Diagnosing Analog Circuits Designed-for-Testability by Using CLP($\Re$)*

Igor Mozetič
Austrian Research Institute for Artificial Intelligence
Schottengasse 3, A-1010 Vienna, Austria
email: igor@ai.univie.ac.at

Franc Novak
Jožef Stefan Institute
Jamova 39, 61111 Ljubljana, Slovenia
email: franc.novak@ijs.si

Marina Santo-Zarnik
Iskra HIPOT, Šentjernej, Slovenia

Anton Biasizzo
Jožef Stefan Institute
Jamova 39, 61111 Ljubljana, Slovenia

Abstract

Recently, a design-for-test (DFT) methodology for active analog filters was proposed with the primary goal in increased controllability and observability. We operationalize and extend the DFT methodology by using CLP($\Re$) to model and diagnose analog circuits. CLP($\Re$) is a logic programming language with the capability to solve systems of linear equations and inequalities. It is well suited to model parameter tolerances and to diagnose soft faults, i.e., deviations from nominal values. The diagnostic algorithm uses different DFT test modes and voltage measurements at different frequencies to compute a set of suspected components. Ranking of suspected components is based on a measure of (normalized) standard deviations from

*This is an extended version of the paper that appears in the *Working notes Fourth Intl. Workshop on Principles of Diagnosis, DX-93*, pp. 105-120, University of Wales, Aberystwyth, September 6-8, 1993.

predicted mean values of component parameters. Presented case studies on a real circuit show encouraging results in isolation of soft faults for a given low pass biquad filter.

1 Introduction

Fault diagnosis of analog circuits has been a research area for more than two decades [IEEE, 1979, Bandler and Salama, 1985]. Although numerous, the results are not satisfactory in practice and, according to [Cavin and Hilbert, 1990], a breakthrough is needed to develop test strategies for analog and mixed analog/digital integrated circuits. Even the design of analog circuits is a challenge due to the sensitivity with respect to component variations and process technology. The number of input/output ports of an analog circuit is small compared to that of a digital circuit, but the complexity arises due to continuous signal values, and the inherent interactions between various circuit parameters. One has to model parameters with tolerances and compute with intervals. As a consequence of non-directional behavior of an analog circuit, any component can be responsible for a symptom. All of these impede efficient testing and diagnosis. As Soma [1990] pointed out: "...analog diagnosis is still more of theoretical interest rather than experimental application."

In-circuit test and functional testing are established techniques for testing both analog and digital circuits. However, guarded test configuration as the basis of in-circuit test is limited in its capacity to operate at high frequencies. Moreover, the technique becomes very difficult for complex circuits where hundreds of test pins and driver-sensor amplifiers are required [Slamani and Kaminska, 1992]. Consequently, functional testing of analog circuits is preferred.

Two general methods of analog fault diagnosis are simulation before test (SBT) and simulation after test (SAT) [Bandler and Salama, 1985]. SBT techniques, based on fault dictionary, find limited applications in practice due to the high cost of analog simulation needed for the large number of potential faults. These techniques are more convenient for the isolation of hard faults. SAT techniques use measurements to compute parameters of the network (parameter identification techniques) or locate the faulty components without computing parameter values (fault verification techniques). They are suitable for diagnosing soft faults (i.e., deviations of element parameters from the specified tolerance range). Depending on the excitation source arrangement which is used for test measurements, SAT techniques are characterized as single vector or multiple vector [Walker and Alexander, 1992]. In the first case the measurement is implemented with a single test frequency while the multiple vector technique uses multiple test frequencies for the solution of fault diagnosis equations.

The complexity of analog test fostered investigation of the design-for-test (DFT) principles for analog and mixed analog/digital circuits [Soma, 1990, Vefling and Viktil, , Fasang, 1989, Eagner and Williams, 1989, Jarwala and Tsai, 1991, Ohletz, 1991]. Most

analog circuits are designed in stages in order to localize the effect of components. By appropriate modifications to the circuit, each individual stage can be controlled, observed and tested in (relative) isolation, without intermediate measurements. In view of the complexity, the general problem of DFT for analog circuits is almost certainly unfeasable. Hence, Soma [1990] proposed a DFT methodology for a restricted class of circuits, namely active analog filters. Another aspect is to establish some general DFT strategy for mixed analog/digital circuits. In this approach, individual analog and digital parts are replaced by functional blocks and DFT is investigated at the functional level [Fasang, 1989, Eagner and Williams, 1989, Jarwala and Tsai, 1991, Ohletz, 1991].

In recent years, model-based systems for fault diagnosis are starting to gain considerable interest in engineering practice [Sierzega and Rastogi, 1990]. Unlike in digital circuits troubleshooting, it is impractical to use fault simulation and to construct diagnostic decision trees for analog circuits. Model-based approach provides a viable alternative since it uses behavioral circuit models for fault diagnosis and is also transferable. Promising results of this approach have been reported recently [McKeon and Wakeling, 1989, McKeon and Wakeling, 1990, Mozetič *et al.*, 1991].

The motivation for this work was the idea to use CLP($\Re$) for the simulation and diagnosis of analog circuits. CLP($\Re$) is a logic programming system extended with a solver for systems of linear equations and inequalities. It is well suited to model AC circuits, real-valued system parameters with tolerances, and feedback loops. First experiments [Mozetič *et al.*, 1991] showed that CLP($\Re$) has some advantages over classical simulation tools (like SPICE or Micro-CAP) since the same model can be used for both, simulation and diagnosis. CLP($\Re$) can locate potential soft faults, i.e., parameter values (resistors, capacitors) deviating from nominal values.

Some work on the diagnosis of analog systems was also done in the AI community, e.g., [de Kleer and Brown, 1992, Dague *et al.*, 1990]. The main emphasis there is on the modeling of circuit parameters with intervals, in efficient interval propagation methods, and in conflict detection using ATMS-based mechanisms. Related to our work is the fault identification procedure in SOPHIE II [de Kleer and Brown, 1992] which uses SPICE to identify component parameter shifts (soft faults) which explain the symptoms. Due to the iterative application of SPICE the procedure is computationally intractable. In contrast, CLP($\Re$) can compute soft faults deterministically and efficiently, provided that single faults are assumed.

In this paper we combine CLP($\Re$) with the DFT methodology [Soma, 1990] in order to focus on individual stages-under-test and to make the DFT methodology operational. Diagnostic algorithm implemented in CLP($\Re$) uses the results of measurements of magnitude and phase characteristics as a function of a real frequency variable in the normal mode and in the test modes. The diagnosis is performed incrementally, in each step reducing the set of potential candidates for the detected fault. The CLP($\Re$) system and measurement instrumentation have been integrated in an experimental system for automatic fault isolation. Experimental results are given to assess the efficiency of the proposed solution.

The paper is organized as follows. In section 2 we give a brief overview of the CLP($\Re$) language and illustrate how it can be used for modeling and diagnosis. The application of the node-analysis method for the justification of the CLP($\Re$) model is discussed in section 3. In section 4 the main points of the DFT methodology for active analog filters are reviewed and the fault isolation procedure is described. Experimental results are given in section 5, together with the computed diagnoses. Finally, some concluding remarks are drawn.

2 Modeling Analog Circuits with CLP($\Re$)

2.1 The CLP($\Re$) language

Constraint Logic Programming (CLP, [Jaffar and Lassez, 1987, Cohen, 1990]) is a generalization of logic programming. Unification, the basic operation in logic programs, is replaced by a more general mechanism of constraint satisfaction over a specific computation domain. An instance of the general CLP scheme is obtained by selecting a computation domain, a set of allowed constraints and designing a solver for the constraints. CLP combines the advantages of logic programming (declarative semantics, nondeterminism, partial answers) with the efficiency of specialized constraint satisfaction algorithms. CLP($\Re$) is an instance of the CLP scheme which extends logic programs with interpreted arithmetic functions and a solver for systems of linear equations and inequalities over the domain of $\Re$eals. In our experiments we use an implementation of CLP($\Re$) [Holzbaur, 1992] which is an extension of SICStus Prolog [Carlsson and Widen, 1991].

A CLP($\Re$) program is a set of clauses of the form:

$H \leftarrow C_1, \ldots, C_n.$

and a CLP($\Re$) query is a clause without head:

$\leftarrow C_1, \ldots, C_n.$

where H is an atom and C_i are negated or non-negated atoms or arithmetic constraints. Arithmetic constraints are equations or inequalities, built up from real constants, variables, $+, -, *, /$ and $=, \geq, \leq, >, <$ where all of these symbols have the usual meaning and parentheses may be used. An atom is a predicate symbol applied to a number of terms. A term is a constant, a variable, an uninterpreted functor applied to a number of terms, or an arithmetic term. Variables start with capitals and are implicitly universally quantified in front of a clause, and constants start with lower-case letters.

We illustrate the CLP($\Re$) language by specifying addition and multiplication of complex numbers. Computation with complex numbers is needed to simulate and diagnose analog circuits under AC conditions. A complex number $Z = Re+j*Im$ is represented by a pair *c(Re,Im)*.

add(c(Re1,Im1), c(Re2,Im2), c(Re1+Re2, Im1+Im2)).

mult(c(Re1,Im1), c(Re2,Im2), c(Re,Im)) $\leftarrow$
*Re = Re1*Re2 − Im1*Im2,*
*Im = Re1*Im2 + Im1*Re2.*

The above CLP($\Re$) program allows for queries involving not only addition and multiplication, but subtraction and division of two complex numbers as well. For example:

$\leftarrow$ *mult(c(1,2), c(3,4), Z).*
Z = c(−5,10)

$\leftarrow$ *mult(X, c(3,4), c(−5,10)).*
X = c(1,2)

Answering the second query actually requires to solve the following system of equations:

*3*Re1 − 4*Im1 = −5,*
*4*Re1 + 3*Im1 = 10.*

which yields the solution *Re1=1, Im1=2.*

2.2 Modeling analog circuits

The purpose of this subsection is to show how analog circuits can be modeled by CLP($\Re$). First we show how the CLP($\Re$) capability to solve systems of inequations can be used to model parameter tolerances. Then we illustrate how the same CLP($\Re$) model is used not only for prediction, but also for diagnosis. When the measured voltage is out of the expected range, CLP($\Re$) computes values of suspected components which differ from the nominal values. This forms the basis for the isolation of soft faults in more complicated analog circuits.

A *model* of a system is a triple $\langle$*SD, COMPS, OBS*$\rangle$ where

1. *SD*, the system description, is a CLP($\Re$) program with a distinguished top-level binary predicate *model(COMPS, OBS)* which relates states of the system components to observations.

2. *COMPS*, states of the system components, is an n-tuple $\langle S_1, \ldots, S_n \rangle$ where n is the number of components, and variables S_i denote states (e.g., normal or abnormal) of components.

3. *OBS*, observations, is an m-tuple $\langle P_1, \ldots, P_i, In_{i+1}, \ldots, In_j, Out_{j+1}, \ldots, Out_m \rangle$ where P are the model parameters, and *In* and *Out* denote inputs and outputs of the model, respectively.

We consider a simple model of two resistors in a series operating under DC conditions. Voltages of 12.5 and 10 Volts are applied at the ends (Figure 1, an example from [McKeon and Wakeling, 1990]). Both resistances are within the range *i(1000, 2000)* Ω. The first question is: What is the voltage range at the node between the two resistors?

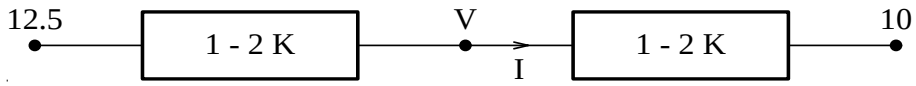

Figure 1: Two resistors with tolerances in a series.

SD consists of the following CLP($\Re$) program:

model(comps(R1,R2), obs(V,I)) $\leftarrow$
V1 = 12.5,
V2 = 10,
resistor(R1, i(1000,2000), V1, V, I),
resistor(R2, i(1000,2000), V, V2, I).

resistor(ok, Rnom, V1, V2, I) $\leftarrow$ *Rnom*I = V1−V2.*
resistor(ab(R), _, V1, V2, I) $\leftarrow$ *R*I = V1−V2.*

COMPS is a pair *comps(R1,R2)*, where R_i denote states of resistors: *ok* when the resistance is within the nominal range, and *ab(R)* otherwise. *R* in *ab(R)* denotes an unknown resistance of a faulty resistor which can be out of the allowed tolerances. *OBS* is a pair *obs(V,I)*, where V is the voltage at the node between the two resistors and I is the current through the resistors.

The model relates states of the resistors to the voltage and the current. Atoms in the body of the first clause represent models of resistors which enforce local constraints between voltages and currents (Ohm's law). Shared variables represent connections between the components and enforce global constraints, e.g., Kirchhoff's law for voltages. The last two clauses define behavior of a resistor for the case when it is normal, and when it is out of tolerances, respectively.

The following query asks for the voltage and current, under the assumption that both resistors are *ok*, i.e., within the allowed tolerances:

$\leftarrow$ *model(comps(ok,ok), obs(V,I)), inf(V,Vmin), sup(V,Vmax).*

$V \leq 10.0 + 2000.0*I,$
$V \leq 12.5 - 1000.0*I,$
$V \geq 12.5 - 2000.0*I,$
$V \geq 10.0 + 1000.0*I,$
$Vmin = 10.833333333333334,$
$Vmax = 11.666666666666666$

CLP($\Re$) returs a set of inequalities as an answer, and also computes infimum and supremum of the voltage between the two resistors. In contrast to our, symbolic approach, [McKeon and Wakeling, 1990] use an iterative, numeric approach to compute the voltage range.

Where do the inequalities come from? In our CLP($\Re$) implementation one can use generalized constants. A generalized constant is an ordered pair of floating-point numbers, defining the lower and upper bound of an interval which contains the original constant. A multiplication of a generalized constant by a variable is equivalent to the following specification:

$i(A,B)*X = Y \leftarrow X \geq 0,\ A*X \leq Y,\ Y \leq B*X.$

$i(A,B)*X = Y \leftarrow X < 0,\ B*X \leq Y,\ Y \leq A*X.$

In the actual implementation the case analysis on X is avoided by a two-pass solution, and at a cost of incompleteness [Mozetič and Holzbaur, 1993].

Now we turn to the second problem: diagnosis of soft faults. Assume that the actual, measured voltage is 12 Volts, i.e., it is outside of the predicted range. This indicates that at least one of the resistors is faulty, i.e., its resistance must be out of the allowed tolerances. We make a single fault assumption and use the same CLP($\Re$) model to compute the resistance of a suspected resistor. The following two queries return lower and upper bounds for resistances of suspected faulty resistors. Note that each query has two possible answers, corresponding to two possible faults:

$\leftarrow$ *model(comps(R1,R2), obs(12,I)), maximize(I).*

I = 0.0005,
R1 = ok,
R2 = ab(4000.0) ;

I = 0.002,
R1 = ab(250.0),
R2 = ok

$\leftarrow$ *model(comps(R1,R2), obs(12,I)), minimize(I).*

I = 0.00025,
R1 = ok,
R2 = ab(8000.0) ;

I = 0.001,
R1 = ab(500.0),
R2 = ok

Suspected component	Computed value [Ω]
R1	250–500
R2	4000–8000

Table 1: Computed resistance ranges of potentially faulty resistors.

Table 1 summarizes the results of diagnosis. The example illustrates the basic idea of using CLP($\Re$) to diagnose soft faults: each individual component is considered in turn and its actual value is computed from the measurements. If tolerances are taken into account then the computed value is a range, otherwise it is a single value. In our experiments with filters we do not take parameter tolerances into account. However, instead of one we take several measurements, and combine the computed values of suspected faulty components in order to rank them in decreasing likelihood of beeing faulty.

CLP($\Re$) is resticted to systems of *linear* equations and inequalities. Non-linear constraints are accepted but not resolved — they are just delayed until (if) they eventually become linear. In order to make CLP($\Re$) applicable to the diagnosis of analog circuits we have to make some assumptions. We have to restrict models to linear or piecewise linear circuits, and assume that there are no multiple faults (a single fault assumption). In the next section we give a formal argument which shows that under those assumptions, linear CLP($\Re$) is indeed sufficient for the diagnosis of single soft faults.

3 Application of the node-analysis method

In our approach, the CLP($\Re$) model is based on the node-analysis method. Thus the unknown quantities are node potentials. Let an electric circuit have n nodes and b branches. One of the nodes of the circuit can be grounded without affecting any potential differences across the nodes bounding the branch. The node-analysis method represents the circuit by three independent matrix equations

$$\begin{aligned} A\,j &= 0 \\ v &= A^T\,e \\ j &= Y\,v \end{aligned}$$

where the variables denote

- j – current vector
- e – vector of the node potentials
- v – voltage vector
- Y – admittance matrix

The first matrix equation represents the Kirchhoff's current law. The second equation determines voltages across the branches of the circuit and is implicitly contained in the CLP($\Re$) model. The last equation represents the Ohm's law for all the branches.

3.1 Simulation of a circuit with voltage input

Assume that an external voltage source is added to the circuit. The Kirchhoff's current low equation must be modified to include the source current, which is an additional unknown quantity. The system gets an additional equation related to the input voltage.

$$\begin{aligned} A\,j + B\,i_{in} &= 0 \\ v &= A^T\,e \\ j &= Y\,v \\ B^T\,e &= u_{in} \end{aligned}$$

The system of equations can be rearranged to

$$\begin{aligned} A\,Y\,A^T\,e \;+\; B_{in}\,i_{in} &= 0 \\ B_{in}^T\,e &= u_{in} \end{aligned}$$

where i_{in} represents the source current, and u_{in} the source voltage, respectively. The matrix B_{in} specifies the nodes to which the source is connected. The resulting system is a system of n linear equations with n unknown quantities. The output voltage can be easily calculated from the resulting node potential by

$$u_{out} = B_{out}^T\,e$$

3.2 Determining the value of a component in terms of the known system input and output voltages

We have shown how to determine the system output voltage in terms of the known voltage input. In diagnosis of soft faults, however, is our main interest in determining the value of particular component in terms of the known voltage input (stimulus) and the output voltage (measurement). We have to show that this problem can be described by a system of linear equations which yield a unique solution. Let A_i be the ith column of A.

$$A = (A_1, A_2, \cdots, A_b)$$

Accordingly, the admittance matrix is

$$Y = \begin{pmatrix} Y_1 & 0 & \dots & 0 \\ 0 & Y_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & Y_b \end{pmatrix}$$

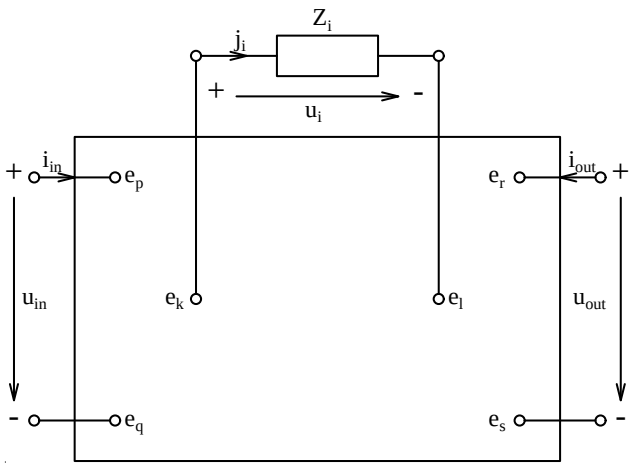

Figure 2: Electric circuit with suspected component Z_i separated.

Let the unknown component be Z_i (Figure 2). The component Z_i is first taken out of the circuit and considered separately from the rest of it. Now the circuit has n nodes (no node deleted) and $b-1$ branches (branch i was drawn out). The new matrix A^* is

$$A^* = (A_1, A_2, \cdots, A_{i-1}, A_{i+1}, \cdots, A_b)$$

and the new admittance matrix is

$$Y = \begin{pmatrix} Y_1 & 0 & \dots & 0 & 0 & \dots & 0 \\ 0 & Y_2 & \dots & 0 & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & & \vdots \\ 0 & 0 & \dots & Y_{i-1} & 0 & \dots & 0 \\ 0 & 0 & \dots & 0 & Y_{i+1} & \dots & 0 \\ \vdots & \vdots & & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 0 & 0 & \dots & Y_b \end{pmatrix}$$

The new current vector j^* does not contain the current j_i.

Two additional currents must be added to the Kirchhoff's current low: the source current and the current through the unknown component. The output current is 0.

$$\begin{aligned} A^* j^* + A_i j_i + B_{in} i_{in} &= 0 \\ j^* &= Y^* v^* \\ v^* &= A^{*T} e \end{aligned}$$

The system has two additional equations corresponding to the source voltage and to the output voltage. The resulting equations are:

$$\begin{aligned} A^* Y^* A^{*T} e + A_i j_i + B_{in} i_{in} &= 0 \\ B_{in}^T e &= u_{in} \\ B_{out}^T e &= u_{out} \end{aligned}$$

This is a system of linear equations and can be solved by CLP($\Re$). The value of the unknown component can be easily calculated by

$$Z_i = \frac{A_i^T \, e}{j_i}$$

4 DFT methodology for active analog filters

In our earlier work [Mozetič *et al.*, 1991] we made an introductory study of automated fault diagnosis of analog circuits by means of CLP($\Re$). Promising results on a relatively simple analog circuit have been obtained. Next we have tried to apply the approach to more complex circuits, in particular in the domain of active analog filters in which we have gathered most practical experiences [Novak *et al.*, 1993]. We have found that the DFT methodology for active analog filters proposed by Soma [1990] can be enhanced and operationalized by using CLP($\Re$). In other words, CLP($\Re$) provides means for automatic fault diagnosis of active analog filters designed in accordance with the proposed DFT methodology.

Soma [1990] presented a DFT methodology applicable to a class of active analog filters based on the standard operational amplifier design. Possible faults are assumed to be limited to the passive components, i.e., the operational amplifiers are fault-free. Testability is defined as controllability and observability of significant waveforms within the filter structure. The significant waveforms are the input/output signals of every stage in the filter, and the methodology permits full control and observation of these signals. The main idea to increase the controllability and observability is to introduce MOS switches to individual stages of a filter in order to reduce the capacitive effects in the impendances of the stages not under test. The normal filter design becomes an "analog scan" structure in the test mode. The modified filter circuit can then be tested in the following modes of operation:

- normal mode (switches initialized such that all DFT transformations are disabled),
- individual stage test (switches set to disconnect the capacitors from the stages not under test),
- all-test mode (switches set to disconnect all the capacitors in the circuit).

As an example, consider the filter circuit shown in Figure 3. The CLP($\Re$) model of the filter is in Appendix. Switches T_1, T_2, and $\overline{T_2}$ have been introduced for the DFT purposes. Their states in individual operation modes are as follows:

- normal mode: T_1 ON, T_2 ON, $\overline{T_2}$ OFF,

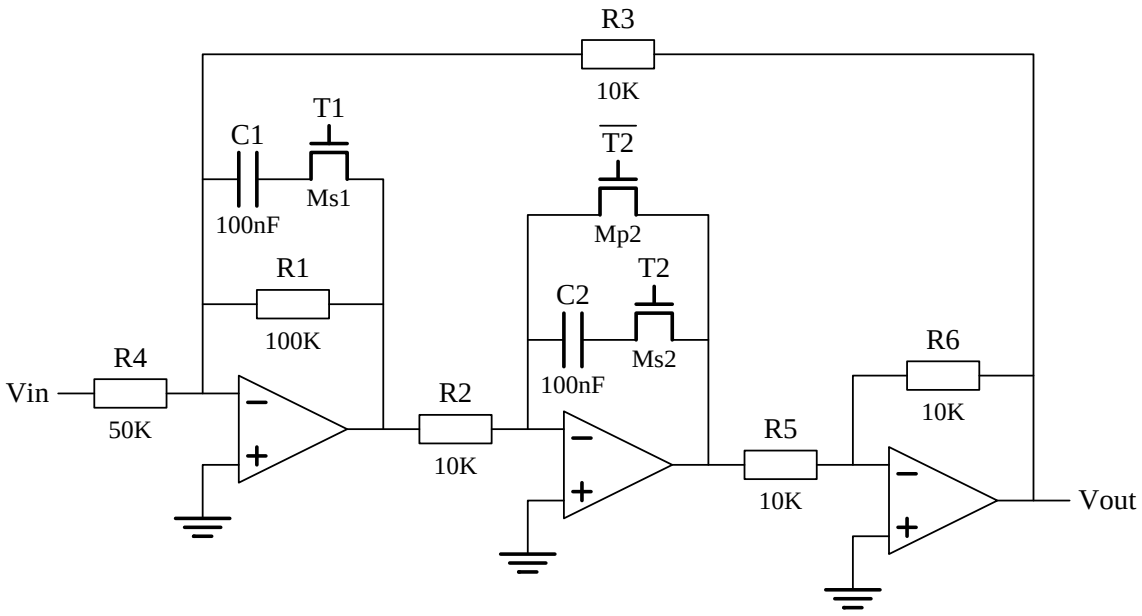

Figure 3: A low pass biquad filter, designed-for-testability, with inserted switches T_1 and T_2 .

- first stage test: T_2 OFF (C_2 disconnected), T_1 ON, $\overline{T_2}$ ON,
- second stage test: T_1 OFF (C_1 disconnected), T_2 ON, $\overline{T_2}$ OFF,
- all-test mode: T_1 OFF, T_2 OFF, $\overline{T_2}$ ON.

The fault isolation procedure described in [Soma, 1990] includes a variety of techniques that may prove useful in diagnosis. However, given examples and case studies do not offer a unique way toward fault isolation. They are rather used to illustrate the rich set of possibilities available to the designer when troubleshooting a filter circuit that has been designed for testability.

In our approach, we use the same modes of operation of the stage-under-test as proposed by Soma [1990], yet we restrict the measurements to the case where measurement results can be fed directly to the CLP($\Re$) system in order to perform the fault isolation process automatically.

4.1 Fault isolation

Figure 4 gives gain and phase characteristics of the fault-free filter from Figure 3 in all testing modes. Simulation under the AC conditions for the fault-free circuit in the normal mode was done by CLP($\Re$) interpreter. Comparison between the actual circuit and the CLP($\Re$) model has shown that simulated results are close to the measured ones within

less than 5%. When the correctness of the model is confirmed, the process of deriving diagnoses proceeds in the following way.

First are measurements of gain and phase at selected test frequencies performed on the actual filter circuit. The results are compared to the simulated fault-free circuit output under the same input stimuli. If a discrepancy is detected, a CLP($\Re$) model is used to compute the deviating values of the suspected components that might have caused the measured faulty circuit output. The values are computed for the normal mode, all-test mode, and each individual stage test mode, respectively. The mean value $\bar{x}$ and coefficient of variation $s/\bar{x}$ (i.e., normalized standard deviation) are computed for each suspected component in each operation mode, and for the composite case. Large $s/\bar{x}$ indicates that the predicted component value varies considerably across several measurements, and consequently the component can be ruled out as a candidate. If we assume that faults are non-intermittent, then a faulty component assumes some value different than nominal, but this value should not change during testing.

In the next step, the list of suspected components is reduced:

- Components with computed negative values are ruled out.
- Components with large value of $s/\bar{x}$ are ruled out.
- Analysis of the all-test mode is performed in order to find out if the faulty component is a resistor or a capacitor.
- In the remaining list, the components with minimum value of $s/\bar{x}$ are declared as potentially faulty.

In the next section we describe some typical fault situations and show how the conclusions of the circuit diagnosis are drawn from the results of computation of the CLP($\Re$) system.

5 Experimental results

The low pass biquad filter, shown in Figure 3, has been implemented for experimental purposes in thick film hybrid technology. LS404C operational amplifiers and HEF4066B MOS switches were employed. HP4192 LF Impedance Analyzer was used for measuring gain and phase values in the range of 5Hz – 10kHz. CLP($\Re$) simulations were performed on a SUN SPARCstation IPC.

In order to evaluate our diagnostic approach we replaced various filter components (one at a time) with components with different nominal values. While it is desirable to keep the number of measurements at minimum, we performed experiments and CLP($\Re$) diagnoses only at two selected frequencies. In more complex circuits, more measurements might be needed to compute reliable diagnoses.

5.1 Example 1: C_1 1nF instead of 100nF

In the first case we have inserted a capacitor C_1 with 1 nF instead of the correct 100 nF into the circuit. Table 2 gives the measured values of gain and phase at selected frequencies of 100 Hz and 200 Hz for the four possible modes of circuit operation with $C_1 = 1$ nF. Table 3 presents the computed mean values $\bar{x}$ of suspected faulty components together with the corresponding coefficients of variation $s/\bar{x}$ for the given operation modes.

Modes	Frequency [Hz]	Gain [dB]	Phase [^{0}]	Gain difference [dB]	Phase difference [^{0}]
normal mode	100	-13.957	176.34	-4.29	2.53
	200	-13.904	172.65	-4.40	158.20
all-test mode	100	-40.40	-178.92	-1.48	0.11
	200	-40.43	179.9	-1.50	-0.02
1st stage test	100	-40.45	176.57	14.05	76.80
	200	-40.52	176.57	19.91	77.71
2nd stage test	100	-13.993	176.36	0.00	-0.04
	200	-14.049	172.79	0.00	0.05

Table 2: Measured gain and phase for the fault $C_1 = 1nF$.

Suspected component	normal mode		all-test mode		1st stage test		2nd stage test	
	[Ω,nF]	$s/\bar{x}$	[Ω,nF]	$s/\bar{x}$	[Ω,nF]	$s/\bar{x}$	[Ω,nF]	$s/\bar{x}$
R_1	1546	0.876	83435	0.002	1832	0.862	99083	0.003
R_2	277	0.497	11985	0.002	313	0.483	10092	0.003
R_3	5548	0.419	2323	0.011	1	1.388	10006	0.000
R_4	2369	41.420	59352	0.002	1668	0.504	49970	0.000
R_5	277	0.495	11985	0.002	313	0.483	10092	0.003
R_6	21202	0.431	8343	0.002	16974	0.435	9908	0.003
C_1	1.16	0.063	x	x	1.21	0.021	x	x
C_2	2.71	0.507	x	x	x	x	1.01	0.003

Table 3: Computed values of suspected components for different test modes for $C_1 = 1nF$.

Inspection of coefficients of variation $s/\bar{x}$ for the normal mode immediately points to C_1 as being faulty because it has minimum $s/\bar{x}$ and its value differs by an order of magnitude from the others. Although the result is achieved in the first step we proceed by the other three in order to illustrate the process of deriving the diagnosis as well as to confirm the initial result.

Resistor R_4 can be ruled out as a fault candidate due to the large value of $s/\bar{x}$ in the normal mode. Simulation results of the fault-free circuit in the all-test mode are close to the measured values (Table 2), hence no further element can be ruled out at this step. The situation implicitly indicates that fault can be expected in C_1 or C_2 since the circuit operates correctly in the all-test mode where the capacitors are inactive.

Computed values for the first stage test mode indicate that R_3 is no longer a candidate of being faulty due to its large value of $s/\bar{x}$.

The results of computation for the second stage test mode show equal values of $s/\bar{x}$ for the remaining suspected elements, except for C_1 which is excluded at this specific measurement situation. Again, the measured values are close to the simulated fault-free circuit (Table 2), which implicitly indicates that C_1 is the probable faulty element.

Suspected faulty component	Computed mean value	Coefficient of variation
C_1	1.19	0.045
R_6	14107	0.506
R_3	4470	0.920
R_2	5667	1.021
R_5	5667	1.021
R_1	46474	1.039
C_2	5.18	1.094
R_4	28340	1.801

Table 4: Ranking of the suspected components and their predicted values for $C_1 = 1nF$.

Table 4 gives the composite average values for the four circuit operation modes. Here again is the coefficient of variation $s/\bar{x}$ of C_1 by an order of magnitude lower than the other values which confirms previously stated diagnosis. Notice also, that the computed value of C_1 closely resembles the actual value of the inserted fault in the experimental circuit.

5.2 Example 2: R_3 $1k\Omega$ instead of $10k\Omega$

For the second experiment the circuit is modified so that R_3 is 1 $k\Omega$ instead of correct 10 $k\Omega$. The measured values for this case are given in Table 5, and computed values of the potentially faulty components in Table 6. In Table 7 we give the final predicted mean values and ranking of the suspected components according to $s/\bar{x}$.

Due to the negative values computed for the normal mode, R_2, R_5, R_6, C_1 and C_2 are ruled out. From the remaining candidates, R_3 seems to be the most probable fault because it has the smallest value of $s/\bar{x}$. At the same time, computed value of $R_3 = 1001\Omega$ matches the actual value in the modified circuit.

Modes	Frequency [Hz]	Gain [dB]	Phase [^{0}]	Gain difference [dB]	Phase difference [^{0}]
normal mode	100	-33.627	179.60	-23.94	5.80
	200	-32.47	179.0	-22.97	164.55
all-test mode	100	-43.52	179.86	-4.59	-0.10
	200	-43.54	179.63	-4.61	-0.293
1st stage mode	100	-56.37	104.34	-1.86	4.56
	200	-61.63	79.09	-1.19	1.86
2nd stage mode	100	-33.983	179.23	-20.00	3.22
	200	-33.98	172.79	-19.92	6.39

Table 5: Measured gain and phase for the fault $R_3 = 1k\Omega$.

Suspected components	normal mode		all-test mode		1st stage test		2nd stage test	
	[Ω,nF]	$s/\bar{x}$	[Ω,nF]	$s/\bar{x}$	[Ω,nF]	$s/\bar{x}$	[Ω,nF]	$s/\bar{x}$
R_1	18	0.837	57422	0.001	31224	0.160	19	0.860
R_2	-129030	0.905	17413	0.001	11904	0.052	1496	0.018
R_3	1001	0.0004	749	0.003	161	0.392	999	0.000
R_4	52706	19.615	84954	0.001	59541	0.052	495708	0.008
R_5	-129201	0.905	17413	0.001	11904	0.052	15032	0.017
R_6	-1292	0.912	5742	0.001	8379	0.056	1.8	0.860
C_1	-1316	0.910	x	x	118	0.045	x	x
C_2	-1315	0.911	x	x	x	x	450	0.027

Table 6: Computed values of suspected components for different test modes for $R_3 = 1k\Omega$.

Measurement results of the all-test mode indicate that the faulty element is one of the resistors. Computed values $s/\bar{x}$ for the all-test mode are all close to zero and no further conclusion can be drawn. Likewise, computed values for the first and the second stage test mode do not contribute to the final diagnosis. The average values of $s/\bar{x}$ given in Table 7 indicate that R_3 is the most probable cause of malfunctioning. The mean value of R_3 is equal to 728Ω. Although less precise, the calculated value closely resembles the actual situation in the circuit.

So far, experimental results and CLP($\Re$) computations were made only for two selected frequencies. While it is desirable to keep the number of measurements at minimum, more complex circuits would probably require more measurements in order to compute reliable diagnoses. Just to get the impression how additional measurements improves the fault isolation, an additional measurement was made at 160 Hz. Computed results for the three frequencies are given in Table 8. In this particular case the results are only slightly

Suspected faulty component	Computed mean value	Coefficient of variation
R_3	728	0.504
R_1	22171	1.160
R_6	3207	1.338
C_1	-599	1.801
R_4	173227	2.532
C_2	-432	2.850
R_2	-21186	3.771
R_5	-21167	3.775

Table 7: Ranking of the suspected components and their predicted values for $R_3 = 1k\Omega$.

Suspected faulty component	Computed mean value	Coefficient of variation
R_3	729	0.492
R_1	21946	1.142
R_6	3214	1.302
C_1	-510	1.750
R_4	142714	2.750
C_2	-349	2.963
R_2	-16981	4.030
R_5	-16963	4.034

Table 8: Ranking of the suspected components and their predicted values after testing at three selected frequencies for the fault $R_3 = 1k\Omega$.

improved.

5.3 Example 3: R_5 $100k\Omega$ instead of $10k\Omega$

In the last experiment, we inserted a deviation fault of R_5 (100 $k\Omega$ instead of correct 10 $k\Omega$) into the circuit. Like in previous examples, measurement results (Table 9), computed values of the suspected components for the four modes of operation (Table 10), and the computed mean values and ranking of the suspected components are given (Table 11).

From the computed values of components for the normal mode, R_3 can be eliminated due to its negative value. R_1 and R_4 are the next candidates to be ruled out because of higher values of $s/\bar{x}$. The difference between the measurement results and the simulated

Modes	Frequency [Hz]	Gain [dB]	Phase [^{0}]	Gain difference [dB]	Phase difference [^{0}]
normal mode	100	-17.987	52.27	-8.31	-121.53
	200	-37.57	5.87	-28.06	-8.57
all-test mode	100	-58.49	-179.33	-19.56	-359.29
	200	-58.49	179.82	-19.56	-0.10
1st stage test	100	-74.47	103.27	-19.89	3.49
	200	-77.98	80.24	-17.54	-14.98
2nd stage test	100	-15.448	147.63	-1.450	-28.77
	200	-18.161	128.47	-4.106	-44.37

Table 9: Measured gain and phase for the fault $R_5 = 100k\Omega$.

Suspected components	normal mode		all-test mode		1st stage test		2nd stage test	
	[Ω,nF]	$s/\bar{x}$	[Ω,nF]	$s/\bar{x}$	[Ω,nF]	$s/\bar{x}$	[Ω,nF]	$s/\bar{x}$
R_1	1008	1.296	9975	0.000	104	4.637	9914	0.000
R_2	77702	0.436	100229	0.000	85783	0.212	100854	0.000
R_3	-2017	0.900	66	0.000	0.6	4.786	5942	0.380
R_4	591867	1.577	475525	0.000	428429	0.212	54560	0.072
R_5	77699	0.436	100230	0.000	85783	0.212	100849	0.000
R_6	1280	0.327	998	0.000	1145	0.169	991	0.000
C_1	767	0.471	x	x	859	0.215	x	x
C_2	779	0.423	x	x	x	x	1005	0.002

Table 10: Computed values of components for different test modes for $R_5 = 100k\Omega$.

values of gain and phase in the all-test mode indicate that the faulty element is one of the resistors.

R_2, R_5, and R_6 remain candidates also after inspecting the values in Table 11. The reason is in the transfer function of the filter circuit, where the three resistors always appear in the same subexpression.

6 Conclusion

Achieved results show that the model-based diagnosis with CLP($\Re$) can be used in automatic fault isolation of active analog filters designed in accordance with the proposed DFT methodology [Soma, 1990]. Although the results refer to a narrow problem domain, they indicate that the DFT methodology can be enhanced and operationalized by using

Suspected faulty component	Computed mean value	Coefficient of variation
R_6	1103	0.195
R_5	91140	0.196
R_2	91142	0.196
C_2	892	0.258
C_1	813	0.296
R_1	5250	0.963
R_4	387595	1.0698
R_3	997	3.371

Table 11: Ranking of the suspected components and their predicted values for $R_5 = 100k\Omega$.

CLP($\Re$) and model-based diagnosis techniques. From the presented examples, derivation of a formal fault diagnosis algorithm is straightforward. The approach has a good chance to scale up, since DFT enables focusing on relatively small stages-under-test.

In the above experiments, nominal values of the model components were taken as constants instead of more realistic tolerance intervals. Parameter tolerances could be modeled with CLP($\Re$) by a technique described in section 2. However, this turned out not to be critical since even without the tolerances, the model matched the actual circuit within less then 5%. More serious are rounding errors of the floating-point arithmetic. A potential solution of using CLP($\mathcal{Q}$) instead of CLP($\Re$) seems unpractical, since in our experiments CLP($\mathcal{Q}$) turned out to be 100 times slower then CLP($\Re$) and needed considerably more space. What is needed is to represent $\Re$eals with floating-point intervals instead of single floating-point numbers, as proposed by [Lee and van Emden, 1992].

One of the restrictions of the approach is the single fault assumption. This is due to the limitation of CLP($\Re$) to *linear* constraints. Coping with multiple soft faults would require the ability to resolve nonlinear constraints. This can be done by ICLP($\Re$) [Lee and van Emden, 1992], for example, where the constraint satisfaction is based on local constraint propagation, and resembles classical iterative numerical approximation techniqes. However, from the practical point of view, diagnosis of multiple soft faults in analog circuits is no pressing issue. A more practical problem is that test frequencies have to be determined by the designer. Implementation of an algorithm for the selection of test frequencies is one of our current research goals.

Acknowledgements

Igor Mozetič is supported by the Austrian "Fonds zur Förderung der wissenschaftlichen Forschung" under grant P9426-PHY. Austrian Research Institute for Artificial Intelligence is supported by the Austrian Federal Ministry of Science and Research. Franc Novak, Marina Santo-Zarnik and Anton Biasizzo are supported by the Slovenian Ministry of Research and Technology under grants C2-1521-106 and Z2-0378-106.

References

[Bandler and Salama, 1985] Bandler, J.W., Salama, A.E. Fault diagnosis of analog circuits. *Proc. IEEE*, Vol. 73, No. 8, pp. 1279-1327, 1985.

[Carlsson and Widen, 1991] Carlsson, M., Widen, J. SICStus Prolog user's manual. Swedish Institute of Computer Science, Kista, Sweden, 1991.

[Cavin and Hilbert, 1990] Cavin, R.J., Hilbert, J.L. Design of integrated circuits: directions, and challenges. *Proc. IEEE*, Vol. 78, No. 2, pp. 418-435, 1990.

[Cohen, 1990] Cohen, J. Constraint logic programming languages. *Communications of the ACM 33 (7)*, pp. 52-68, 1990.

[Dague *et al.*, 1990] Dague, P., Deves, P., Luciani, P., Taillibert, P. Analog systems diagnosis. *Proc. 9th European Conf. on Artificial Intelligence, ECAI-90*, pp. 173-178, Stockholm, Sweden, 1990.

[de Kleer and Brown, 1992] de Kleer, J., Brown, J.S. Model-based diagnosis in SOPHIE III. In Hamscher, W.C., Console, L., de Kleer, J. (Eds.) *Readings in Model-based Diagnosis*, pp. 179-205, Morgan Kaufmann, San Mateo, CA, 1992.

[Eagner and Williams, 1989] Eagner, K.D., Williams, T.W. Design for testability of analog/digital networks. *IEEE Trans. Indust. Elect.*, Vol. 36, No. 2, pp. 227-230, 1989.

[Fasang, 1989] Fasang, P.P. Analog/digital ASIC design for testability. *IEEE Trans. Indust. Elect.*, Vol. 36, No. 2, pp. 219-226, 1989.

[Holzbaur, 1992] Holzbaur, C. DMCAI CLP reference manual. Report OEFAI-TR-92-24, Austrian Research Institute for Artificial Intelligence, Vienna, Austria, 1992.

[IEEE, 1979] *IEEE Trans. Circuits Syst.*, Special issue on automatic fault diagnosis, Vol. CAS-26, 1979.

[Jaffar and Lassez, 1987] Jaffar, J., Lassez, J.-L. Constraint logic programming. *Proc. 14th ACM Symp. on Principles of Programming Languages*, pp. 111-119, Munich, 1987.

[Jarwala and Tsai, 1991] Jarwala, M., Tsai, S.J. A framework for testability of mixed analog/digital circuits. *Proc. IEEE 1991 Custom Integrated Circuit Conf.*, pp. 13.5.1-13.5.4, 1991.

[Lee and van Emden, 1992] Lee, J.H.M, van Emden, M.H. Adapting CLP($\Re$) to floating-point arithmetic. *Proc. Intl. Conf. on Fifth Generation Computer Systems*, pp. 996-1003, ICOT, Tokyo, 1992.

[McKeon and Wakeling, 1989] McKeon, A., Wakeling, A. Fault diagnosis in analogue circuits using AI techniques. *Proc. Intl. Test Conf., ITC-89*, pp. 118-123, 1989.

[McKeon and Wakeling, 1990] McKeon, A., Wakeling, A. Model-based analogue circuit fault diagnosis. *Proc. TEST-90*, pp. 1-14, 1990.

[Mozetič *et al.*, 1991] Mozetič, I., Holzbaur, C., Novak, F., Santo-Zarnik, M. Model-based analogue circuit diagnosis with CLP($\Re$). *Proc. 4th Intl. GI Congress*, pp. 343-353, Munich, Springer-Verlag, 1991.

[Mozetič and Holzbaur, 1993] Mozetič, I., Holzbaur, C. Interval arithmetic with CLP($\Re$). Report OEFAI-TR-93-19, Austrian Research Institute for Artificial Intelligence, Vienna, Austria, 1993.

[Novak *et al.*, 1993] Novak, F., Biasizzio, A., Santo-Zarnik, M., Mozetič, I. On automatic fault isolation using DFT methodology for active analog filters. *Proc. 3rd European Test Conf., ETC-93*, pp. 534-535, IEEE Press, Rotterdam, April 19-22, 1993.

[Ohletz, 1991] Ohletz, M.J. Hybrid built-in self-test for mixed analogue/digital integrated circuits. *Proc. European Test Conf., ETC-91*, Munich, pp. 307-316, 1991.

[Sierzega and Rastogi, 1990] Sierzega, K., Rastogi, R. Model-based reasoning finds analog PCB faults. *Electronic Test*, pp. 26-30, March 1990.

[Slamani and Kaminska, 1992] Slamani, M., Kaminska, B. Analog circuit fault diagnosis. *Design and Test Comput.*, Vol. 9, No. 1, pp. 30-39, 1992.

[Soma, 1990] Soma, M. A design-for-test methodology for active analog filters. *Proc. IEEE Intl. Test Conf. 1990*, pp. 183-192, Washington D.C., IEEE, 1990.

[Vefling and Viktil,] Vefling, H., Viktil, M. Test of mixed analog/digital components by boundary scan methods. Esprit project 2478, report R8.E5.

[Walker and Alexander, 1992] Walker, A., Alexander, W.E. Fault diagnosis in analog circuits using element modulation. *Design and Test Comput.*, Vol. 9, No. 1, pp. 19-29, 1992.

Appendix: CLP($\Re$) model of the filter

```
%  File: soma.pl
%
%  A low pass biquad filter designed-for-testability according
%  to Soma [1990].
%  A complex number Z = Re+j*Im is represented by a pair c(Re,Im).
%  Component tolerances not taken into account.

test( Test, F, Gain, Phase ) :-
   V1 = c(1,0),
   test_mos( Test, MOS ),
   filt( comps(ok,ok,ok,ok,ok,ok,ok,ok,ok,ok,ok), MOS, F, V1, V2 ),
   gain( V1, V2, Gain ),
   phase( V2, Phase ).

   test_mos( origin,  mos(shrt,shrt,dis) ).
   test_mos( normal,  mos(res,res,dis) ).
   test_mos( alltest, mos(dis,dis,res) ).
   test_mos( stage1,  mos(res,dis,res) ).
   test_mos( stage2,  mos(dis,res,dis) ).

filt( Comps, MOS, F, Vin, Vout ) :-
   frequency( F, W ),
   R1 = 100000,
   R2 = 10000,
   R3 = 10000,
   R4 = 50000,
   R5 = 10000,
   R6 = 10000,
   C1 = 100.0e-9,
   C2 = 100.0e-9,
   T1 = 60,
   T2 = 60,
   NT2 = 60,
   filter( Comps, MOS,
           param(W,R1,R2,R3,R4,R5,R6,C1,C2,T1,T2,NT2),
           volt(Vin,Vout,_,_,_,_,_,_,_,_) ).

filter( comps(SR1,SR2,SR3,SR4,SR5,SR6,SC1,SC2,SA1,SA2,SA3),
        mos(ST1,ST2,SNT2),
```

```
        param(W,R1,R2,R3,R4,R5,R6,C1,C2,T1,T2,NT2),
        volt(Vin,Vout,V1,V2,V3,V4,V5,Vt1,Vt2) ) :-
   Vgnd = c(0,0),
   cmp_add( I1, Ib1, Ir4 ),
   cmp_add( I2, Ir1, I1 ),
   cmp_add( Ir3, Ic1, I2 ),
   cmp_add( Ic1, Ir1, I3 ),
   amplifier( SA1, W, Vgnd, V1, V2, Ia1, Ib1, Io1 ),
   amplifier( SA2, W, Vgnd, V3, V4, Ia2, Ib2, Io2 ),
   amplifier( SA3, W, Vgnd, V5, Vout, Ia3, Ib3, Io3 ),
   cmp_add( I3, Io1, Ir2 ),
   cmp_add( I4, Ib2, Ir2 ),
   cmp_add( I5, Ic2, I4 ),
   cmp_add( I4, Io2, Ir5 ),
   cmp_add( Ir6, Ib3, Ir5 ),
   cmp_add( Ir3, Ir6, I6 ),
   cmp_add( Io3, I6, c(0,0) ),
   bridge( ST1, T1, Ic1, Vt1, V2 ),
   bridge( ST2, T2, Ic2, Vt2, V4 ),
   bridge( SNT2, NT2, I5, V3, V4 ),
   capacitor( SC2, W, C2, Ic2, V3, Vt2 ),
   capacitor( SC1, W, C1, Ic1, V1, Vt1 ),
   resistor( SR6, R6, Ir6, V5, Vout ),
   resistor( SR5, R5, Ir5, V4, V5 ),
   resistor( SR2, R2, Ir2, V2, V3 ),
   resistor( SR4, R4, Ir4, Vin, V1 ),
   resistor( SR1, R1, Ir1, V1, V2 ),
   resistor( SR3, R3, Ir3, V1, Vout ).


%  Inserted MOS switches T1, T2 and ~T2 have states:
%     shrt - short circuit
%      dis - disconnected
%      res - resistor

bridge( shrt, _, _, V, V ).
bridge( dis, _, c(0,0), _, _ ).
bridge( res, R, I, V1, V2 ) :-
   resistor( ok, R, I, V1, V2 ).


%  Operational amplifier
%              _
```

```
%        Iina | \_
%   Vina -->--|+  \_
%             |    _>-->-- Vout
%   Vinb -->--|- _/   Iout
%        Iinb |_/

amplifier( ok, W, Vina, Vinb, Vout, Iina, Iinb, Iout ) :-
   A0 = 46.4,   % 46.416=pow(10,5/3),
   Rin = 0.5e6,
   R3 = 75,
   Voff = 0.0,
   Ibias = 0.0,
   Ioff = 0.0,
   F1 = 25,
   F2 = 1.0e6,
   R1 = 1000, R2 = 1000,
   constant_pi( Pi ),
   C1 = 1/(2*Pi*R1*F1),   % 6.37e-6,
   C2 = 1/(2*Pi*R2*F2),   % 1.59e-10
   C3 = 1.0e-12,
   cmp_add( Vina, c(Voff,0), Va ),
   resistor( ok, Rin, Ir, Va, Vinb ),
   cmp_add( Ir, c(Ibias,0), Iina ),
   cmp_add( Iinb, Ir, c(Ibias-Ioff,0) ),
   cmp_add( V0, Vinb, Va ),
   stage( A0, V0, R1, C1, W, c(0,0), V1 ),
   stage( A0, V1, R2, C2, W, c(0,0), V2 ),
   stage( A0, V2, R3, C3, W, Iout, Vout ).

stage( A0, V0, R, C, W, I, V ) :-
   Vgnd = c(0,0),
   cmp_mult_real( A0/R, V0, Is ),
   resistor( ok, R, Ir, Vgnd, V ),
   capacitor( ok, W, C, Ic, Vgnd, V ),
   cmp_add( Is, Ir, Isr ), cmp_add( Isr, Ic, I ).


%  Normal behavior of R and C

resistor( ok, R, I, V1, V2 ) :-
   cmp_add( DV, V2, V1 ),
   cmp_mult_real( R, I, DV ).
```

```
capacitor( ok, W, C, I, V1, V2 ) :-
   cmp_add( DV, V2, V1 ),
   cmp_mult_imag( C*W, DV, I ).


%  Weak fault models

resistor( ab(R), _, I, V1, V2 ) :-
   cmp_add( DV, V2, V1 ),
   cmp_mult( c(R,_), I, DV ).

capacitor( ab(C), W, _, I, V1, V2 ) :-
   cmp_add( DV, V2, V1 ),
   cmp_mult( c(_,C*W), DV, I ).


%  Complex numbers, addition and multiplication

cmp_add( c(Re1,Im1), c(Re2,Im2), c(Re1+Re2,Im1+Im2) ).

cmp_mult( c(Re,Im), Cmp0, Cmp ) :-
   cmp_mult_real( Re, Cmp0, Cmp1 ),
   cmp_mult_imag( Im, Cmp0, Cmp2 ),
   cmp_add( Cmp1, Cmp2, Cmp ).

cmp_mult_real( Re, c(Re0,Im0), c(Re*Re0,Re*Im0) ).

cmp_mult_imag( Im, c(Re0,Im0), c(-Im*Im0,Im*Re0) ).


%  Gain and phase

constant_pi( 3.14 ).

frequency( F, 2*Pi*F ) :- constant_pi( Pi ).

gain( c(Re1,Im1), c(Re2,Im2), Gain ) :-
  pow(10,G) = (Re2*Re2+Im2*Im2)*(1/(Re1*Re1+Im1*Im1)),
  Gain = 10*G.   % in original listing Gain = 10*G

phase( c(0,_), 0 ) :- true, !.
phase( c(Re,Im), Phase ) :-
   Re>0, true, !,
```

```
   constant_pi( Pi ),
   tan(Rad) = Im*(1/Re),
   Phase = 180/Pi*Rad.
phase( c(Re,Im), Phase ) :-
   Re<0,
   constant_pi( Pi ),
   tan(Rad) = Im*(1/Re),
   Phase = 180/Pi*Rad + 180.
```

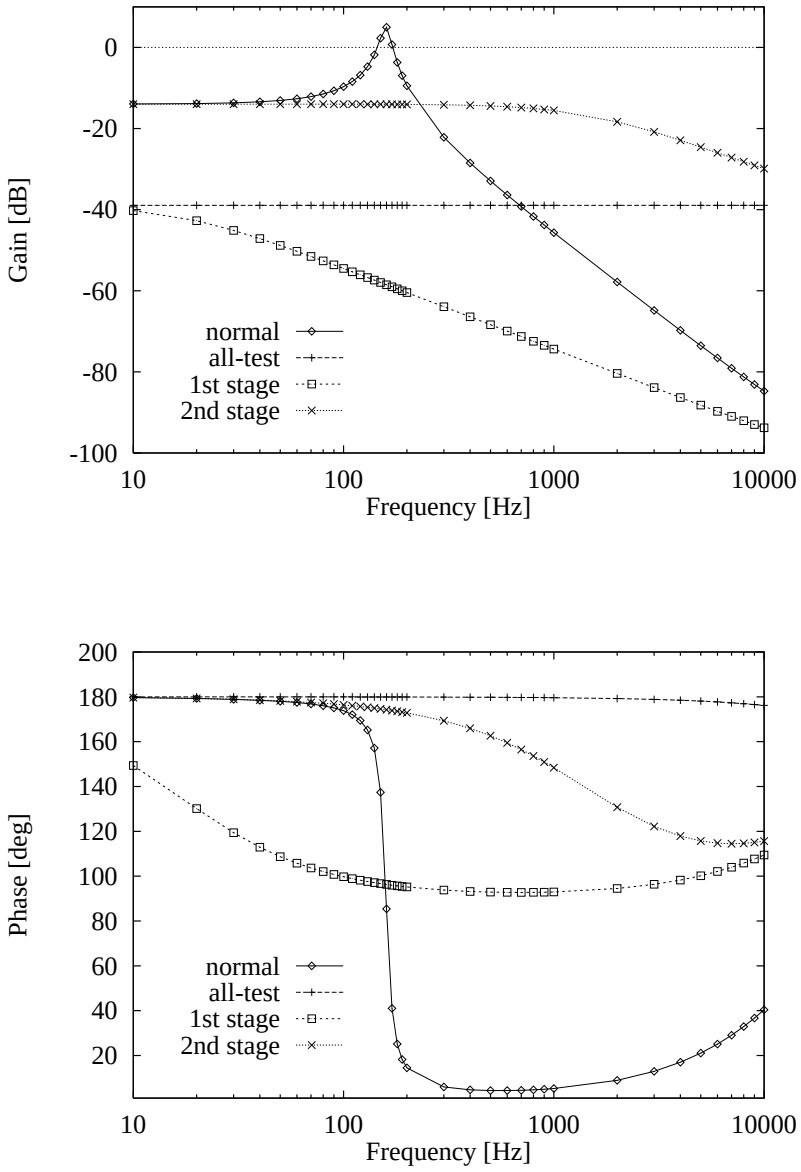

Figure 4: Gain and phase characteristics of the filter in all testing modes.