Formal Verification of Continuous Models of Analog Circuits

Syeda Hira Taqdees
School of Electrical Engineering and Computer Science,
National University of Sciences and Technology (NUST),
H-12, Islamabad 44000, Pakistan
Email: 11mseestaqdees@seecs.nust.edu.pk

Osman Hasan
School of Electrical Engineering and Computer Science,
National University of Sciences and Technology (NUST),
H-12, Islamabad 44000, Pakistan
Email: osman.hasan@seecs.nust.edu.pk

I. Introduction

The goal of verifying an analog circuit is to make sure that the implementation of the circuit exhibits the desired behavior. Traditionally, analog circuits are analyzed using simulation techniques. However, simulation results cannot be termed as 100% accurate due to the approximations introduced by using computer arithmetics, such as floating or fixed point numbers, for constructing computer based models of the continuous analog circuits. Moreover, the circuits are analyzed for some specific test cases only since exhaustive simulation is not possible due to the continuous nature of inputs. Due to these limitations, more rigorous and accurate analysis techniques for analyzing analog circuits are actively sought and formal verification., i.e., a computer based mathematical analysis technique, offers a promising solution [5]. The rigorous exercise of developing a mathematical model for the given system and analyzing this model using mathematical reasoning usually increases the chances for catching subtle but critical design errors that are often ignored by simulation.

However, to the best of our knowledge, all the existing formal verification approaches work with abstracted discretized models of analog circuits (e.g., [3],[2]). This is mainly because of the inability to model and analyze continuous systems by the widely used formal verification techniques, such as model checking or automated theorem proving. Thus, despite the inherent soundness of formal verification methods, such analysis cannot be termed as absolutely accurate.

We propose to use higher-order-logic theorem proving in order to formally verify continuous models of analog circuits. Higher-order logic is a system of deduction with a precise semantics and, due to its high expressiveness, can be used to describe any mathematical relationship. We argue that the high expressibility of higher-order logic can be leveraged upon to formalize the continuous models of analog circuit implementations and their desired specifications. Their equivalence can then be verified within the sound core of a theorem prover. Due to the high expressibility of higher-order logic, the proposed approach is very flexible in terms of analyzing a variety of analog circuits and reasoning about their generic properties.

There are two main challenges in the proposed approach. Firstly, due to the undecidable nature of higher-order logic, the proofs have to be done interactively, which may become very tedious due to the involvement of continuous elements and transcendental functions. Secondly, no closed form solutions exist for a large number of analog circuits and thus for these kinds of circuits, we cannot formally reason about approximate solutions in a theorem prover. We overcome both of these challenges in the proposed methodology, depicted in Figure 1, by developing a library of analog circuit analysis definitions, theorems and automatic simplifiers to minimize the user effort in the formal reasoning process and by using the support of computer algebra system for solving differential equations for which no closed form solutions exist.

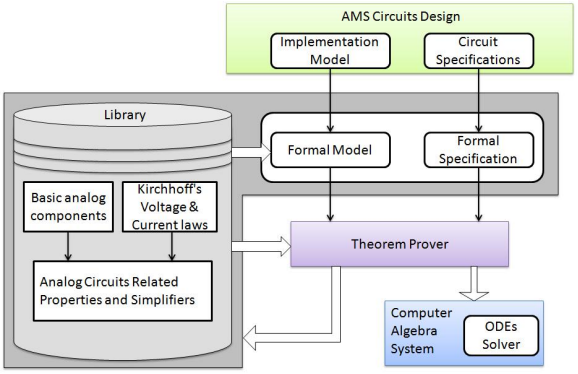

Fig. 1. Proposed Methodology for the Formal Verification of AMS circuits

The first step in the proposed methodology is to obtain an implementation model of the given analog circuit by using the behavior of its individual components and its overall structure. To facilitate this formalization, we developed a database of formal definitions of commonly used analog components, like resisters, capacitors and inductors, and Kirchhoffs voltage and current laws. The second step in the proposed methodology is to develop a formal model of the specification of the circuit, which is usually expressed as a differential equation. For this purpose, we choose the HOL4 theorem prover to implement the proposed methodology since it provides formalized libraries of real numbers and Calculus foundations [4]. The third step is to verify the equivalence or implication relationships between the formalized implementations and specifications.

To minimize the user interaction, required in this step, we formally verified most of the frequently used properties and developed some simplifying tactics with access to these results so that they can automatically verify most of the proof goals associated with analog circuit verification. Finally, once the differential equation corresponding to the given analog circuit is verified, it can be fed to a computer algebra system, like Mathematica, to obtain an approximate solution for it. It is important to note here that the soundness of the analysis is not compromised at all by the computer algebra system link since it would only be invoked for the cases where a closed-form precise solution cannot be attained.

II. Foundations of Analog Circuits

We now present some of the foundational formalization that is required to formally model analog circuits. The V-I characteristics of fundamental analog components like resistors, inductors, capacitors and op-amps can be formalized as:

Definition 1: *Resistor, Inductor, Capacitor and Op-amp*

```
⊢ ∀ R i.resistor_voltage R i = (λt.i t * R)
⊢ ∀ L i.inductor_voltage L i =
    (λt. L * deriv i t)
⊢ ∀ C i Vo. capacitor_voltage C i Vo =
    (λt. Vo + 1/C * integral (0,t) i)
⊢ ∀ Vpos Vneg A. op_amp_voltage Vpos Vneg A=
    (λt. A * (Vpos t - Vneg t))
```

The variable i represents the time dependant current. While the variables R, L and C represent the constant resistance, inductance and the capacitance of their respective components, respectively. The variable Vo is used to model the initial voltage across the capacitor. All these functions return a (`real` $\rightarrow$ `real`) type function that models the corresponding time dependant voltage. The function `deriv` accepts two parameters f and x and returns the derivative of the function f at point x. Likewise, the function `integral` takes three parameters f, a and b and returns the integrated result of f in the interval (a,b). The parameters `Vpos`, `Vneg` and `A` represent non-inverting input, inverting input and gain of an op-Amp, respectively.

Kirchhoffs current law (KCL) can be formalized as

Definition 2: *Kirchhoff's Current Law*

```
⊢ ∀ V t. kcl I t =
  (∀ x. 0 < x ∧ x < t ⇒
    (sum (0,LENGTH I) (λn. EL n I x) = 0))
```

The function `kcl` accepts a list I of functions of type (`real` $\rightarrow$ `real`), which represents the behavior of time-dependant currents in the given circuit and a time variable t as a $real$ number. It returns the predicate that guarantees that the sum of all the currents at a node is zero for all time instants in the interval $(0, t)$.

III. Application: Delta Sigma Modulator

In order to illustrate the proposed methodology, we present the formal verification of the first order delta sigma modulator, shown in Figure 2, which is widely used benchmark in formal verification of analog circuits (e.g. [1]).

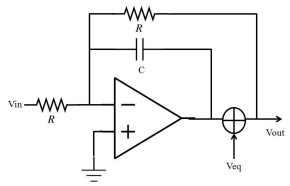

Fig. 2. First Order Delta Sigma modulator

The implementation model of this circuit can be obtained by applying KCL function at the input node of the op-amp:

Definition 3: *Implementation Model of Delta-Sigma Modulator*

```
⊢ ∀ R C Vin Vout Vc Veq y.
delta_sigma_imp R C Vin Vout Vc Veq y =
  (kcl [resistor_current R Vin;
        resistor_current R Vout;
        capacitor_current C (λx. -Vc x)] t)∧
  (Vout = (λt. Veq t - Vc t))
```

The next step is to formalize its specification:

Definition 4: *Behavioral Model of Delta-Sigma Modulator*

```
⊢ ∀ R C Vin Vout Veq y.
delta_sigma_behav R C Vin Vout Veq y =
  (diff_eq [1; R * C] Vout y =
  -Vin y + diff_eq [0; R * C] Veq y)
```

The function `diff_eq` accepts the list of coefficients of a differential equation, the differentiable function and the differentiation variable and returns the corresponding differential equation.

Next, we formally verified the following implication between the implementation and specification of the given first order delta-sigma modulator.

Theorem 1: *Implementation implies Specification*

```
⊢ ∀ R C Vin Vc Vout Veq t .
  delta_sigma_imp R C Vin Vc Vout Veq t⇒
  delta_sigma_behav R C Vin Vout Veq t
```

The proof was automatic due to the available formally verified properties and simplifiers, whose implementation details are ignored due to limited space. Based on the result of Theorem 1, we can now feed the differential equation of Definition 4 to a computer algebra system to obtain its solution and other interesting characteristics of the delta-sigma modulator.

References

[1] Ghiath Al-Sammane, Mohamed H. Zaki, and Sofiéne Tahar. A Symbolic Methodology for the Verification of Analog and Mixed Signal Designs. In *IEEE/ACM Design Automation and Test in Europe*, pages 249–254, 2007.

[2] W. Denman, B. Akbarpour, S. Tahar, M. Zaki, and L. C. Paulson. Formal Verification of Analog Designs using MetiTarski. In *Formal Methods in Computer Aided Design*, pages 93–100. IEEE, 2009.

[3] G. Frehse, C. Le Guernic, A. Donzé, S. Cotton, R. Ray, O. Lebeltel, R. Ripado, A. Girard, T. Dang, and O. Maler. Spaceex: Scalable Verification of Hybrid Systems. In *Computer Aided Verification*, volume 6806 of *LNCS*, pages 379–395. Springer, 2011.

[4] J. Harrison. *Theorem Proving with the Real Numbers*. Springer-Verlag, 1998.

[5] Mohamed H.Zaki, Sofiéne Tahar, and Guy Boisr. Formal Verification of Analog and Mixed Signal Designs: A survey. In *Microelectronics Journal*, volume 39, pages 1395–1404. Elsevier, 2008.