Automated CAD Tool for Design of Nullor-based Amplifiers

Roberto Castañeda-Sheissa, Arturo Sarmiento-Reyes, Luis Hernández-Martínez
National Institute for Astrophysics, Optics and Electronics
Electronics Department, CAD Group
P.O. Box 51, 72000, Puebla, Pue., Mexico

Abstract

Computer aided design (CAD) tools are aimed to speed-up processes, simplify calculations and perform complex tasks with minimum effort. CAD tools are present in many fields like chemistry, physics, mathematics, astrophysics, engineering and electronics. For electronics, CAD tools are developed mainly to perform verification tasks and, because the complexity it implies, there is a lack of electronic design tools. By means of a design methodology known as Structured Design, it is possible to develop a CAD tool. This work presents the development of a CAD tool for automated design of amplifiers based on the guidelines provided by Structured Design and programmed in C++.

I. Introduction

The production of any electronic circuit is concerned with successively taking a requirement; developing it; producing an accurate abstract representation (a circuit diagram); making, evaluating, and testing it; and then returning to the original circuit design to correct errors. This process is iterative, frequently extending across the lifetime of the circuit, with modifications and improvements introduced and specifications changed continuously. This way to perform a development involves the construction and manipulation of various (often very large) representations of a circuit and the examination and testing of these with respect to the initial requirements. The development of any sub-circuit representation for running a test can be long and complicated. Performing a complete set of these design activities for a system of any size is long and, specially, costly. The computer's ability to contain large amounts of data and to facilitate its access and updating in a dynamic way, together of course with powerful programs for aiding in the synthesis and analysis of circuits, is a major for their use.

With each new level of refinement, more information concerned with the detailed physical and functional implementation of their circuit is included in the description. The design process involves transformation between these representations during both synthesis and verification. There are many programs designed to produce one representation of the design process and the output given to another tool for further post-processing. However for many of these tasks, programs are not perfect and much human intervention is still needed. For large designs, this is very expensive because more than one product iteration is frequently required to remove all errors before a working design is produced.

The term CAD *framework* has come to mean all of the underlying facilities provided to the CAD tool *developer*, the CAD system *integrator*, and the *end user* (IC or system designer) which are necessary to facilitate their tasks.

Broadly speaking, these three groups of people represent the user of the CAD framework, each with their own needs and particular emphasis. The CAD framework plays an analogous role in the development of engineering-specific, or even electrical-engineering-specific, software systems to the role played by an operating system for the development of general-purpose software applications, or the role of a specific programming environment for software development in a particular programming language [1].

II. Electronic Structured Design

Electronic circuit design consists on a search through many combinations with several kind of components and different kind of properties. It would be a great time investment to find an apt circuit by means of extensive tests with every possible combination of available components. Therefore it is necessary to stablish a strategy to find the right solution as soon as possible. The most common strategy for circuit design is to use certain circuit topology, then by changing some parameters within the circuit a solution that fulfils the requirements of the designer is found [2]. The way that this new design behaves within certain constraints, is going to stablish the validity of it. In case that the design is not useful, the design process is restarted. Even though that this strategy provides valid results sooner or later, there are serious disadvantages on it:

- There is no guarantee that the found solution is the *optimum* and does not provide any information on how close it is.
- Relationship between component parameters and circuit performance is never explicit. Incidentally, circuit performance can be sensitive for certain irrelevant parameter because the circuit is not suitable for the application.
- It is really complicated to determine what it should be changed in a circuit when, for instance, constraints are changed.

To create order from the chaos that the electronic amplifier design is, design is separated in smaller problems, all orthogonal if possible, which allows to clarify the problem and makes easier the solution process. *Structured electronic design* [2], [3] is a method that allows find just **ONE** solution to the design problem in a quick way. Nevertheless this is a method based on certain amount of suppositions and a limited amount of rules. Therefore, the practical application of the electronic structured design should never, and by any means, create a dogmatic reject for the given results by the traditional design strategy or any other strategy provided that this would incite a delay on the evolution of the design strategy.

Amplifier structured design focuses on three fundamental aspects in order to provide the description on how the circuit behaves:

- Noise.
- Distortion.
- Bandwidth.

In order to accelerate the design process, several suppositions should be established as stated previously. These will be enforced based on the basic specs that should be provided, on the circuit behaviour given the results by employing simpler models and to the decisions concerning every stage of the design in particular. This kind of design is based on the following basic elements:

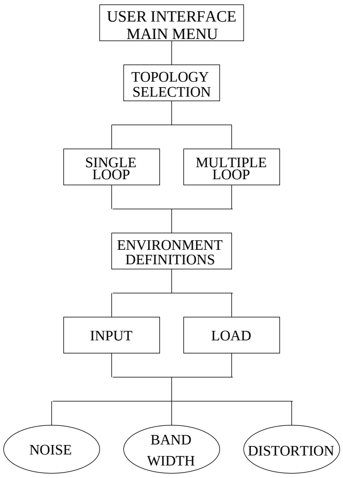
- Orthogonality - Circuits should be order in such a way that the behaviour of the the three fundamental aspects can be designed orthogonally, that is, the behaviour of a stage should not have any influence on the other two.
- Simplicity - Simple models are defined to obtain quick predictions on the feasibility of the circuit. Non feasible solutions can be detected on the first stages of the design. Special planning should be done in order to stablish that the estimated results are as close as possible to the real ones.
- Hierarchy - Design hierarchy allows to reduce the complexity of the design problem because allows its efficient division in smaller and independent design problems. Planning at this point should be arranged in such a way that decisions taken on a certain hierarchical level are kept valid throughout the rest of the design.

III. CAD Tool Structure

Given the basic blocks that structured design is based on to create an amplifier, a rather general structure for the design based on these blocks is shown in Figure 1. This tool has been developed under the code name *Descad_Wizard* to reflect that the adequate way to operate this tool is using a *wizard* approach, that is, development process is performed by filling up some options within a window. Calculations are programmed in C++ [4] and the graphical user interface is provided by Qt [5].

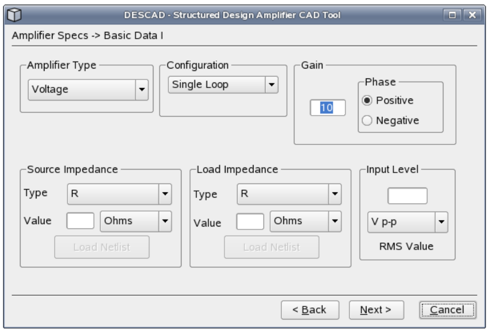
Figure 2 shows the first window where basic specifications are typed. Basic requirements refers to the basic amplifier structure like:

- Amplifier types.
- Amplifier configuration.
- Source and load types and values.
- Gain and sign specs.



Obtener la corriente de polarizacion adecuada

Fig. 1. General structure for the CAD tool.


DESCAD - Structured Design Amplifier CAD Tool

Amplifier Specs -> Basic Data I

Amplifier Type: Voltage

Configuration: Single Loop

Gain: 10

Phase: ● Positive ○ Negative

Source Impedance: Type R; Value ; Ohms

Load Impedance: Type R; Value ; Ohms

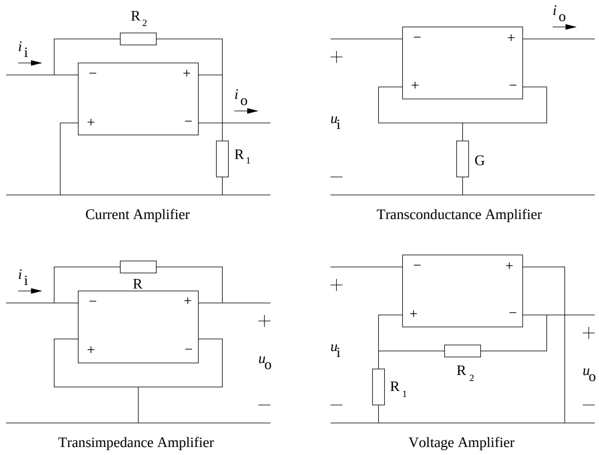
Input Level: ; V p-p; RMS Value

Fig. 2. Basic specifications window.

The gain value can be provided as a floating-point number or in scientific format; phase value refers to the output phase that is desired at the output port of the amplifier. Phase can be positive (no phase change) or negative (phase is shifted 180 ° away). Another specification to be provided is the desired configuration, it could be one of two options. First option is to select one of the single-loop amplifiers: (1) voltage amplifier, (2) transconductance amplifier, (3) transresistance amplifier and (4) current amplifier. These amplifiers are depicted in Figure 3. Second option is a two-loop amplifier. This type of amplifier comprises the transconductance and transresistance amplifiers. Figure 4.

For the synthesis of noise, clipping and bandwidth stages a scheme is provided in Figure 5. It means that for every stage it is possible to choose a single device or differential (two devices) configurations; it is worth noting that in order to keep the negative feedback behaviour a differential configuration must be placed either on noise or clipping stages. Windows for the noise and clipping stages are depicted in Figure 6 and Figure 7.

Bandwidth synthesis process is divided in two steps: (1) calculate the LP-product [2] in order to verify if design is capable to handle the desire bandwidth and (2) in order to guarantee a flat output frequency response, amplifier poles must be located at Butterworth locations [3]. In case that the LP-product is not high enough for the desired band-



Current Amplifier

Transconductance Amplifier

Transimpedance Amplifier

Voltage Amplifier

Fig. 3. Amplifier types.

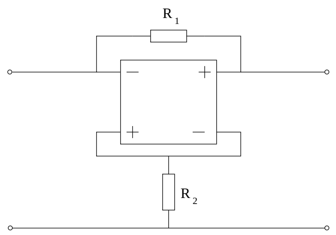


Fig. 4. Two-loop amplifier topology.

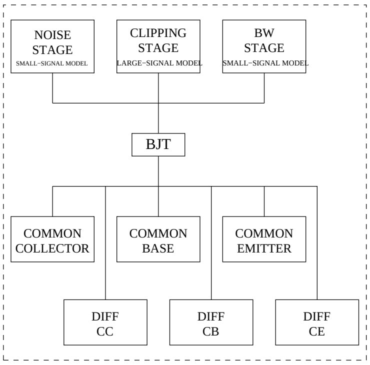
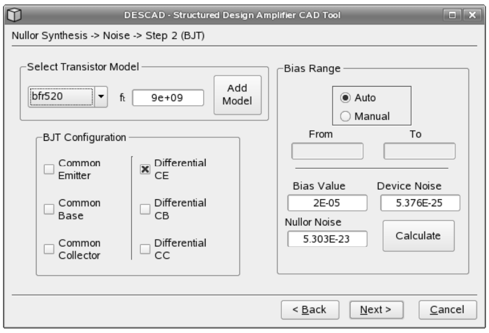


Fig. 5. Noise, clipping and bandwidth synthesis scheme.


DESCAD - Structured Design Amplifier CAD Tool

Nullor Synthesis -> Noise -> Step 2 (BJT)

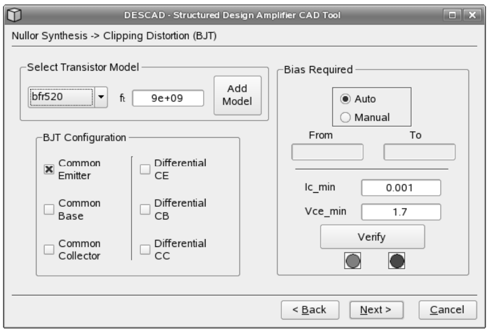
Select Transistor Model: bfr520 ft 9e+09

BJT Configuration:
□ Common Emitter
□ Common Base
□ Common Collector
☒ Differential CE
□ Differential CB
□ Differential CC

Bias Range:
● Auto
○ Manual
From To

Bias Value 2E-05
Device Noise 5.376E-25
Nullor Noise 5.303E-23

Fig. 6. Noise stage window.


DESCAD - Structured Design Amplifier CAD Tool

Nullor Synthesis -> Clipping Distortion (BJT)

Select Transistor Model: bfr520 ft 9e+09

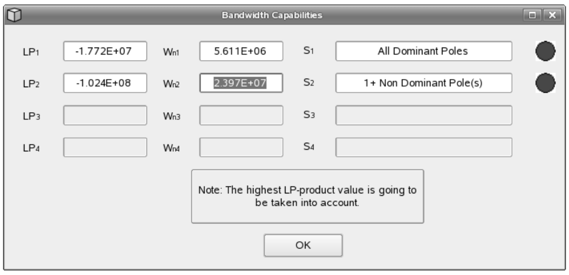
BJT Configuration:
☒ Common Emitter
□ Common Base
□ Common Collector
□ Differential CE
□ Differential CB
□ Differential CC

Bias Required:
● Auto
○ Manual
From To

Ic_min 0.001
Vce_min 1.7

Fig. 7. Clipping stage window.

width, there are three options to be applied in this order: (1) increase collector current for the noise stage as high as the noise behaviour is kept within constraints. (2) If the LP-product is still low, an increase on the collector current for the clipping stage is performed. This increase can only be applied if clipping is below the allowed value. (3) Last case is to add another device to the design increasing the amplifier degree by 1. Figure 8 shows the LP-product window.

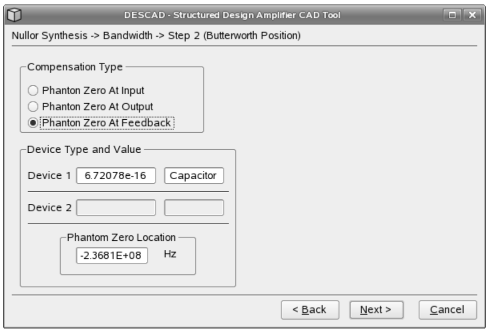

Bandwidth Capabilities

LP1	-1.772E+07	Wn1	5.611E+06	S1	All Dominant Poles
LP2	-1.024E+08	Wn2	2.397E+07	S2	1+ Non Dominant Pole(s)
LP3		Wn3		S3	
LP4		Wn4		S4	

Note: The highest LP-product value is going to be taken into account.

Fig. 8. The LP-product window.

Once the LP-product is high enough the process to place the closed-loop poles of the amplifier in Butterworth position is performed. [2] and [3] provide the adequate compensation techniques in order to place the amplifier poles at the required location. Given the fact that it is possible to perform compensation at the input port, output port and feedback network the user is capable to select where to perform the compensation. Nevertheless if compensation fails to adjust the poles position, a warning message is displayed and the option is disabled then the user must select any available option. Figure 9.


DESCAD - Structured Design Amplifier CAD Tool

Nullor Synthesis -> Bandwidth -> Step 2 (Butterworth Position)

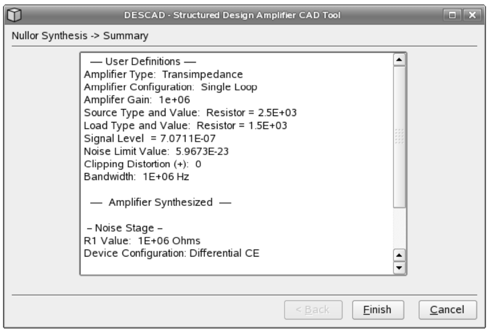
Compensation Type:
○ Phanton Zero At Input
○ Phanton Zero At Output
● Phanton Zero At Feedback

Device Type and Value:
Device 1 6.72078e-16 Capacitor
Device 2

Phantom Zero Location: -2.3681E+08 Hz

Fig. 9. Butterworth compensation stage.

Finally a summary is displayed providing the constraints given by the user and the results of the synthesis procedure. Besides an input file for an industrial simulator is generated. Figure 10.



Nullor Synthesis -> Summary

```
 — User Definitions —
Amplifier Type:  Transimpedance
Amplifier Configuration:  Single Loop
Amplifer Gain:  1e+06
Source Type and Value:  Resistor = 2.5E+03
Load Type and Value:  Resistor = 1.5E+03
Signal Level  = 7.0711E-07
Noise Limit Value:  5.9673E-23
Clipping Distortion (+):  0
Bandwidth:  1E+06 Hz

 —  Amplifier Synthesized  —

 – Noise Stage –
R1 Value:  1E+06 Ohms
Device Configuration: Differential CE
```

Fig. 10. Design summary window.

IV. Example

As an example a transconductance amplifier is going to be designed and must satisfy these constraints:

- Source Type and Value: Resistor - $5K\Omega$
- Load Type and Value: Resistor - $1.5K\Omega$
- Input Level: .5 μA
- Output Level: .5 V
- Noise Level: 10 dB
- Clipping: 0 %
- Bandwidth: 1 MHz

Noise values calculated for the feedback and source impedances:

- Source Noise: 3.315E-24 A^2/Hz
- Feedback Noise: 1.658E-26 A^2/Hz

The nullor synthesis for the noise stage is performed with the transistor BFR520 [6] in differential common emitter configuration. The minimum I_C and noise values are:

- $I_C = 2E - 5\ A$
- Noise=1.351E-25 A^2/Hz

Clipping distortion stage synthesis is provided by the transistor BFR520 in common emitter configuration. Values are:

- $I_C = 0.001\ A$
- $V_{CE} = 1.7$ V

Note that values on the clipping stage are compared to the maximum allowable values taken from the device datasheet. For the bandwidth stage and the LP-product calculation process on the first run the result was that the LP-product is high enough but had one or more non dominant poles. This behaviour is not desired because affects the overall gain, consequently an adjustment is required. The adjustment is performed on the noise stage because it has the best I_C range. Consequently the new values for the noise stage are:

- $I_C = 0.001\ A$
- Noise=3.226E-26 A^2/Hz

Hence for the Butterworth compensation process it is performed at the feedback network. In this case a capacitor is placed in parallel to the feedback resistor and its value is $C = 9.50872E - 16$F.

Simulation on the amplifier circuit is performed using HSpice [7]. Instead of using the actual active device, small-signal elements are employed. Small-signal values are calculated using the I_C found for each synthesised stage. Figure 11 shows the resulting. As it can be seen a flat output behaviour is achieved on the desired frequency.

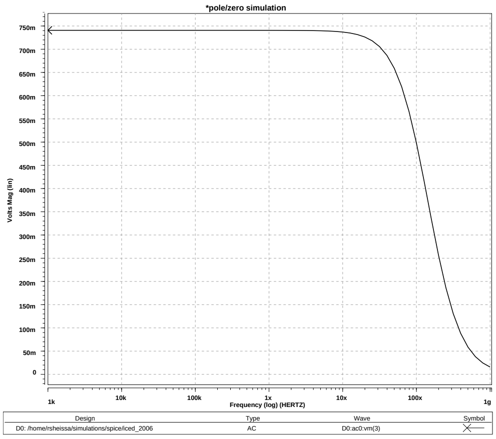


Fig. 11. Amplifier simulation output.

V. Conclusions

An automated design tool for nullor-based amplifiers has been presented. Based on the *Structured Design* methodology and with the aid of the C++ programming language, this CAD tool is capable to speed-up the design process. A *wizard* driven tool helps the user to make easier decisions and concentrate only on certain aspects of the design while calculations are performed without user intervention. Results given by the tool can be simulated for verification purposes using the small-signal equivalent of their components. Consequently simulation can be performed directly on the AC domain without the worry about DC concerns.

VI. References

[1] Timothy J. Barnes, *Electronic CAD Frameworks*, Kluwer Academic Publishers, 1992.
[2] C.J.M. Verhoeven, A.van Staveren, G.L.E. Monna, and M.H.L. Kouwenhoven, *Strutured Electronic Design, Negative-feedback Amplifiers*, Delftese Uitgevers Maatschappij, 2001.
[3] E.H. Nordholt, *Design of High Performance Negative-feedback amplifiers*, Elsevier Scientific, 1983.
[4] L. Joyanes-Aguilar, *Borland C++*, McGraw-Hill, 1993.
[5] Trolltech, Oslo, Norway, *Qt Reference Documentation*, 2005, Ver. 3.3.4.
[6] Philips Electronics, *BFR520 Product data sheet*, September 2004.
[7] Avant! Software, *Star-Hspice Manual*, 2001, Release 2001.2.