TEACHING OF ELECTRICAL CIRCUITS USING SYMBOLIC AND SEMISYMBOLIC PROGRAMS

Prof. Dalibor Biolek*), Dr. Zdenek Kolka **), Dr. Bohumir Sviezeny***)

*) *Military Academy Brno, K301, Kounicova 65, PS13, 612 00 Brno, Czech Republic*
Tel. (+420) 5 41149190, fax (+420) 5 41149192, email: dalibor.biolek@vabo.cz

**) *Brno Univ. of Technology, Dept. of Radioelectronics, Purkynova 118, 612 00 Brno, Czech Republic*
Tel. (+420) 5 41149146, fax (+420) 5 41149192, email: kolka@urel.fee.vutbr.cz

***) *Institute Superieur d'elektronique de Paris (ISEP), 21, Rue de Assas, 75270 Paris, France*
tel. (+33) 1 45535490, fax (+33) 1450593 39, email mirko@isep.fr

Introduction

Programs for symbolic and semisymbolic analyses enable us to obtain results of electrical circuit analysis in the form of **mathematical formulae**. From these formulae it is possible – among others – to gain numerical and graphical information, which is also generated by common numerical circuit simulators such as Electronics Workbench *(MultiSim)*, PSpice, MicroCap, etc. However, our results contain something more, which the teacher can utilize for his explication to be more effective and the student or designer for a deeper understanding of phenomena in the circuit analyzed.

Classical numerical simulators offer only quantitative results, mostly in the form of graphs. The student acquires these results immediately after defining the circuit model, the analysis requirements, and the program starting. In other words, no subresults are available that would help him to understand **why** the results are as they are.

Programs for symbolic and semisymbolic analyses can mostly provide this quantitative graphical information too. However, they also generate the essential subresults, i.e. analytical equations. From these equations, important connections between the circuit and its behaviour can be read, e.g.

- *Which of the elements of amplifier and which of the transistor parameters influence the AC gain,*
- *What has to be fulfilled to maintain steady oscillations in the oscillator, and which elements influence the frequency,*
- *What are the equilibrium conditions of a concrete AC impedance bridge,*
- *Which parameters of the operational amplifier must be „watched" for an active filter to have the required frequency response,*
- *What is the optimum value of neutralization capacitance in an RF amplifier,*

etc.

In addition, the symbolic programs can be used to verify principles of a given circuit, especially in the area of synthetic elements (L, FDNR, ..), frequency filters, linear circuits with OpAmps (Operational Amplifiers), current conveyors, and other active elements. As an example, let us mention the verification that:

- *the circuit analyzed transforms capacitance from the output port into inductance to the input port; the dependence of the inductance and its serial resistance on the circuit elements is apparent from the symbolic result;*
- *the circuit analyzed, which contains two OTA's (Operational Transconductance Amplifiers), represents a 2nd-order bandpass filter, where the resonance frequency depends on the geometrical average of transconductances;*

The effect of parasitic elements and real properties of components on the circuit parameters can be investigated very efficiently, e.g. that:

- *in the transfer function of the lowpass „Sallen-Key" filter, the finite OpAmp GBW (Gain Bandwidth Product) along with its nonzero output resistance causes additional complex conjugate zeros and a real pole, and modifies the current complex poles; the results consist in the shift of cutoff frequency and a disagreeable modification of frequency response in the stopband region;*
- *if the transistor B-E capacitance exceeds a value of $2x10^{-14}$ h21e, the amplifier starts to oscillate;*

Using the classical simulators, these results can be gained either in no case at all, or in rare cases only by labourious step-by-step analysis based on the „try and look“ method.

Basically, prospective user of symbolic or semisymbolic programs has the following possibilities:

- *he utilizes one of the program packages for scientific computations, which enables symbolic computations, e.g. MAPLE, MATHCAD, MATLAB with the symbolic toolbox, MATHEMATICA, etc.*
- *he utilizes one of the symbolic programs specially developed for the solution of electrical circuits.*

The first approach is advantageous due to considerable features of the above universal programs which user can utilize. However, the fact that these programs have not been made specifically for the solution of electrical problems can be a source of problems: They do not offer such options as the building or utilization of existing libraries of electrical components, effective data entering and circuit modification, etc.

If the teacher (or student) decides to use a symbolic analyzer of electrical circuits, he has not much choice. The TINA and MultiSIM *(Electronics Workbench v. 6)* numeric programs offer symbolic analysis as one of the types of analysis. Because of the numeric core of these programs, the symbolic and semisymbolic analyses have some limitations in certain respects. Furthermore, these programs do not belong to the low-cost products and their demo-versions do not enable analysis of more complicated circuits. The ISAAC program [1] from Katholieke Universiteit Leuven enables so-called symbolic simplification, but it is executable only on power workstations. It is designed for the solution of large circuits. The computation method leads to time-expensive computation. Other programs available are *COCO* [2], SAPWIN [3], *LTP2* [4], and SNAP [5], [6], which represent a certain compromise between computation speed and analysis performance. What is very important is that these programs have been developed by teachers.

The Analog Insydes/Macsyma program [7] by R. Sommer et al. belongs to the top products in the category of symbolic programs for circuit analysis. This software works on the basis of the SINGULAR algebraic computational system [8], a sophisticated system for extremely fast computation of complicated polynomial equations (developed by Greuelem et al., Center for Computer Algebra, University of Kaiserslautern). This fact along with the cost are a hindrance to its common utilization at universities. Analog Insydes is now available also as a toolbox of MATHEMATICA program.

All the illustrative symbolic computations and demonstrations below have been generated by the SNAP *v. 2.6.* program.

Numerical versus symbolic result

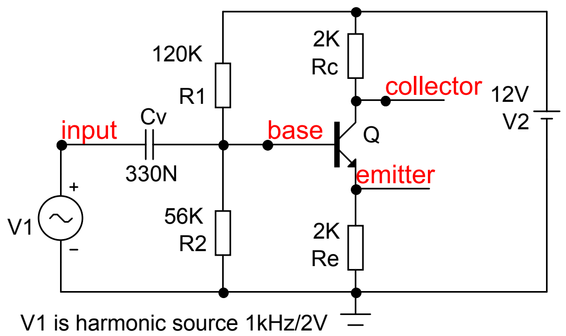

Fig. 1. Transistor amplifier analyzed.

Consider transistor amplifier in Fig. 1. By means of a classical circuit simulator (MicroCap VI in our case) one can find a set of numerical circuit parameters, e.g. the DC operating point:

Vcollector = 9.179V, Vemitter = 2.837V, Vbase = 3.509V, Icollector = 1.41mA.

An experienced program user – after some effort - can also obtain various low-frequency two-port transistor parameters in the given operating point, for example

h11e = 5kΩ (AC input resistance, short output),
h21e = 500 (current gain, short output),
h12e ≈ 0, h22e ≈0.

These parameters together with resistances Rc and Re determine the AC voltage gain. But according to which rules? If a student uses a classical simulator, he has only one method how to follow these rules: he performs a number of analyses of the gain for a set of observed parameters. Then he compares and evaluates the results. However, it is obvious that this procedure is cumbersome and that the results will be poor.

Let us use a symbolic analyzer. To compute only AC parameters for the medium working frequencies, it is sufficient to start from the simplified schematic according to Fig. 5: The DC supply is shorted as well as the coupling capacitor and resistors R1 and R2 are left out because they do not affect the AC gain.

The input and the output gates are denoted by elements of types “In” and “Out”. When considering the output at emitter (emitter follower), it is necessary to connect element “Out” between the emitter and the common line.

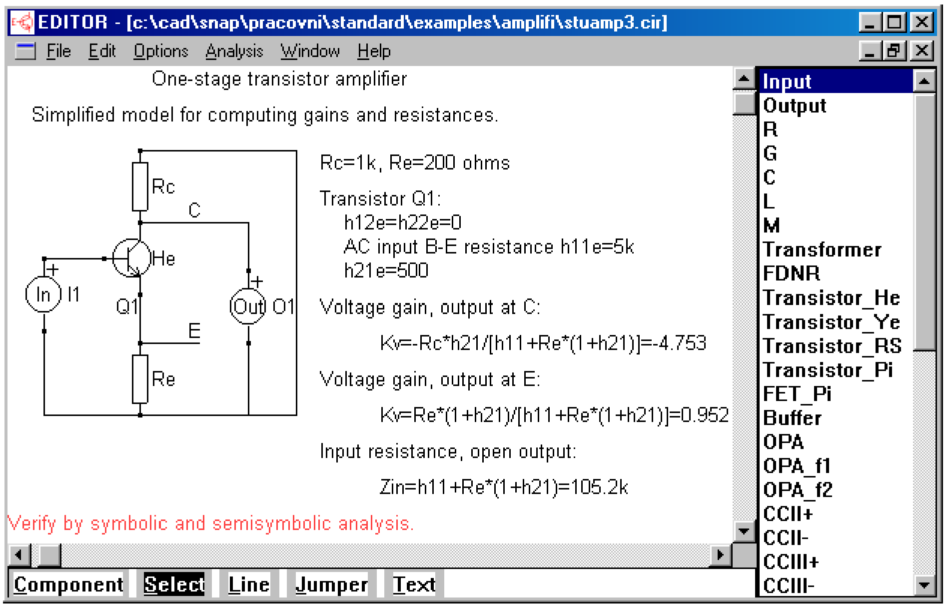

Fig. 2. Simplified schematic of the amplifier in Fig. 1 for purposes of AC analysis by the symbolic method.

The symbolic analyzer generates directly the gain formulae.

If <u>output at collector</u>:

program output	*interpretation*
______________symbolic__________________ -Rc*h21e -------------------------------------- h11e +Re +Re*h21e	$K_V = -\dfrac{R_c h_{21e}}{h_{11e} + R_e + R_e h_{21e}}$

If <u>output at emitter</u>:

program output	*interpretation*
______________symbolic__________________ Re +Re*h21e -------------------------------------- h11e +Re +Re*h21e	$K_V = \dfrac{R_e + R_e h_{21e}}{h_{11e} + R_e + R_e h_{21e}}$

One can see from the equations how the individual circuit parameters Rc, Re, h21e, and h11e participate in the formation of gain. Moreover, to perform experiments, what happens, if we change this and the other, now we do not need a complex numerical simulator, which solves a large set of equations during each simulation run.

For example, if the student is interested in the input or output AC amplifier resistance, the symbolic program generates formulae known from the theory of transistor amplifiers:

<u>Input resistance</u>:

h11 +Re +Re*h21

1

<u>Output resistance, output at collector</u>:

Rc*h11 +Rc*Re +Rc*Re*h21

----

h11 +Re +Re*h21

Output resistance, output at emitter:

Re*h11
----
h11 +Re +Re*h21

Symbolic, semisymbolic, and numerical results

Consider an oscillator with a T-cell and one OpAmp according to Fig. 3.

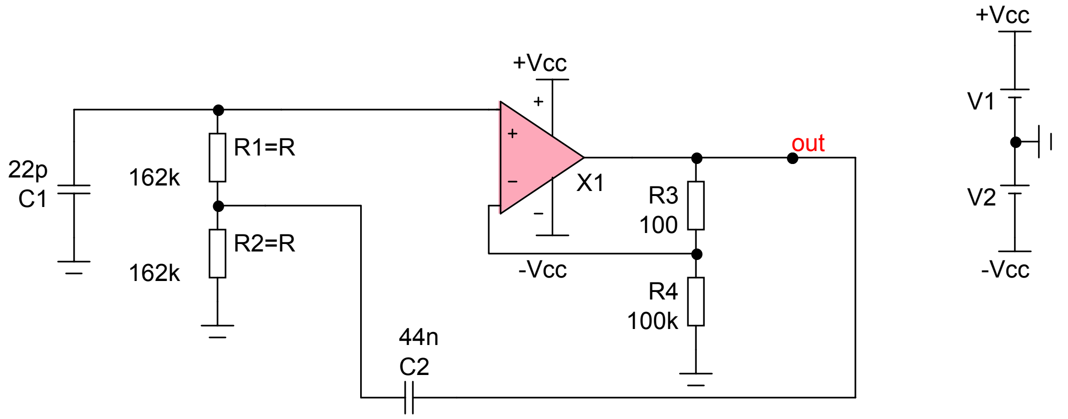

Fig. 3. Oscillator with T-cell.

There is no nonlinear or inertial element for amplitude control in the circuit. That is why the simulation of such a circuit by means of a classical simulator is a complicated problem. Oscillations will start only for certain values of circuit elements that occur in the “close vicinity” of the values in the schematic. The frequency of oscillation is about 1kHz.

By means of the symbolic analyzer, the student can exactly derive both the oscillation condition and the formula of oscillation frequency. We compute the open loop gain and then set it to one. The gain formula is as follows:

program output	*interpretation*
s*(R*C2*R4 +R*C2*R3) ------------------------------------- R4 +s*(R*C2*R4 +2*C1*R*R4) +s^(2)*(C1*R^(2)*C2*R4)	$\frac{s(R_3+R_4)RC_2}{R_4+s(C_2+2C_1)RR_4+s^2R^2R_4C_1C_2}$

The symbol s indicates complex frequency $s = j\omega$.
Utilizing this substitution, the quation

$$\text{gain} = 1$$

can be arranged to the following form:

$$j\omega(C_2R_3-2C_1R_4)R = R_4-\omega^2R^2R_4C_1C_2.$$

This complex equation falls into two real equations - oscillation conditions:

$\omega = \frac{1}{R\sqrt{C_1C_2}}$ or $f = \frac{1}{2\pi R\sqrt{C_1C_2}}$ .. formula of the oscillation frequency

$\frac{C_2}{C_1} = 2\frac{R_4}{R_3}$.. condition of steady state oscillation with frequency f.

Substituting numerical values from the schematic leads to the conclusion that the condition of steady state oscillation is fulfilled exactly. The frequency is about 998.5 Hz.

The program can also generate results in so-called semisymbolic form: instead of symbols we set numerical values of circuit parameters. Then the only symbol will be complex frequency *s*:

program output	*interpretation*
____________semisymbolic______________ Multip. Coeff. = 2.80864197530864E+0005 1.00000000000000E+0000 * s ------------------------------------- 3.93635821993473E+0007 2.80864197530864E+0005 * s 1.00000000000000E+0000 * s^(2)	$\dfrac{2.808e5\, s}{3.936e7 + 2.809e5\, s + s^2}$

We can readily verify that the maximum open loop gain is exactly on the frequency of 1kHz, and that the extreme value is 1. The symbolic program can help us to demonstrate this to students if it also includes numerical analysis of semisymbolic expressions.

A number of important facts result from the formulas of oscillating frequency and of steady state oscillation, that cannot be obtained directly from the classical numerical simulators. For example, with the R4/R3 ratio growing above a given level, the open loop gain falls below 1 and the oscillations die out. Resistance R is not included in the condition of steady state oscillation. That is why oscillating frequency can be tuned by a double potentiometer R1-R2, etc., etc.

Conclusion

Symbolic and semisymbolic circuit analyzers operate on a quite different principle than conventional numerical simulators. In the paper, we explain some pedagogical and technical advantages of these programs and show how to use them in the teaching process.

References

[1] GIELEN,G.-WAMBACQ,P.-SANSEN,W.: ISAAC version 4.0 – User Manual. Katholieke Univ. Leuven, Belgium, 1991.

[2] CAJKA,J.-VRBA,K.-BIOLEK,D.: Improved program for symbolic analysis and synthesis of linear circuits. AMSE'94, Lyon, Vol.1, pp. 299-302, 1994, France.

[3] FEDI,G.-GIOMI,R.- LUCHETTA,A.- ANETTI,S.-PICCIRILLI, M.C.: Sapwin 2.0: a symbolic software tool for educational purposes in analysis and synthesis of analog circuits, ICSEE'99, San Francisco, Ca-USA, January 1999.

[4] VALSA,J.: LTP – A Tutorial Program for Analysis of Linear Two-Ports. Radioengineering, Vol. 2, December 1993, No. 4, pp. 5-8.

[5] BIOLEK,D.-KOLKA,Z.: SNAP program v. 2.6: New Opportunities for teaching and research. Proceedings of the Conference on Circuit Theory STO-7, „*Modern Trends in Teaching Electrical Engineering and Electronics*“, VA Brno, 1999, pp. 66-69. Czech Republic.

[6] BIOLEK,D.-KOLKA,Z.: SNAP: A Tool for the Analysis and Optimization of Analogue Filters. TSP'99, Brno, 1999, pp.60-63.

[7] SOMMER,R.-HENNIG,E.: A Sizing Strategy for Linear Transistor Amplifiers. In *Proc. 3rd International Workshop on Symbolic Methods and Applications to Circuit Design*, Sevilla, Oct. 1994

[8] Macsyma, Inc., *Macsyma Reference Manual V14*, Arlington, 1993.

[9] BIOLEK,D.: SNAP v. 2.6. Getting started. 40 pages, Brno 1999. Czech Republic.

[10] http://www.fee.vutbr.cz/UREL/snap