) \cos(2k\theta) \\
 \cos(z \sin \theta) &= J_0(z) \\
 &+ 2 \sum_{k=1}^{\infty} J_{2k}(z) \cos(2k\theta)
 \end{aligned}
 \tag{19}$$

Similarly, the reduction of the tree is based on the harmonic expansion of $\sin(z \cos \theta)$ and $\sin(z \sin \theta)$. These well known formulas are recalled here for completeness (Eq. (21) and Eq. (22)), where $J_k(z)$ are the Bessel functions of the first kind and of integer order [7]. These equations allow one to reduce the leaf $\text{SIN}(\text{A01S1 } \text{COS}(\text{wt}))$ and $\text{SIN}(\text{B01S1 } \text{SIN}(\text{wt}))$. The leaf $\text{SIN}(\text{A00S1})$ is already in the

right form.

$$\sin(z \cos \theta) = J_0(z) \tag{21}$$

$$+ 2 \sum_{k=1}^{\infty} (-1)^k J_{2k+1}(z) \cos((2k+1)\theta)$$

$$\sin(z \sin \theta) = J_0(z) \tag{22}$$

$$+ 2 \sum_{k=1}^{\infty} J_{2k+1}(z) \sin((2k+1)\theta)$$

9. HARMONIZATION OF THE OTHER FUNCTIONS

For all the other transcendental functions (log, tan, acos, ...) one proceeds by introducing an additional unknown to the equation system. This transforms the initial differential equation system into an algebro-differential system. This handling is trivial as shown in the following example.

$$\dots + \tan V_1 + \dots + \text{acos} V_2 + \dots = 0 \tag{23}$$

becomes

$$\dots + H_1 + \dots + H_2 + \dots = 0 \tag{24}$$

$$H_1 \cos V_1 = \sin V_1 \tag{25}$$

$$\cos H_2 = V_2 \tag{26}$$

The latter equations are actually written under the harmonized forms. The same method can be directly applied to the inverse of a function and to the composite functions.

10. CONCLUSION

In this paper, an extension of the nonlinear dipolar method has been presented to reduce by one order of magnitude the simulation time of ultrastable quartz oscillators. It has been demonstrated that replacing Spice simulations by a symbolic calculation is faced with two difficulties: The processing of the nonlinear components and the management of a large number of coefficients.

A solution to these problems has been found by rewriting the differential equations under the form of a binary tree. Each node of this tree corresponds to an algebraic operation, each leaf is either a parameter of the circuit (R, L, C, I_s, \dots) or an unknown of the problem (V_1, V_2, I_l, \dots).

This tree is then progressively reduced. At the end of this reduction process, the differential equation is transformed into a nonlinear equation involving only the Fourier coefficients of the unknowns. At each step of a node reduction, a set of equations relating the previous coefficients to the new ones is generated so that the number of equations describing the oscillator increases linearly with the number of components.

REFERENCES

- [1] R. Brendel, D. Gillet, N. Ratier, F. Lardet-Vieudrin, and J. Delporte, "Nonlinear dipolar modelling of quartz crystal oscillators," in *Proc. of 14th EFTF*, 2000.
- [2] M. Addouche, N. Ratier, D. Gillet, R. Brendel, F. Lardet-Vieudrin, and J. Delporte, "Adoq: A quartz crystal oscillators simulation software," in *Proc. Int. Freq. Contr. Symp.*, 2001, pp. 753–757.
- [3] M. Addouche, R. Brendel, D. Gillet, N. Ratier, F. Lardet-Vieudrin, and J. Delporte, "Modeling of quartz crystal oscillators by using nonlinear dipolar method," *IEEE Trans. Ultrason., Ferroelect., Freq. Contr.*, vol. 50, no. 5, pp. 487–495, May 2003.
- [4] Giuseppe Massobrio and Paolo Antognetti, *Semiconductor Device Modeling with SPICE*, MacGraw-Hill, Inc., second edition edition, 1993.
- [5] Jiří Vlach and Kishore Singhal, *Computer methods for circuit analysis and design*, Van Nostrand Reinhold, second edition edition, 1994.
- [6] Emmanuel Chailloux, Pascal Manoury, and Bruno Pagano., *Developing Applications With Objective Caml*, Online version (03/2005): <http://caml.inria.fr/pub/docs/oreilly-book/>.
- [7] Milton Abramowitz and Irene A. Stegun, *Handbook of Mathematical Functions*, Dover Publications, Inc., 1972.