International Journal of Science Engineering and Advance Technology

Electronic Circuits Diagnosis Using Artificial Neural Networks

Yogesh Kumar
Research Scholar, Dr. K. N. Modi University, Rajasthan, INDIA
singhyogesh2002@yahoo.co.in

Abstract— When we expect about something that does not treat as it should be, we are initiating the process of diagnosis. Diagnosis is a commonly used activity in our everyday lives (Benjamins & Jansweijer, 1990). Complicated systems are always prone to faults or failures. In the simplest term, a fault is every change in a system that prevents it from operating in the proper manner. We define diagnosis as the task of identifying the cause and location of a fault manifested by some observed behaviour. Basically this is two-stage process: first the fact that fault has occurred must be recognized – this is referred to as *fault detection*. That is, in general, achieved by testing. Secondly, the nature and location should be determined such that appropriate remedial action may be initiated.

***Index Terms*—Fault Detection, Diagnostic System, Detectabilty, Observeability, fuzzyfication, Neural Networks.**

INTRODUCTION

The explosion of integrated circuit technology has brought with it some difficult testing problems. The recent growth of mixed analogue and digital circuits complicates the testing problem even further. It becomes more complicated to determine a set of input test signals and output measurements that will provide a high degree of fault coverage. There is also a timing problem of testing the circuits even on the fastest automated equipment.

The general structure of a diagnostic system is shown in Fig. 1. Signals u(t) and y(t) are input and output to the system, respectively. Faults and disturbances (here measurement errors) also influence the system under test, here denoted as the "Process", but there is no information about the values of these errors. The task of the diagnostic system is to generate a diagnostic statement S, which contains information about fault modes that can explain the behaviour of the Process. Note that the diagnostic system is assumed to be passive i.e. it cannot affect the Process itself.

The whole diagnostic system can be divided into smaller parts referred here to as tests.

These tests are also diagnostic systems, *DSi*. It is assumed that each of them generates diagnostic statement *Si*. The purpose of the decision logic (voting system) is then to combine this information in order to form the final diagnostic statement ***S***.

The number of possible faults in an electronic system may be large and can be located everywhere in the system. To diagnose in such conditions one frequently uses hierarchical approach where successive diagnostic statements are generated as the level of description of the system is lowered going down towards the fault itself (Ho et al., 2001; Sheu & Chang, 1997). This allows for smaller sets of faults to be considered at a time for the given hierarchical level. Modern automatic test pattern generator may support such concepts (Soma et al., 2001).

CONCEPTS OF DIAGNOSIS

Besides the human expert that is performing the diagnosis, one needs tools that will help, and ideally, perform the diagnosis automatically. Such tools are a great challenge to design engineers because, usually, the diagnostic problem is underspecified. In addition, it is a deductive process with one set of data creating, in general, unlimited number of hypotheses among which we try to find a solution. This is why the research community continues to be attracted by this problem (Bandler & Salama, 1985).

During the life-cycle of a product, testing is implemented in both the production phase and the implementation phase. We claim, however, that the sustainability of a product is strongly influenced by the design phase. So, to make a sustainable product, one should design the test procedure and synthesize test signals early in the design phase.

It is frequently possible to perform functional verification of the system. That, most frequently, happens when a small number of input/output terminals is present. In the majority of cases however, full functional testing becomes time consuming and is not acceptable. So, one applies defect-oriented (structural) testing, as will be discussed in more detail in what follows.

We consider testing to be: the selection of a set of defects regarded as the most probable, the description of a set of measurements, the selection of a set of testing points (or output signals) and most importantly, the synthesis of optimal testing signals that will be applied at the system inputs allowing for detectability and observability of the listed fault effects. Here, optimality means that one test signal covers as many faults as possible.

Selection of the type of measurements and testing points is specific to the circuit. One should stick to those measurements that are prescribed for functional verification. Specific measurements such as supply current monitoring are frequently adopted, too. Separate test points may be added in order to improve detectability or observability. Specific design for testability concepts can be applied.

Thanks to the advances in computational intelligence in the last decades new diagnostic paradigms have been applied based on: model-based concepts (Benjamins & Jansweijer, 1990); production rule based artificial intelligence (Pipitone et al., 1991); ANNs (Hayashi et al., 2002); genetic algorithms (Golonek & Rutkowski, 2002); and fuzzy-reasoning (Pous, et al., 2002); all trying to create an approach that contains properties that we might consider to be "intelligent behaviour".

In order to get an idea of why and how ANNs are applied to analogue electronic circuit diagnosis, the diagnostic concept (Fig. 1) will be elaborated in some detail first. It involves collaboration of design, test, and field engineers and the mutual distribution of responsibilities throughout the life cycle of an electronic product. We assume that field engineers are expected to react after a functional failure of the system. In order to diagnose such a system they need to be supplied with: testing equipment, a list of specific measurements to be done (including a set of signals and test points), and diagnostic software to process the measurement data. A similar set of data and tools would be given to a test engineer in a production-plant environment in order to evaluate the production yield and create feedback to process engineers when prototyping the circuit. We believe, however, design engineers are the most familiar with the product and the most qualified and capable to synthesize test and diagnostic signals, and procedures. This means the SBT (simulation before test) has to be applied to create fault dictionaries containing exhaustive lists of faults and corresponding responses. The fault dictionary is in fact a table representing the mapping from the fault list into a list of faulty (or possibly, fault-free) responses. In that way the diagnostic process becomes a search through the fault dictionary. Alternatively, modern diagnostic techniques using traditional artificial intelligence and reasoning methods typically fall into the simulation after test (SAT) category. This will increase the time spent on diagnosing systems at production time (Spina & Upadhyaya, 1997). SBT systems typically require more initial computational costs, but provide faster diagnosis at production time being the second reason why this concept was accepted here. We claim here that ANNs, being universal approximators (Scarselli & Tsoi, 1997), are the best way both to capture the mapping, and to search through the dictionary, thereby to perform diagnosis.

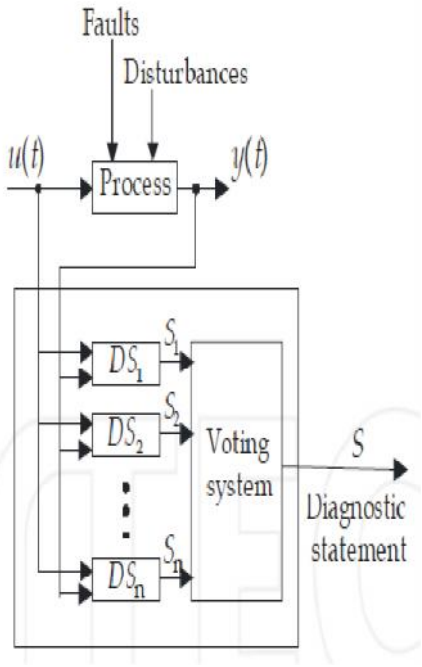

Fig. 1. A General Diagnostic System

DIAGNOSIS OF NONLINEAR DYNAMIC ANALOGUE CIRCUITS

Analogue electronic circuits are known to be difficult to test and diagnose. Apart from the huge number of possible faults, this difficulty is a consequence of the inherent nonlinearity of these circuits. Even linear circuits (having linear input-output signal interdependence) exhibit nonlinear relations between circuit parameters and the output response. There are no linear active networks. Active networks are nonlinear with nonlinear reactive elements.

They may be linearized and thought of as such in situations where signal and parameter changes are small in comparison to nominal values. When large parameter changes or even catastrophic faults occur (affecting the DC state), however, one must distinguish between linear and analogue circuits. This is not the case in most research reports bringing confusion into the subject.

Several concepts were applied to diagnosis of analogue networks. Among them we will first mention the ones relying on reasoning based on measured data and some

measure of distance between the response of the good circuit and the faulty one. Starting with the basic research reported in (Bandler & Salama, 1985) and (Milor & Visvanathan, 1989) several ideas were reported. In (Luchetta et al., 2002) the fault location phase is considered as an optimization problem where the parameter value is searched for in order to minimize the difference among the actual and simulated response. Linear circuits in the frequency domain are considered being characterized by symbolic functions.

Similarly in, (Catelani & Giraldi, 1998) applying SAT multiple faults may be diagnosed in linear circuits described by symbolic functions what is characterized as model based method. SBT based method for soft faults diagnosis in linear circuit was proposed in (Alippi et al., 2002) where harmonic analysis was used for selecting the most suitable test input stimuli and nodes by means of global sensitivity approach. In (Huang & Cheng, 2000) and (Yoon et al., 1998) passive circuits were diagnosed based on graph theoretical approach, and on pass and fail regions for the circuit poles and zeroes in the real-imaginary plane, respectively, while in (Chang, 2002) a Boolean decision scheme was proposed for the diagnosis of linear circuits described in the frequency domain. In order to diagnose multiple soft faults in the same type of circuits the Woodbury formula was applied to the modified nodal equation to construct the so called fault equation in (Liu & Starzyk, 2002). A decomposition method was proposed in (Starzyk & Liu, 2002) aiming to cope with circuit complexity. In one approach, small parameter changes were allowed in nonlinear circuits (Tadeusuewicz et al., 2002). Soft faults were considered only when linear programming method was used for diagnostic decisions.

Large parametric fault diagnosis was described in (Worsman & Wong, 2002) using piecewise linear models for DC analysis, and separate considerations were given for diagnosis of faults in the dynamic part of the network (considered linear) based on large change sensitivity computations. Further, in (Cota et al., 1999) the diagnostic method applied consists of injecting probable faults in a mathematical model of the linear circuit, and later comparing its output with the output of the real faulty circuit. Transfer functions transformed into the Z domain were created and fault injection was performed. In (Cherubal & Chatterjee, 1999) methodology based on linear regression model using prior circuit simulation which relates a set of measurements to the circuit's internal parameters was applied in order to solve for the circuit parameter values using iterative numerical techniques. Linear circuits in the frequency domain were diagnosed in (El-Yazeed & Mohsen, 2003) where the AC response to a set of sinusoidal input frequencies was calculated at selected test nodes. Prony's method was then utilized as a preprocessor to extract an optimal set of features representing nodal voltage waveforms. In (Dai & Xu, 1999) a solution to the same problem was proposed based on noise measurements.

Soft and hard faults (shorts and opens) in nonlinear dynamic circuit were diagnosed in (Pinjala et al., 2003). The procedure employs a statistical method of computing Mahalanobis distance to find defects in load board traces and components. Short list of defects was reported. A low-noise amplifier was diagnosed in (Liobe & Margala, 2004) by using digital signatures suitable for built-in self test design concepts. Hard and soft faults were diagnosed the former modelled as resistors having convenient values. A specific aspect of diagnosis is the number and location of the test points. Simply, we can say that internal test points should be avoided and measurements on the primary inputs and outputs are preferred. This is not only related to their automatic accessibility but also to the nature of the diagnostic reasoning. Namely, one looks for functionality to diagnose something, and the function is seen at the primary terminals. Of course, in order to compensate for the small number of test points more measurements with different types of applied signals are, generally, needed to extract complete information about the system behaviour. For complex analogue systems, however, hierarchical approaches based on decomposition (Ho et al., 2001; Sheu & Chang, 1997; Bandler & Salama, 1985; Starzyk & Liu, 2002) are inevitable provided that no propagation of the fault effect arises between partitions what is not easy to achieve. Of course, there are circuits that may be partitioned based on functionality known *a priori* from the design process as mentioned in the introduction.

Another aspect of fundamental importance is related to the choice of the output quantities that are to be measured. In most cases these are voltages at the output of the circuit under test (CUT) or at selected test points. It is shown, however, that measurement of the supply current (Iddq) may be successfully used for testing of both analogue and digital circuit (Dragic & Margala, 2002; Margala et al., 2002; Papakostas & Hatzopoulos, 1991; Bell et al., 1991; Zwolinski et al., 1996). This idea was used for diagnosis of analogue circuits using ANN that will be discussed later.

Several results were reported where the so called artificial intelligence concepts were applied to diagnosis of analogue circuits or at least linear ones. In (Savioli et al., 2005) method based on fault trajectory concept for fault diagnosis of analog linear continuous time networks, which relies on evolutionary techniques, where a genetic algorithm (GA) was coded to optimize test vector generation, was reported. GA was applied into (Golonek & Rutkowski, 2002) creation “transfer functions“ enabling creation of a new type of fault dictionary. The classical signature dictionary has been replaced by fault decoder based on transfer functions. In order to obtain a sharp diagnosis about the possible wrong component of the

circuit, a tool based on qualitative reasoning was used in (Pous et al., 2002). In particular, the results were refined by means of fuzzy techniques. This means that inputs, outputs, rules and the corresponding operators to combine them were defined. A production rule based concept was reported in (Pipitone et al., 1991).

ANNs have previously been applied to diagnosis (Spina & Upadhyaya, 1997; Materka, 1994; Rodrigez et al., 1994; Aminian & Aminian, 2000; He et al., 2002; Andrejević & Litovski, 2004; Aminian et al., 2002; Stopjakova et al., 2004; Yu et al., 1994; Collins et al., 1994; Catelani & Gori, 1996; Maidon et al., 1997; Yang et al., 2000). As in the case with the classical concepts, however, ANNs were predominantly applied to linear analogue circuits. In (Materka, 1994) feed-forward ANNs were used for parameter identification (soft fault diagnosis) of linear circuits. In (Rodrigez et al., 1994) linear power networks were diagnosed by feed-forward ANNs. In order to enhance the performance of the ANN applied for diagnosing of soft faults in linear active networks, in (Spina & Upadhyaya, 1997), new “criteria“ - a discriminating measure based on discrepancy of the autocorrelation function of the fault-free and the correlation function of the faulty and fault-free circuit, were introduced. The same problem was attacked in (Aminian & Aminian, 2000; Aminian et al., 2002) where the impulse response was analyzed by wavelet decomposition, principal component analysis, and data normalization preprocessors before introduced to the ANN. Soft faults were considered only. In (He et al., 2002) a method based on extraction of a “feature vector“ from the differences between vectors of node voltages of faulty and fault-free linear circuit for every fault was described. This feature vector is then presented to the ANN as a teaching session. Network tearing is applied in order to manage the circuit complexity in an 11 transistor bipolar circuit. Every partition was considered linear although catastrophic faults were present (e.g. transistor base disconnected). Two faults were diagnosed only. In (Andrejević & Litovski, 2004) a linear resistive circuit was diagnosed using feed-forward neural nets. Soft and hard faults (shorts and opens) were considered. Comparably large set of faults was taken into account. In the scheme presented in (Catelani & Gori, 1996) (one opamp/one capacitor, three resistors and two diodes) programmable function generator was used to generate the set of stimuli sequentially injected into the input of the CUT. Six test frequencies were chosen. For each stimulus the frequency response of the CUT has been considered and five Fourier components were measured at the output test point with the spectrum analyzer. For the purpose of diagnosis, four neural networks were used distance was to be learned by the ANN in order conclusions to be created on the origin of the fault. Bipolar analogue integrated circuits (Maidon et al., 1997) were diagnosed and their resistances determined from the magnitudes of the Fourier harmonics in the spectrum responses to a sinusoidal input test signal using multilayer perceptron ANN. The input vector to the ANN consists of the magnitudes of the Fourier harmonics of the response waveforms owing to the input stimulus, and the class represents the type of circuit faults, while the outputs map to resistance values of the faults. Probabilistic neural network was applied in (Yang et al., 2000). It is a four layer feedforward neural network that realizes the Bayes classifier. The ANN creates the probability that a circuit is faulty and points to the type of fault. In (Stopjakova et al., 2004) a large number of circuit versions was created by introducing sets of models for every separate fault. In fact, hard faults were considered while the opens and the shorts were modelled by resistors of variable resistivity. Then statistical properties of the time domain response (in this case the supply current) to a pulse excitation were extracted in order to create knowledge of the fault to fault-effect mapping. The supply current was successfully used for diagnosing gate oxide shorts in CMOS circuits by the help of ANNs in (Yu et al., 1994; Collins et al., 1994). After introducing a fault model of the MOS transistor built as a series connection of two MOS transistors with a common gate (i.e. considering this as a soft fault), several faults per transistor (for all transistors in an 11 transistor operational amplifier) were created by changing the possible position of the gate short relative to the source-to-drain ends of the channel. Sinusoidal and ramp signals were used for creation of a fault dictionary in an SAT method. The response i.e. the supply current was sampled to give a series of values used to train the feed-forward (Yu et al., 1994) and a Cohonen (Collins et al., 1994) neural network. Two examples of fault diagnosis in non-linear dynamic circuit are - one refers to an analogue circuit, and the second to the mixed-mode circuit.

We describe the results of applying feed-forward ANNs to the diagnosis of non-linear dynamic electronic circuits with no restriction on the number and type of faults. This method is based on fault dictionary creation and using an ANN for data compression by memorizing the table representing the fault dictionary. Only DC and small signal sinusoidal excitations will be applied, so preserving the usual measurement procedure for generating the data given in a component’s and/or a circuit’s data-sheets. The ANN so created is, consequently, used for diagnosis by applying to it the signals obtained by measuring the faulty network. This process may be considered as looking-up a fault in the fault dictionary.

FAULT DICTIONARY CREATION AND APPLICATION EXAMPLE

In order to describe the way in which the fault dictionary was created, the circuit in Fig. 2 is used as an

application example. This is a CMOS operational amplifier consisting of seven transistors. To our knowledge this example belongs to the category of the most complex ones reported, both from the number of circuit elements point of view and the number of faults inserted. Note that three (nonlinear) capacitors are associated with every transistor totalling the number of nonlinear circuit elements to 28 but, for the sake of simplicity, are not shown in the figure. In order to emphasize the method as such, while not offering a full solution of the diagnostic problem for this circuit, having in mind abundance of possible faults, a reduced set of faults was considered. To this end only single transistor faults are sought. That, of course will not affect the generality of the ideas implemented in the next.

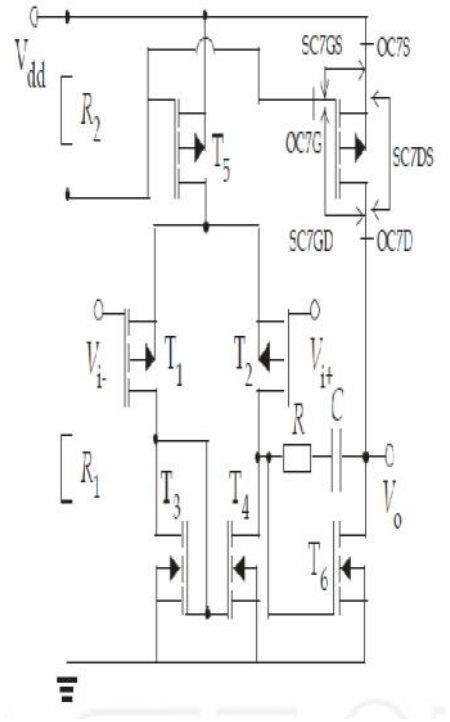

Fig. 2. Operational amplifier circuit.
SC=Short Circuit, OC=open circuit

Ten faults per transistor, six catastrophic and four parametric were added to the dictionary. As shown in the figure (using T7 as an example) there exist three open-circuit faults (OC) and three short-circuit faults (SC) per transistor (for example, OC3G stands for open gate of transistor T3, and SC1DG stands for drain and gate shorted in transistor T1, Table 1.). As opposed to (Stopjakova et al., 2004) and some others, the shorts (some of them behaving as bridging fault) and opens were really implemented instead of resistors modelling them. To effectively simulate perfect short and opens we used our model of the ideal switch (Mrčarica et al., 1999) what is not possible in the SPICE simulator. Of course, there was no obstacle for us to use resistors to model shorts and opens. Simply, what we did, we considered satisfactory. In addition, two faulty values for every channel length (±20%) (denoted as L+ and L- in Table 1.), and two for every channel width (±20%) (denoted as W+ and W- in Table 1.) were introduced, totalling 10 faults per transistor. The soft faults considered here are expected to model design errors and, in a specific way, gate oxide short having in mind the fault model reported in (Yu et al., 1994). For the whole circuit this gives a set of 70 faults observed. The DC output values (*V* oDC *m*) were first obtained by simulation. Here *m*=0,1,2,…,69 stands for the fault code. In addition, the frequency response of the circuit (the non-inverting input terminal was excited by a signal of amplitude 1mV) was obtained by simulation over a fixed frequency range in order to extract two response parameters: the nominal gain (*Am*) and the 3-dB cut-off frequency (*f* 3dB *m*). For the example given, we considered this signature to be satisfactory complex. If additional fault need to be used one might think on additional measurements such as supply current. Note that, for the DC supply current point of view, the fault effects of most open faults at sources and drains in series connected transistors, may have equivalent signatures.

Type	A_m	f_{3dBm} [MHz]	V_{oDCm} [V]	Code (m)
FF	419	0.01527	0.127	0
1L+	0.0053	6.791	0.0497	37
OC1G	0.047	501.187	0.127	49
OC3G	0.049	544.042	0.093	47
SC1DG	0.042	320.440	0.0458	6
SC2DS	0.071	312.071	3.3	27
SC5DS	0.656	0.57	0.0186	55
6W-	5770	0.0018	0.2146	13
OC5D	0.056	507.298	3.3	25
SC5GS	0.109	0.036	0	2

Table. 1. Part of the fault dictionary for the circuit of Fig 2 The faults are chosen at random

Because of the nonlinearity of the circuit, every fault is expected to change the transistor's quiescent points. Consequently, new linear transistor-models are created by SPICE-like program and used for frequency domain performance extraction for each fault. In order to find the new quiescent point for every fault, we have to insert the fault i.e. to create a faulty model of the circuit for DC analysis. This procedure is described elsewhere (Milovanović & Litovski, 1991; Milovanović & Litovski, 1994) and will not be discussed here. Fault SC3DG is untestable because of the existing connection between the gate and drain of T3. This reduces the fault dictionary to 69 elements. Therefore, the fault dictionary created here has four columns containing the set of circuit performances i.e. the signatures and the fault code: { *V*

oDC *m*, *Am*, *f* 3dB *m*, *m*}. First three items in a row are considered inputs to the neural network, while the fault code is learnt as an output. The fault coding is an important issue. In fact, some defects exhibit very similar effects. So, input data (signatures) can have very close numerical values, and if the output values (defect codes) were also similar, the network could not always be trained successfully. Such an example is given in Fig. 3. Here the signatures of three faults are compared. By careful inspection we can see that only the *f* 3dB values suggest a difference between the fault effects. Faults are coded randomly, so that faults with similar effects are unlikely to have similar codes. This approach is proven to be good, because the way of coding influenced the training time, and also, the training error. Part of the fault dictionary for the circuit in Fig. 2, is given in Table 1., where *m*=0 denotes the fault-free circuit.

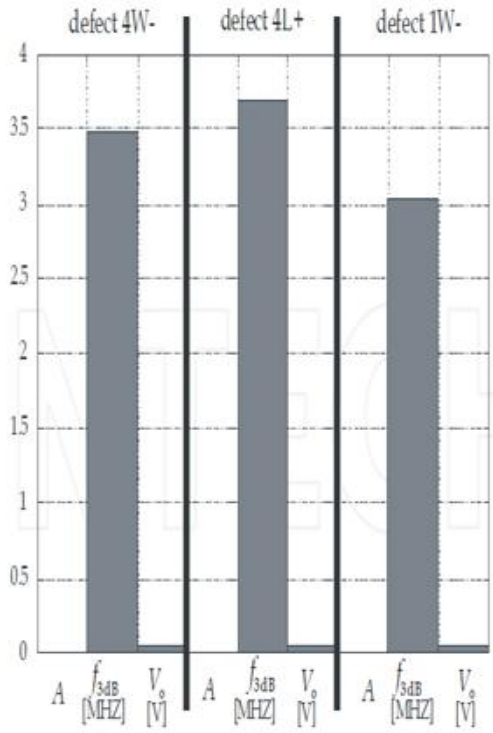

Fig. 3. Fault effects of faults 4W-, 4L + & 1W-

Here we come to an additional issue concerning the applicability of rule-based approaches to the diagnosis of systems of this kind. Because of the similarity of the circuit performance in the presence of different faults, no set of rules can be established to distinguish between these three faults. Furthermore, by inspection of Table 1., we can see that the performance values cover a broad range. For example, for the voltage gain, the smallest value in the Table is 0.0053, and the largest is 5770. So, we cannot establish a rule defining the difference that occurs as a consequence of the presence of a certain fault. Thus rule-based approaches are impractical for systems exhibiting responses as continuous functions. Note, in addition, that we expect noisy data to be obtained when field measurements are performed for diagnosis which further complicates the creation of any rules. We may claim that similar is related to use fuzzyfication in order to boost the difference among signatures. To go further, this puts in a similar prospective the simulation-after-test i.e. the model based concept. Namely, in this concept one is supposed to create a set of hypotheses that will be checked against the measurement data by successive simulations of the circuit under test at the repairing site. Having in mind Fig. 3., however, we do not believe the creation of a qualified set of hypotheses is an achievable task. The fault dictionary can be further reduced in size by processing the *ambiguity groups* or the groups of equivalent faults. According to (Manetti & Piccirilli, 2003) "an ambiguity group is, essentially, a group of components where, in case of fault, it is not possible to uniquely identify the faulty one". Here, we can say that an ambiguity group consists of a set of *faults* that propagate identical signatures to the output, making the faults detectable and the circuit testable, but no distinction between the individual faults is possible making them undiagnosable. Table 2. shows all ten ambiguity groups for this example, systematically collected after simulation. The faults italicized in Table 2. represent the same topological connection in the circuit, so the effect would be expected to be the same.

Ambiguity group	Faults included	A	f_{3dB} [MHz]	V_{oDC} [V]
1	OC1D	0.31	20000	0.0179
	OC1S			
2	OC3D	0.041	365.8	3.3
	OC3S			
3	OC4D	0.303	20000	0.0458
	OC4S			
	SC4GS			
	SC3GS			
	SC3DS			
4	OC5D	0.056	507.298	3.3
	OC5S			
5	OC6D	0.063	0.039	3.3
	OC6S			
6	OC7D	$A \to \infty$	Indeterminate	0
	OC7S			
7	SC1GS	0.055	515.993	3.3
	SC2GS			
8	SC5GS	0.109	0.036	0
	SC7GS			

Table 2. Ambiguity groups and fault effects

A specific ambiguity group is the case when the gain (A) although small ever rises within the given frequency range. This is denoted as $A \rightarrow \infty$. The 3-dB cut-off frequency is in these cases indeterminate. Some of defects exhibiting this property, however, have different V oDC values, so they can be distinguished. In these cases, to avoid use of infinite numbers during the training of the ANN we assigned a value of 1000 to the gain. This is the case when simulating defects OC7S and OC7D, but since these defects produce completely the same effect, they form ambiguity group number 6. A similar situation occurs when the gain is almost zero, A=0. The 3-dB cut-off frequency is then again indeterminate. Ambiguity group number 9 covers three such defects, with the same V oDC. Only one representative of each ambiguity group was included in the fault dictionary. From Table 2., we find that the complete fault dictionary in this case has 70-1-24+10=55 elements. With three pieces of data for each fault, the neural network input structure was restricted to three input terminals. The ANN diagnoses the fault by outputting the fault-code (m) as a signal level, so we needed only one output neuron. The number of hidden neurons, n, was found by trial and error after several iterations starting with an estimation based on that in (Baum & Haussler, 1989). The goal was to find the optimum n that leads to a satisfactory classification even with noisy excitations. Using too many neurons would increase the training time, but using too few would starve the network of the resources needed to solve the problem. Also, an excessive number of hidden neurons may cause the *overfitting* problem (Masters et al., 1993), when a network has so much information capability that it learns insignificant aspects of the training sets, irrelevant to the general population. In practice, 30 hidden neurons were used. After successful training, no mistakes were observed for all 55 faults.

Code	A_j	f_{3dBj} [MHz]	V_{oDCj} [V]	ANN response
0	419	**0.0145**	0.127	0.02128
1	129.6	**0.0248**	0.079	1.09057
2	0.109	0.036	**-0.05**	2.01405
3	6028	**0.001575**	0.1712	2.93868
5	4453	**0.002415**	1.0255	5.03203
6	**0.0441**	320.44	0.0158	6.03224
9	1000	1000	0.05	9.0707
10	**0.043**	365.8	3.3	10.0278
12	1000	1000	3.39	12.1771
13	5770	**0.00171**	0.2146	13.2376
16	8220	**0.00197**	0.4876	16.031
18	0.32	1000	**0.133**	17.8458
20	0	1000	3.46	20.4409
21	0.83	1000	3.46	20.6497
25	**0.0588**	507.298	3.3	25.0605
26	11.739	**0.114**	0.127	26.0098
27	0.071	312.071	3.46	27.0091
34	5809	**0.00169**	0.1811	33.7541
35	209	0.0237	**0.115**	35.47

Table 3. Inputs with noise and ANN responses

The generalization property of the network was verified by supplying noisy data to its inputs. This is presented in Table 3. 30 samples were examined. For each sample, one input (boldfaced in the Table 3.) is incremented by +5% or -5%, representing noise generated during the measurement process. The responses of the network are given in the last column of the table. The ANN response was considered to be correct (i.e. acceptable) when its value was in the range [(m-0.5), (m+0.5)]. We can see that all faults can still be diagnosed though some with difficulties (for m=20, 35, and 54).

In the previous text we have presented our first results in the development of a new technique for fault diagnosis of nonlinear dynamic circuits. The method we proposed may be summarised as follows. In applying ANNs to the diagnosis of nonlinear dynamic electronic circuits, as described, we have demonstrated the implementation of the method and a set of results. These results vindicate the technique. In further work we intend to resolve the elements of ambiguity groups. In addition, more complex systems will be considered and larger fault dictionaries generated. Consequently additional measurements will be needed in order to keep the number of test points low. This will, hopefully, allow for implementation of these ideas to diagnosis of mixed-signal circuits.

FAULTS IN THE SPECIFIC CIRCUIT DESIGN

As an example of a complex circuit, the sigma-delta modulator in Fig. 4. is chosen (Xu & Lucas, 1995). This is a mixed-signal circuit, having both analogue and digital elements. Switches in the circuit are modelled as truly ideal switches, with zero resistance for closed switch and infinite resistance for open switch. Simulations are performed using *Alecsis* (Glozić, 1994) simulator

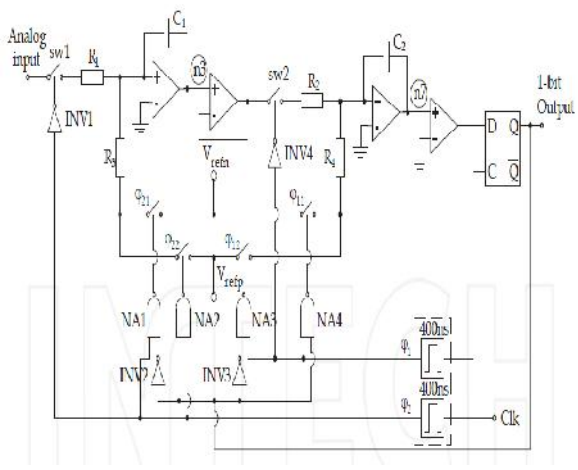

Fig. 4. Sigma-delta modulator architecture

The integrator charging time is invariable with respect to clock rate in order to keep the gain constant. This means that the analog switch must be turned on for fixed time duration regardless of clock rate. This is achieved by using monostable multivibrator as a fixed-width pulse generator in the circuit. The monostable multivibrator between the clock input and switch control block functions as a pulse generator to produce control signals of fixed time duration. Fig. 5 shows reaction of the system when the input is excited by a ramp signal.

We consider in this chapter defects only in the digital part of the circuit. There are two types of defects observed: catastrophic defects and delays of rising and falling edge of output digital signals. These delay defects are neither catastrophic, nor parametric, because there is no change in circuit topology, and no change in element values.

Digital signal can be “stuck-at-1” or “stuck-at-0”. In the circuit in Fig. 4., analogue switches are controlled by digital signals, so there are pairs of the same fault effects, such as: the effect is the same when the switch is stuck at ON (OFF) and the logic circuit’s output is “stuck-at-1” (“stuck-at-0”). So, we will consider hard faults (which refer to the analogue part of the circuit) as stuck switches (Andrejević et al., 2006). The cases when switches in the feedback loop (ϕ11, ϕ12, ϕ21, ϕ22) are permanently closed are excluded, because voltage references *V* refp and *V* refn would be shorted in such cases.

Having in mind that clock period in the circuit is 1.2μs (half period is 600ns), we examined effects of delays not greater than 400ns. In fact, effects of rising edge delay are simulated for values of delay: 100ns, 250ns, 400ns, and for falling edge, we simulated smaller values: 50ns, 100ns, 150ns. The goal was to determine how these delays influence the output, and whether different delay values produce different outputs (Andrejević et al., 2006). All digital gates are examined (4 inverters and 4 nand circuits). The first conclusion was that delays in the circuit of inverter 2 (INV2) do not influence output signal, meaning that output is not changed. Further, there exist groups of delays causing the same effect. Such groups are known as ambiguity groups, and they are listed in Table 4. The first four groups show the same effect of delays. In the second column of the Table 4, defects causing the same effects are named, and accordingly, third column presents that same effect (signature). The fifth ambiguity group is in a way different. The members of that group are both catastrophic and delay defects. Note that only one representative of each group is given in the fault dictionary.

Ambiguity group	*Defect type*	*Signature*
1	na3(tf=50ns)	104108210
	na4(tr=250ns)	
2	na3(tr=400ns)	102104208
	na4(tr=400ns)	
3	na4(tf=100ns)	404210240
	inv3(tf=100ns)	
4	FF	20440480A
	inv2(tr=50ns)	
	inv2(tr=100ns)	
	inv2(tr=150ns)	
	inv2(tf=50ns)	
	inv2(tf=100ns)	
	inv2(tf=150ns)	
5	φ21OFF	000000000
	na1(tf=150ns)	
	sw1ON	

Table 4. Ambiguity groups

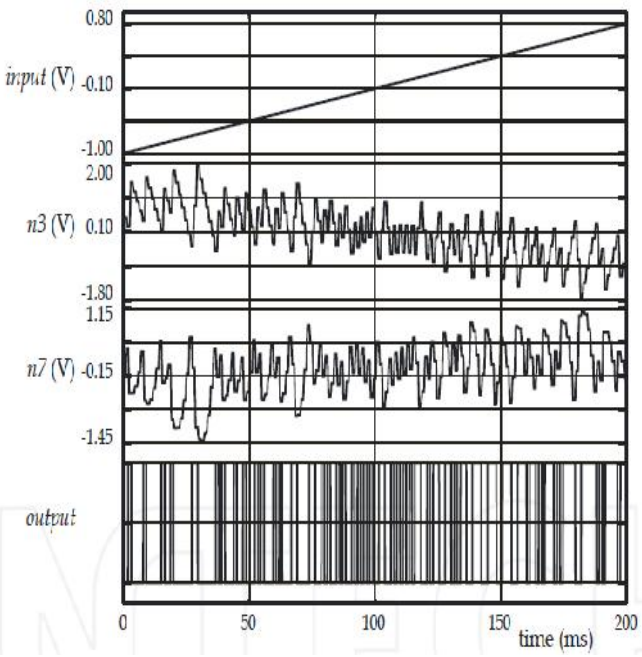

Fig. 5. Simulation results for ramp excitation

Fault dictionary can be created using the response of the circuit to an input ramp signal. The circuit output value is registered after every clock period, so these output digital values form the output signature.

ANN was trained for modelling the look-up table. It is a feed-forward neural network with one hidden layer. The signatures are inputs to the network, and the fault code is network output to be learned. It means that the neural network has 9 inputs (one input per hexadecimal digit) and one output neuron. Hexadecimal values are presented as decimal when they are inputs to the network. After learning was completed, the number of hidden neurons in the resulting ANN was 10, what was found by trial and error after several iterations starting with an estimation based on (Masters et al., 1993; Baum & Haussler, 1989).

The structure of the obtained ANN is verified by exciting the ANN with faulty inputs. Responses of the ANN show that there were no errors in identifying the faults what is presented in Table 5. Only negligible discrepancies may be observed.

CONCLUSION

In applying ANNs to the diagnosis of nonlinear dynamic electronic circuits, as described, we have demonstrated the implementation of the method and a set of results. In the second part of this chapter, effects of delay and catastrophic defects in sigma-delta modulator were examined. The diagnosis was successful. Accordingly, we may conclude that ANNs are convenient and powerful means for diagnosis, and, what is important, realizable as a hardware that may be as fast as necessary to follow the changes of the system's response in real time.

Defect type	Defect code	ANN output	Defect type	Defect code	ANN output
FF	0	-0.000215	inv4(tf=100ns)	23	22.9998
sw1OFF	1	0.999861	sw2ON	24	23.9997
inv1(tr=150ns)	2	1.99969	inv3(tr=150ns)	25	24.9999
na1(tr=400ns)	3	2.99981	inv4(tr=100ns)	26	26.0009
inv1(tf=50ns)	4	3.99985	na3(tr=250ns)	27	27
na1(tf=100ns)	5	4.99988	inv4(tr=150ns)	28	28
na2(tf=50ns)	6	6.00003	na1(tr=100ns)	29	29.0001
na3(tr=400ns)	7	7.00006	na3(tf=100ns)	30	30
na3(tf=50ns)	8	7.99998	inv4(tf=50ns)	31	31.001
na3(tf=150ns)	9	8.99991	na4(tf=100ns)	32	32.0003
na4(tr=100ns)	10	10.0004	sw2OFF	33	33.0021
o21OFF	11	10.9998	inv1(tf=150ns)	34	34
inv1(tf=100ns)	12	11.9997	na2(tf=150ns)	35	34.9998
na2(tr=100ns)	13	12.9994	inv1(tr=50ns)	36	36
na2(tf=100ns)	14	13.9997	inv3(tf=150ns)	37	37.0001
inv3(tr=100ns)	15	15	inv4(tf=150ns)	38	37.9998
na3(tr=100ns)	16	15.9996	na2(tr=250ns)	39	39.0024
inv1(tr=100ns)	17	16.9998	na1(tr=250ns)	40	40.0015
na2(tr=400ns)	18	18	inv3(tf=50ns)	41	40.9997
inv4(tr=50ns)	19	19.0018	φ12OFF	42	41.9996
na1(tf=50ns)	20	19.9997	φ22OFF	43	42.9998
na4(tf=50ns)	21	20.9999	inv3(tr=50ns)	44	43.9997

Table 5. ANN output results.

REFERENCES

[1] Alippi, C., Catelani, M., Mugnaini, M. (2002). SBT Soft Fault Diagnosis in Analog Electronic Circuits: A Sensitivity-Based Approach by Randomized Algorithms, *IEEE Transactions on Instrumentation and measurement*, Vol. 51, No. 5, pp. 1116-1125.

Aminian, M., and Aminian, F. (2000). Neural-network based analog-circuit fault diagnosis using wavelet transform as preprocessor, *IEEE Transactions on CAS – II: Analog and Digital Signal Processing*, Vol. 47, No. 2, February 2000, pp. 151-156.

[2] Aminian, F., Aminiam, M., Collins, H. W. (2002). Analog Fault Diagnosis of Actual Circuits Using Neural Networks, *IEEE Trans. On Instrumentation and Measurement*, Vol. 51, No. 3, June 2002, pp. 544-50, ISSN 0018-9456.

[3] Andrejević, M., Litovski, V. (2004). ANN application in electronic diagnosis-preliminary results , *Proceedings of IEEE 24th International Conference on Microelectronics MIEL*

[4] Andrejević, M., Litovski, V., Zwolinski, M. (2006). Fault Diagnosis in Digital Part of Mixed-Mode Circuit, *Proceedings of IEEE 25th International Conference on Microelectronics MIEL 2006*, pp. 437-440, ISBN 1-4244-0116-X, Niš, Serbia, May 2006.

Andrejević, M., Litovski, V. (2006a). Fault Diagnosis in Digital Part of Sigma-Delta Converter, *Proceedings of Neurel 2006 Conference*, pp. 177-180, ISBN 1-4244-0432-0, Beograd, Serbia, September 2006.

[5] Andrejević, M., Litovski, V. (2006b). Fault Diagnosis in Analog Part of Mixed-mode Circuit, *VI Symposium on Industrial Electronics (INDEL 2006)*, pp. 117-120, Banja Luka, Bosnia and Herzegovina, November 2006.

Andrejević, M., Petrović, V., Mirković, D., Litovski, V. (2006). Delay Defects Diagnosis Using ANNs, *Proceedings of L Conference of ETRAN*, pp. 27-30., Belgrade, Serbia, June 2006

Andrejević, M. (2006). Artificial neural networks application in electronic circuits diagnosis, *PhD Thesis*, University of Niš, Serbia, July 2006, (in Serbian).

Bandler, J., and Salama, A. (1985). Fault diagnosis of analog circuits, *Proceedings of the IEEE*, Vol. 73, No. 8, pp. 1279-1325, ISSN 0018-9219.

[6] Baum, E. B., and Haussler, D. (1989). What size net gives valid generalization, *Neural Computing*, Vol. 1, pp. 151-60.

Bell, I. M., Camplin, D. A., Taylor, G. E., Bannister, B. R. (1991). Supply Current Testing Of Mixed Analogue And Digital ICs, *Electronics letters*, Vol. 27, No. 17, pp. 1591-1583, ISSN 0013-5194.

Benjamins, R., Jansweijer, W. (1990). Toward a competence theory of diagnosis, *IEEE Expert*, Vol. 9, No. 5, pp. 43-52, ISSN 0885-9000.

[7] Catelani, M. and Gori, M. (1996). On the application of neural networks to fault diagnosis of electronic analog circuits, *Measurement*, Vol. 17, pp. 73-80.

Catelani, M., Giraldi, S. (1998). Fault diagnosis of analog circuits with model based techniques, *IEEE Instrum. Meas. Techn. Conference,* Vol. V 1, pp. 501-504.

Chang, Y.-H. (2002). Frequency-domain grouping robust fault diagnosis for analog circuits with uncertainties, *International Journal of Circuit Theory and Applications*, Vol. 30, pp. 65-86, ISSN 0098-9886.

[8] Cherubal, S., Chatterjee, A. (1999). Parametric Fault Diagnosis for Analog Systems Using Functional Mapping, *Proceedings of Design, Automation and Test in Europe (DATE '99),* p. 195, Munich, Germany.

[9] Collins, P., Yu, S., Eckersaal, K. R., Jervis, B. W., Bell., I. M., and Taylor, G. E. (1994). Application of Cohonen and Supervised Forced Organization Maps to Fault Diagnosis in CMOS Opamps, *Electronics letters*, Vol. 30, No. 22, pp. 1846-1847, ISSN 0013-5194.

[10] Cota, É. F., Carro, L., Lubaszewski, M. (1999). A Method to Diagnose Faults in Linear Analog Circuits Using an Adaptive Tester, *Proceedings of Design, Automation and Test in Europe Conference, DATE '99,* pp. 184-188, Munich, Germany.

[11]

[12] Dai, Y., and Xu, J. (1999). Analog circuit fault diagnosis based on noise measurements, *Microelectronics and Reliability*, Vol. 39(8), pp. 1293-1298, ISSN: 0026-2714.

[13] Dragic, S., and Margala, M. (2002). A 1.2 V Built-in Architecture for High Frequency OnLine Iddq/delta Iddq Test, *Proc. IEEE Computer Society Annual Symposium*