Modélisation de fautes et diagnostic pour les circuits mixtes/RF nanométriques

Ke Huang

UNIVERSITÉ DE GRENOBLE

THÈSE

Pour obtenir le grade de

DOCTEUR DE L'UNIVERSITÉ DE GRENOBLE

Spécialité : **Micro et Nano Électronique**

Arrêté ministériel : 7 août 2006

Présentée par

Ke HUANG

Thèse dirigée par **Salvador MIR**
et co-encadrée par **Haralampos-G. STRATIGOPOULOS**

Préparée au sein du **Laboratoire TIMA**
Dans **l'École Doctorale Électronique, Électrotechnique, Automatique et Traitement du Signal (E.E.A.T.S)**

Modélisation de fautes et diagnostic pour les circuits mixtes/RF nanométriques

Thèse soutenue publiquement le **16 Novembre 2011**,
devant le jury composé de :

M. Philippe BENECH
Professeur, Université Joseph Fourrier, Président
M. Hans-Joachim WUNDERLICH
Professeur, Université de Stuttgart (Allemagne), Rapporteur
Mme. Luz BALADO
Associate Professor, Université Polytechnique de Catalogne (Espagne), Rapporteur
M. Bram Kruseman
Senior Enginneer, NXP Semiconductors (Pays-Bas), Examinateur
M. Salvador MIR
Directeur de recherche, CNRS Grenoble, Directeur de thèse
M. Haralampos-G. STRATIGOPOULOS
Chargé de recherche, CNRS Grenoble, Co-encadrant de thèse

Remerciements

Je tiens tout d'abord remercier chaleureusement mon directeur de thèse, M. Salvador MIR, Directeur de rechercher CNRS ainsi que mon co-encadrant de thèse M. Haralampos STRATIGOPOULOS, Chargé de recherche CNRS pour leurs encadrements, leurs conseils précieux, leurs discussions, leurs encouragements et le temps qu'ils ont consacré durant ces trois années de thèse. Je tiens également à remercier Mme. Dominique BORRIONE, Directrice du Laboratoire TIMA, de m'avoir accueilli dans le laboratoire.

Je voudrais remercier sincèrement M. Philippe BENECH, Professeur à l'Université Joseph Fourrier, d'avoir accepté de présider le jury de ma thèse. Je voudrais remercier également M. Hans-Joachim WUNDERLICH, Professeur à l'Université de Stuttgart et Mme. Luz BALADO, Associate Professor à l'Université polytechnique de Catalogne, qui m'ont fait l'honneur d'accepter d'être les rapporteurs de cette thèse. Je voudrais aussi remercier M. Bram KRUSEMAN pour sa participation au jury de thèse.

J'aimerais également remercier M. Alexandre CHAGOYA, Ingénieur de support des logiciels de simulation au CIME, pour sa gentillesse et sa patience durant toute la période de ma thèse.

J'aimerais également remercier tous les partenaires du projet européen TOETS d'NXP Pays-Bas de nous avoir fourni les données.

Je remercie vivement tous les membres est anciens membres du groupe RMS : Fabio, Rafik, Nourredine, Laurent, Yoan, Asma, Rshdee, Brice (et plein d'autres gens que je n'ai pas pu citer leurs noms!) ainsi que tout les personnels du laboratoire TIMA d'avoir créé une ambiance très sympathique. Je remercie notamment mon collègue Louay ABDALLAH pour tous ses aides, ses encouragements et tous les échanges enrichissants pendant la thèse. Je remercie également mes amis Hai, Wenbin et Yi pour la préparation du pot de thèse.

Un grand merci à ma famille et mes amis pour leurs soutiens et patiences durant la thèse, notamment à ma femme Ting pour tous ses soutiens pendant les moments difficiles.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Introduction

Recent advances in Very Large Scaled Integrated circuits (VLSI) have continued to shrink device geometries at a steady rate in accordance with Moore's Law. It is often desirable to manufacture Integrated Circuits (ICs) on advanced technologies due to the substantial increase in density integration and reduction in power consumption. Continued scaling of semiconductor devices would reduce the cost per function 25-29% each year and promote market growth for ICs (averaging 17% each year) [1]. Advancement in technology allows for the non-digital functionalities (e.g., RF communication, power control, passive components, sensors, actuators) to migrate from the system board-level into the chip-level or package-level, and ultimately into 3D ICs. Figure 1.1 shows the general semiconductor industry trends.

However, this advancement has also been accompanied by increasing variations in the performances of fabricated circuits. Performances are very susceptible to natural manufacturing process variations. For example, varying impurity densities, gate oxide thickness, and junction depth variations may cause transistor parameters such as threshold voltage V_{th} to shift resulting in performance degradation. Furthermore, as transistor density increases, defects and imperfections created during the manufacturing process can cause device failures.

Integration of both analog and digital parts in a reduced chip size poses key challenges for test. It is very important to verify the functionality of devices after fabrication and in the field of operation, which is the role of test. Figure 1.2 shows a brief description of a typical design flow of an analog IC. Testing analog devices consists of verifying the specifications which are often defined by lower/upper measurement limits. With continuous shrinking of device geometries, analog IC test becomes a severe challenge nowadays due to limited accessibility and observability of internal nodes. According to the time at which the test is applied, it can be classified into characterization, production and on-line test in the field. The goal of characterization test is to verify thoroughly at the design stage the design weaknesses, the reliability of devices with regard to process variations and the eventual failure so as to make the final design as robust as possible. Production test verifies the specifications of devices at a high production volume level. Since the number of devices to be tested is large, production test must be as fast and economical as possible. Finally, on-line test is applied during the lifetime of devices in

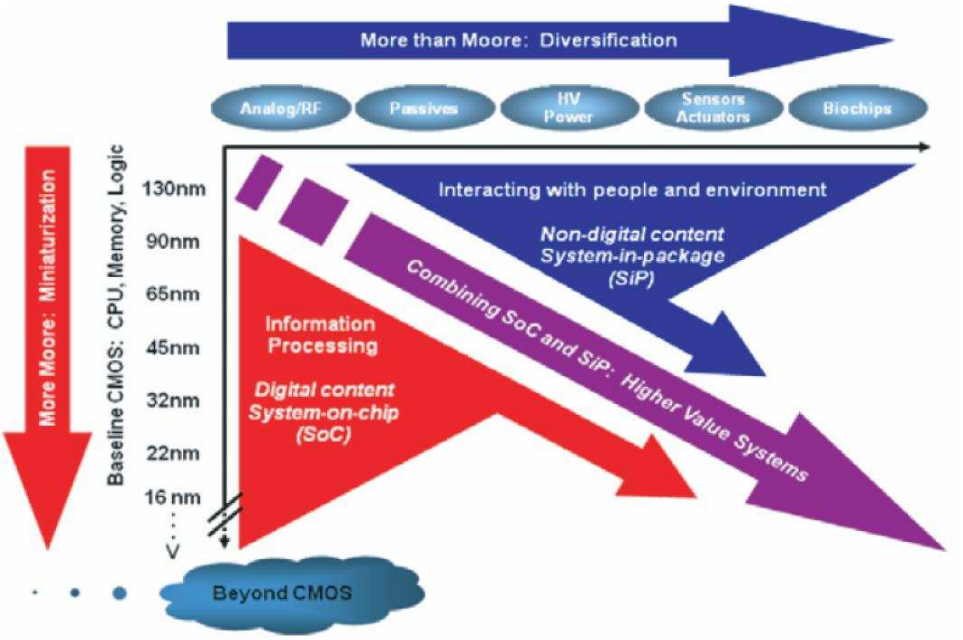


Figure 1.1: Semiconductor industry trends [1].

order to guarantee their reliability against aging phenomena and harsh environmental conditions. Examples of such applications include medical and automotive electronic systems. Failures can occur at any stage of the lifetime of the IC, as indicated in Figure 1.2. Failure mechanism analysis (i.e. fault diagnosis) is essential to reduce the time to market, enhance yield and expand the safety features.

1.2 Motivation

Fault diagnosis of ICs has grown into a special field of interest in semiconductor industry. At the design stage, the test development time cycle is affected by a number of factors. Unlike in digital parts where advance CAD tools exist to verify the design, the lack of automation in the design of analog/mixed devices makes it time-consuming and the design must be verified with fabricated prototypes, which increases design iterations. Diagnosing the sources of failures in IC prototypes at this stage is very critical to reduce design iterations in order to meet the time-to-market goal. Failure at this stage is related to the incomplete simulation models and the aggressive design techniques that are being adopted to exploit the maximum of performances out of the current technology.

In a high-volume production environment, diagnosing the sources of failures can assist the designers in gathering information regarding the underlying failure mechanisms. Identifying failure mechanisms is very important to prevent economic consequences of reduced yield in production. Traditional failure analysis (FA) methods consist of observing failures by their optical characteristics such as light-emission methods, picosecond imaging or laser probe methods. However, the time required for applying these methods has become intolerable with the increasing reduction in feature sizes and the high complexity of modern IC integration [15]. In order to determine the root cause of failure and

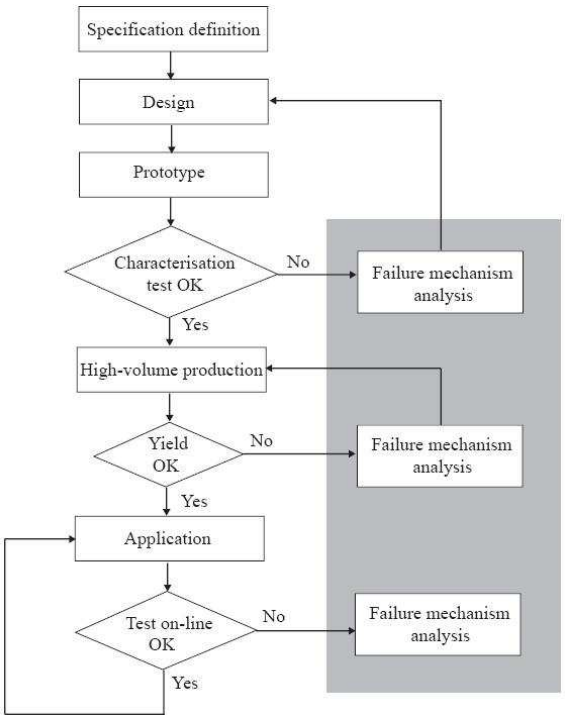


Figure 1.2: Typical design flow of an analog IC.

implement corrective actions within the time available to bring a new part to market or to bring yield and reliability to competitive levels, it becomes essential to develop a test diagnosis approach.

In cases where the IC is part of a larger system that is safety critical (e.g. automotive, aerospace), it is necessary to guaranty zero *ppm* production failures and the highest possible reliability in the lifetime. In the case of a failure in the production or a customer return, it is important to identify the root-cause of failure and apply corrective actions that will prevent failure reoccurrence and, thereby, expand the safety features.

It is necessary to understand the failure mechanisms to construct a list of realistic faults for diagnosis purpose. Nowadays fault models for digital circuits are well defined and widely used in CAD design tools and ATPG (Automatic Test Pattern Generation) to verify the fault coverage of test vectors. These models form the basis for representing the faulty circuit behaviour as well as for generating test patterns. However, fault modelling of analog circuits is still a challenge due to the continuous nature of analog circuit operation, the non-linearity, the sensitivity of performances to process variations, etc. In the absence of an acceptable fault model, analog test remains largely functional (i.e. specification test) in nature [16].

1.3 Objectives

This thesis aims at first to develop a fault modelling approach for analog ICs. To this end, it is necessary to understand all possible failure mechanisms. In general, failures in analog ICs are due to two types of faults: catastrophic and parametric faults. Catastrophic faults are often caused by spot defects in production. They can take the form

of missing or extra material and they result in a modification of the circuit topology. On the other hand, parametric faults are caused by excessive process variations, harsh environmental conditions, aging phenomena, etc. They do not change the circuit topology and they result in deviations of circuit performances. In a fault modelling approach, both catastrophic and parametric faults should be considered.

Secondly, we aim at developing a fault diagnosis approach. Catastrophic and parametric faults were treated separately in the past in the context of fault diagnosis. Diagnosis of catastrophic faults consists of identifying the location of the defect and diagnosis of parametric faults consists of predicting the parametric deviations that have resulted in performance deviation. The proposed diagnosis approach should be able to identify failures of different natures.

This thesis is carried out within the framework of the European CATRENE project CT302-TOETS. TIMA Laboratory and NXP Semiconductors cooperate in the area of fault diagnosis. The proposed diagnosis approach is validated with data of failed devices from NXP Semiconductors.

1.4 Contribution

As mentioned in the previous section, catastrophic and parametric faults are treated separately in the literature in the context of fault diagnosis. However, when an IC is found to be faulty, i.e., one or more specifications are violated, the type of fault is unknown and we cannot make any distinction regarding its type. To this end, we have developed a new diagnosis approach in this thesis based on machine learning that treats both catastrophic and parametric faults without requiring any prior knowledge, i.e., no assumption is made regarding the type of fault that has occurred in the Device Under Test (DUT).

The proposed approach has been demonstrated for validating failed devices from NXP Semiconductors. The case study is a Controller Area Network (CAN) transceiver that is used in automobiles. For this particular case study, spot defects are considered as the most frequent failure mechanism. Thus, we focus on spot defect localization for diagnosis purposes. To this end, we develop a spot defect modelling approach by considering the resistive and capacitive behaviour of the defect. Then, we use statistical methods to derive the likelihood occurrence of the modelled defects in a faulty DUT. This lets us analyze the misdiagnosed DUTs and the resulting ambiguity groups in a statistical fashion. The proposed approach can be used to guide the classical, tedious failure analysis approach and to reduce the time-to-diagnose.

For this large-scale, industrial case study, we have encountered missing values due to convergence problems in fault simulation. On the other hand, the missing value problem also concerns the real diagnostic measurement pattern due to instrument limit. To this end, we have carried out statistical analysis with missing data. Finally, the diagnosis result shows that, rather than just using pass/fail data, incorporating the actual values of measurements can greatly improve fault diagnosis.

1.5 Thesis overview

Chapter 2 introduces IC failure mechanisms and fault modelling of analog/RF circuits. In Chapter 3, the state of the art on fault diagnosis of analog/RF ICs is presented. A new fault diagnosis approach based on machine learning is presented in Chapter 4. This methodology takes into account both catastrophic and parametric faults in a unified approach. In Chapter 5, a new diagnosis approach based on non-parametric density estimation using non-idealized spot defect models is presented. The experimental results are presented in Chapter 6. The conclusion and directions for future work are given in Chapter 7.

Chapter 2

Fault modelling of analog/RF ICs

2.1 Introduction

This chapter introduces IC failure mechanisms and fault modelling approaches for analog/RF circuits. To verify the functionality of an IC and defect failures, the device is subjected to a variety of electrical tests during its lifetime. The different test steps include wafer test, final test and on-line test. Figure 2.1 shows a brief description of different test steps. During wafer test, all individual dies that are present on the wafer are tested by a wafer prober using Automated Test Equipment (ATE). The wafer prober also exercises any test circuitry on the wafer scribe lines. These special test structures in the scribe lines are designed to detect any large global deviations across the wafer without testing each individual die. Dies which fail the wafer test are often marked by different colours. The result of wafer test can be represented on a wafer map to trace manufacturing defects and mark bad dies. Figure 2.2 shows an example of a wafer map with green colours representing the good dies and other colours representing dies with different types of failures. The proportion of dies on the wafer found to perform properly is referred to as the yield

$$Yield = \frac{N}{M} \tag{2.1}$$

where N denotes the number of dies which pass the test and M denotes the total number of fabricated dies.

After the wafer test, the wafer is sliced into the dies, each of which is called a die. The good dies are then connected to the pins of the package by tiny gold wires. The

Figure 2.1: Different test steps of an IC.

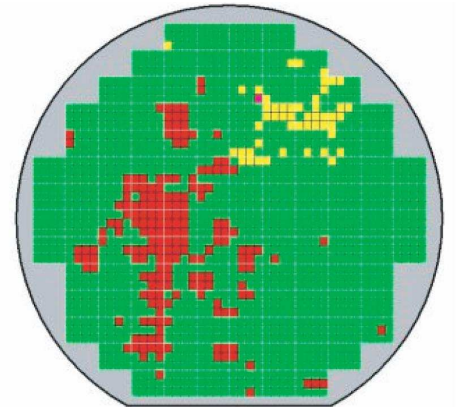


Figure 2.2: An example of wafer map.

final test consists of testing the packaged devices to ensure that they are not damaged during packaging and that the die-to-pin interconnect operation is performed correctly. Finally, on-line tests are carried out during the normal operation of the devices to verify their robustness regarding to harsh environmental conditions and ageing.

Failures can occur at any stage during the lifetime of an IC. Knowledge of the electrical failure modes and the physical mechanisms that cause failures is fundamental to implementing realistic fault models and it can give guidelines for the design of testable and reliable devices. Furthermore, the credibility of a diagnosis approach is directly related to the accuracy of fault models. Nowadays fault models for digital circuits are well defined and widely used in CAD design tools [17]. However, fault modelling of analog circuits is still a challenge due to the continuous nature of analog circuit operation, the non-linearity, the sensitivity of performances to the process variations, etc. Thus, knowing the failure mechanisms and constructing the corresponding fault models are essential for analog fault diagnosis.

2.2 Failure mechanisms in ICs

During the design stage, IC prototypes can fail due to design weaknesses or inaccurate simulation models. This type of failures can be corrected progressively during the design iterations. In a production environment, an IC is susceptible to various yield loss mechanisms. As indicated in [18], the outcome of a manufacturing operation is subjected to three major factors: the process control parameters, the layout of the IC, and some randomly changing environmental factors, called disturbances. Control parameters are manipulated in order to achieve some desired change in the fabricated IC structure. Examples of control parameters are temperature, gas pressures, step duration, etc. The layout factor is represented by lithography masks. The disturbances are environmental factors in the production. An error in any of these three factors can lead to IC failures. These factors can be further classified into global process deviations, local process variations, spot defects, and aging phenomena as shown in Figure 2.3. The rest of the section provides a detailed description of these failure mechanisms.

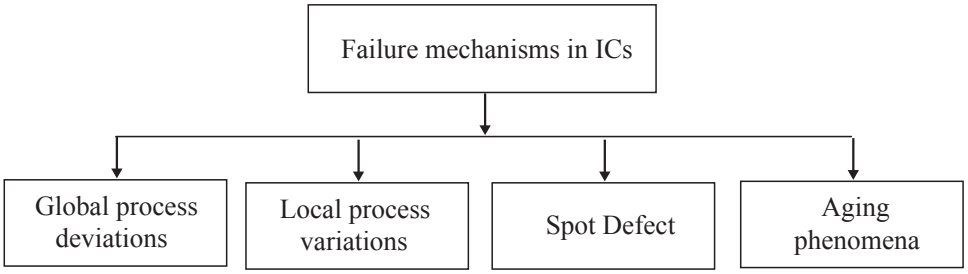


Figure 2.3: Failure mechanisms in ICs.

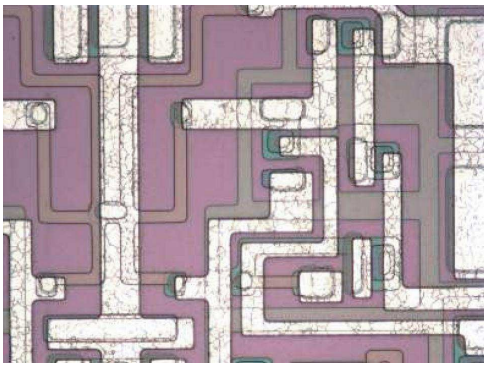


Figure 2.4: Example of mask misalignment [2].

2.2.1 Global process deviations in production

In an immature technology, ICs can fail due to a serious error in a process control parameter, the layout control or disturbances. Examples of such errors are [18]:

1. A human error or an equipment failure.

2. Instabilities in the process conditions. For example, a turbulent flow of gasses used for diffusion and oxidation can lead to global variations in the corresponding process parameters such as doping diffusion and gate oxide thickness, which in turn affect device parameters such as the threshold voltage V_{th} of MOS devices. The inaccuracies in the control of furnace temperature can also lead to global temperature variations in the production.

3. Material instabilities. These are variations of materials in the manufacturing process such as physical parameters of the chemical compounds.

4. Mask misalignment. These are errors in the position of a lithography mask which can lead to deformation of the geometry of an actual IC. This could be due to limited mechanical and optical accuracy of the processing equipment, and shape variations of the wafers. Figure 2.4 shows an example of mask misalignment.

It should be noted that under certain conditions, the aforementioned global variations can interact with each other in an indirect way. For example, high temperature processes

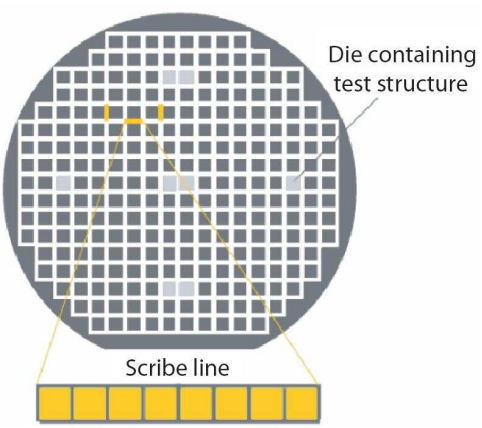


Figure 2.5: Scribe lines and dies containing test structures.

may cause an increase in the lithography errors due to the deformations in the shape of the wafer.

In the IC production, a few of the chips on the wafer or some space in the wafer scribe lines are set to contain special test structures (see Figure 2.5). These test structures are designed to have performances sensitive to the quality of specific processing steps. Examples of test structures are long contact chains, large capacitors and arrays of different transistors [18]. These structures are often referred to as Process Control Monitors (PCMs), and the measurements obtained using them are called *in-line measurements* [18]. With a PCM, technology specific parameters such as V_{th} in MOS devices, V_{be} in bipolar devices, and resistance/capacitance per unit area can be obtained. If one of the tests in PCM falls outside the predefined allowable test range, the wafer is considered defective and is discarded. Thus, any large process deviations which lead to dysfunction of the whole wafer can be readily detected by the PCMs. Therefore, large process deviations are typically not considered in the context of fault modelling and diagnosis analysis.

2.2.2 Local process variations in production

Global deviations affect all devices on a wafer in a very similar way. On the other hand, local process variations affect the components of each device on a wafer individually. In general, these variations can lead to deviation of some process related device parameters but they do not change the circuit topology. Examples of local process variations are:

1. Local geometrical deformations. These are processing effects which cause the location of the boundary of a region in an actual IC to vary. Geometrical deformations can have lateral or vertical effects as shown in [18]. Examples of lateral deformations are variations of effective channel length L_{eff} or effective channel width W_{eff} of MOS devices [19]. Figure 2.6 shows the impact of local geometrical deformations on L_{eff} and W_{eff} for a MOS device. As shown in [20], the variance of the threshold voltage $\sigma^2_{V_{th}}$ of MOS devices is inversely proportional to the term $L_{eff} \times W_{eff}$, which denotes the effective channel area.

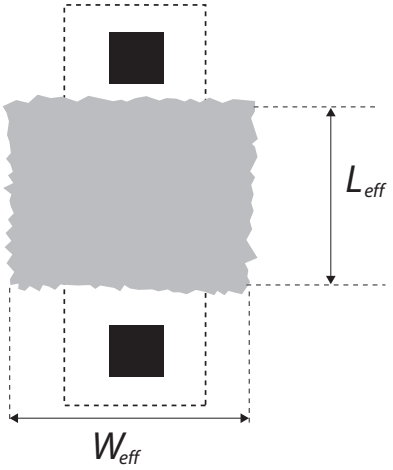


Figure 2.6: Local geometrical variations on L_{eff} and W_{eff}.

$$\sigma^2_{V_{th}} \propto \frac{1}{L_{eff} \times W_{eff}} \tag{2.2}$$

On the other hand, vertical effects are deformations in the thickness of IC layers and include deformations which are due to the p-n junction depth variations and deformations in the thickness of the oxide and other deposited layers. Junction depth variations are a direct consequence of the fluctuations in the impurity concentrations while deformations in the thickness of the deposited or oxidized layers are due to process instabilities such as turbulent gas flow, temperature fluctuations, etc.

2. Local variations in process parameters. Example of this type of variations is local doping concentration variations. Variations of doping concentration can be global as mentioned in the previous section. They can also be local due to the non-uniformity of the dopant ions density distribution or the non-uniform distribution of the threshold adjust implant atoms in the gate oxide [20]. They can result in variations of the threshold voltage V_{th} of MOS devices.

As defined before, local process variations do not change the topology of the devices. However, the mismatch in critical device pairs caused by local variations can lead to performance degradation, even device failures. As shown in [20], mismatch in MOS devices can lead to a significant yield loss for a Digital-to-Analog Converter.

2.2.3 Spot defects

Spot defects are undesired materials occurred in the IC fabrication caused by dust, particles, contamination, etc. As discussed in [16], not all defects are due to lithographic processing steps. Some defects arise from process variability such as incomplete step coverage. Therefore the way in which individual process steps are executed is of critical importance to avoid spot defects. Each of these steps has its own deviations or disturbances from the ideal process which can generate physical changes in the structure of

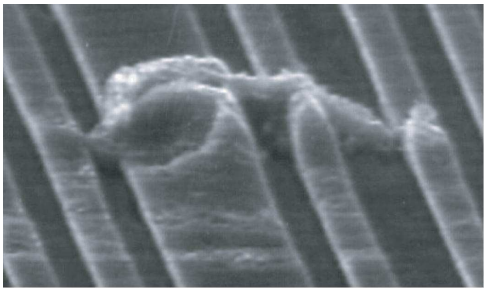


Figure 2.7: A short-circuit between conduction lines caused by a particle [3].

the IC and thus create defects. According to [10], spot defects are random phenomena occurring with certain stochastic frequency and size. This section provides a brief description of different types of spot defects met in production.

Particles, contamination in IC production environment

In the IC fabrication environment, a controlled level of contamination should be specified by the number of particles per cubic meter at a specified particle size. Nevertheless, the rare environmental pollutants such as dust can still be introduced in the IC fabrication process by the production equipment, fabrication environment, humans, etc. Particles can also be induced in the fabrication process in the form of residues such as etching residue, resin residue or different materials used during the deposition process. These particles can occur at any stage in IC fabrication and their impact on the circuit behaviour depends on the location where they are affected. The type of particles and contamination can be:

1. Contaminations on the substrate. They are referred to as "bulk failures" in [5]. They can be caused by the abnormally high leakage currents which may be observed when a crystal defect, where impurities usually precipitate, is located in the depleted region of a diffused or induced junction. These currents affect the performances of both bipolar and MOS devices. These defects can be observed with the transmission electron microscope, or by X-ray topography [5]. Contaminations can also be introduced by large crystal defects creating low-resistance paths, which shorts the collector and the emitter in bipolar circuits [21]. These defects were traditionally considered of small relevance for reliability, however, their influence on production yield and reliability is increasing with growing circuit complexity.

2. Particles in metal layers. This type of defects could be due to the ionic residues which result in short-circuits or open-circuits. Figure 2.7 shows a short-circuit between conduction lines caused by residues.

3. Residues in the fabrication process. Production process such as etching or deposition can produce contaminations and residues. This type of residues is often removed in the cleaning step of fabrication. However, the residues can remain in some cases. Figure 2.8 shows an open-circuit in the contact caused by a spot of residue between aluminium and poly interconnects.

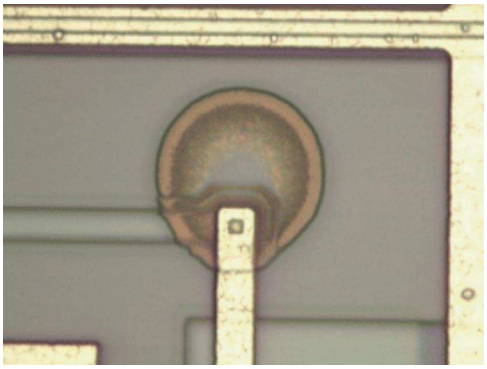


Figure 2.8: An open-circuit in a contact [3].

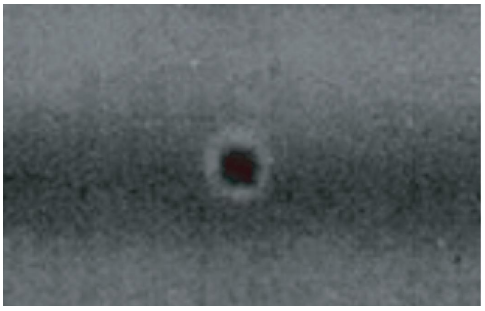


Figure 2.9: Example of a pinhole defect in the oxide [4].

4. Dusts on the mask. The dusts presented on the mask during the photolithography process can result in short-circuits and open-circuits. This type of defects is particularly dangerous since the error will be repeated on all devices of the wafer lot.

Process related defects

The process related defects can occur in any step of the IC production and they are caused by the specific fabrication process.

1. Pinhole. The pinhole defects are the small "holes" formed in dielectric insulators such as thin and thick silicon oxides, oxidized polysilicon, chemical vapour deposited insulators, etc [10].

   Pinhole defects can occur in a gate oxide when a voltage is applied. They can also occur in the insulator of the overlap region between two conductor layers that cross each other. They can create a region in the oxide which has a low electric resistance resulting in a leakage current, even a short circuit between the gate and the substrate or between two isolated metal layers [22]. Figure 2.9 shows an example of a pinhole defect in the oxide.

2. Hillock. These are the excrescences of metal in conduction layers due to non-uniform metal oxide formation on the surface of the metal structure, as well as due to high temperatures associated with the subsequent chemical vapor deposition

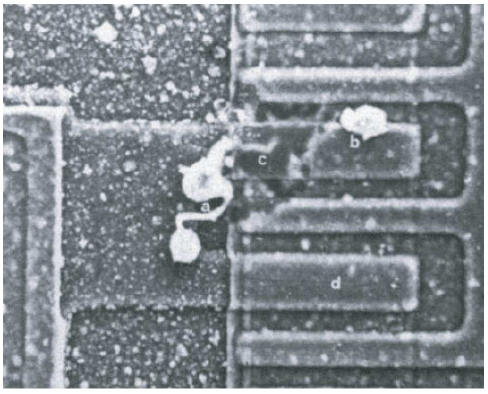


Figure 2.10: Example of a hillock [5].

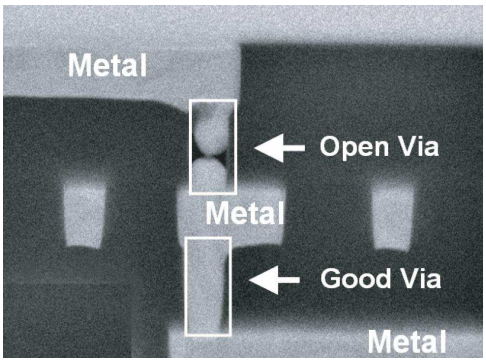


Figure 2.11: Example of an open via caused by a void [6].

(CVD). Hillock formation observed in Al and Al-Si films at elevated temperatures is caused by a build up of bilateral stress due to differences in thermal expansion between the aluminum and silicon substrate [23]. As shown in [24], hillock growth during thermal treatments of thin aluminum films used as interconnect lines can lead to problems such as dielectric cracks and line shorts either immediately or over time. Figure 2.10 shows an example of a hillock.

3. Void. They are often formed in the conduction metal layer or via between metal layers. Void can be caused by over-etching, under-etching or errors in the deposition which result in a cavity and create contaminations, short-circuit or open-circuit. Figure 2.11 shows an example of an open via caused by a void. As shown in [16], open defects caused by void can manifest themselves as broken lines or open via with a low resistance value due to the Titanium barrier layer that remains in the cavity. This type of open defects is referred to as "weak" opens and the resistance value of the open defects can follow some distribution for a specific technology. The probability of occurrence of an open via becomes higher with the increase of the complexity of modern IC devices which can have millions of vias in a structure of 6-8 layers.

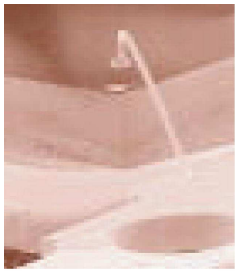


Figure 2.12: Example of a lifted ball bond [7].

Summary

Since the spot defects can lead to a modification of the circuit topology (creation of short or open circuits), they are often considered as the source of catastrophic faults. According to several reports [25, 26, 5, 27], spot defects have been recognized for a long time as the main root cause of IC failures.

2.2.4 Package-related Failure

Package-related failures occur in the assembly and packaging stages of IC production, and include ball lifting, bond shorting, contamination, die failures, etc.

Ball lifting

Ball lifting is the detachment of a ball bond from the bond pad of a semiconductor device. It can be due to a variety of factors. Poor wire bond equipment set-up and bond pad surface contamination are primary causes of ball lifting. Poor set-up includes improper wirebond parameter settings, unstable workpiece holders, and worn-out wirebonding tools. These result in poor initial welding and inadequate inter-metallic formation between the bond pad and the ball. An excessively high bonding force may tear the bonding wire and damage the pad metallisation, or even crack the oxide below the metal pad, shorting the pad to the substrate. Figure 2.12 shows an example of a lifted ball bond.

Bond Shorting

Wirebond-related shorts refer to failures that involve the occurrence of unintended electrical shorting between two wires. The point of shorting may be at any of the two wire bonding, or along the span of the wire itself. Figure 2.13 shows an example of a wire-to-wire short due to a swept wire.

Contamination

The sources of contamination can be the presence of a foreign material, whether attached or unattached, anywhere on the internal or external portions of the package

Figure 2.13: Example of bond short due to a swept wire [7].

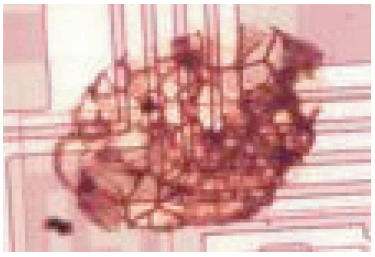


Figure 2.14: Example of an internal contaminant on the die [7].

body and/or its interconnection features (e.g. leads, solder balls, etc.). Figure 2.14 shows an example of an internal contaminant on the die.

Since certain contaminants can affect the performance and reliability of the device, they need to be identified promptly and, if necessary, traced to their root cause. Corrective actions may then be implemented to prevent recurrence.

Die failures

Die failures refer to the failure mechanisms which affect the whole die such as die corrosion, die cracking, die lifting, etc. They can be caused by fracture within the die, imperfections in the die attach materials, such as voids or some mechanical effects. Figure 2.15 shows an example of die scratches resulting in a laceration damage on the die active region.

2.2.5 Ageing phenomena

Failures can also be induced during the lifetime of an IC due to ageing, wear-and-tear, harsh environments, overuse, or due to defects that are not detected by the production

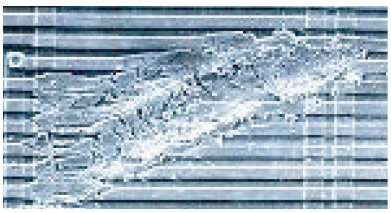


Figure 2.15: Example of die scratches [7].

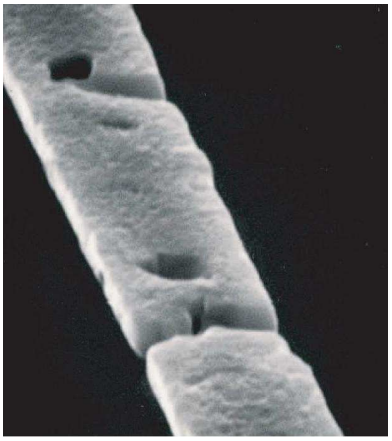


Figure 2.16: Example of an open circuit on the metal layer caused by electromigration. [8].

tests and manifest themselves later in the field of operation. This section provides a brief description of failure mechanisms due to aging phenomena.

Electromigration

Electromigration is a term applied to the transport of mass in metals when the metals are stressed at high current densities. It is due to the migration of atoms in the conduction layers caused by the electric current. As the structure size of ICs decreases, the practical significance of this effect increases. Because of the mass transport of metal atoms from one point to another during electromigration, this mechanism leads to the formation of voids at some points in the metal line and hillocks or extrusions at other points. It can therefore result in either: 1) an open circuit if the void formed in the metal line becomes big enough; or 2) a short circuit if the extrusions become long enough to serve as a bridge between the affected metal and another adjacent metal. Figure 2.16 shows an example of an open circuit on the metal layer caused by electromigration.

In [28], an empirical model to estimate the mean time to failure (MTF) of a conduction layer due to electromigration is defined as

$$\frac{1}{MTF} = AJ^2 \exp(-\frac{\varphi}{kT}) \tag{2.3}$$

where MTF denotes the mean time to failure in hours, A is a constant which contains a factor involving the cross-sectional area of the conductor, J is the current density in Amperes per square centimetre, φ is an activation energy in electron volts, k is the Boltzman's constant and T is the temperature of the conductor in degrees Kelvin. As can be observed in equation (2.3), the current density J and the temperature T are deciding factors in the design process that affect electromigration. In order to keep conductors reliable with rising temperatures, the maximum tolerable current density must necessarily decrease.

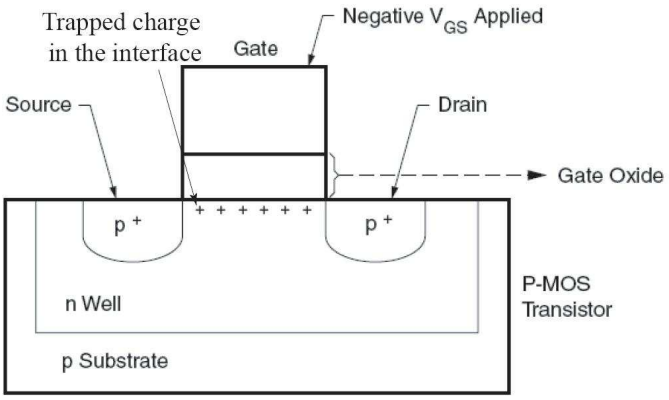


Figure 2.17: NBTI in PMOS Transistor.

Negative Bias Temperature Instability (NBTI)

The Negative Bias Temperature Instability (NBTI) occurs in PMOS devices stressed with negative gate voltages at elevated temperatures. The semiconductor process evolution that produces small transistors increases the potential for interface traps in PMOS transistors during prolonged times of negative bias stress (see Figure 2.17). An interface trap is located near the Si-oxide/Si-crystal lattice boundary where holes (positive charge) can get stuck resulting in a shift of the threshold voltage V_{th}. This hole trapping creates interface states as well as fixed charges. Both are positive charges and result in a negative shift of V_{th}. NMOS transistors are far less affected because interface states and fixed charges are of opposite polarity and eventually cancel each other.

As shown in [29], the degradation of V_{th} exhibits logarithmic dependence on time. This degradation can be caused by: voltage stress on the gate oxide, temperature, and the duty cycle of the stressing voltage. This effect becomes more severe as:

- Transistor dimensions continue to shrink.
- The electric field applied to the gate oxide increases.
- The operating voltage becomes lower which makes a given threshold degradation cause a relatively larger impact on the circuit behavior.

In the design stage, the bias conditions of each PMOS transistor must be considered not only at the beginning but throughout the expected lifetime of the product in order to improve reliability.

Hot carriers injection (HCI)

The HCI occurs when either an electron or a "hole" gains sufficient kinetic energy to overcome a potential barrier necessary to break an interface state. It usually refers to the effect in MOS devices, where a carrier is injected from the conducting channel in the silicon substrate into the gate dielectric. Injected carriers that do not get trapped in the gate oxide become gate current. On the other hand, the majority of the holes from the

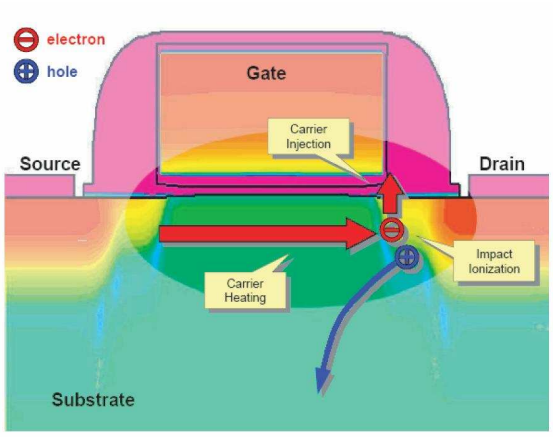


Figure 2.18: Hot carriers injection phenomena.

electron-hole pairs generated by impact ionization flow back to the substrate, comprising a large portion of the substrate drift current. Excessive substrate current may therefore be an indication of hot carrier degradation. Figure 2.18 shows the principle of the HCI phenomena.

Over prolonged periods, the presence of such mobile carriers in the oxides can lead to deviations of device parameters such as the threshold voltage V_{th}. The useful lifetime of CMOS integrated circuits is thus affected by the lifetime of the MOS devices themselves. As shown in [30], the degradation in V_{th} can be expressed as:

$$\Delta V_{th} = C(\exp(L_0/L_{eff}))(\exp(-V_0/V_d))(t/t_0)^n \tag{2.4}$$

where C is a constant in mV, L_0/V_0 is a characteristic length/voltage depending on the device, L_{eff} is the effective length, V_d is the drain voltage, t denotes the stress time and t_0 is a constant.

To ensure that integrated circuits manufactured with minimum geometry devices will not fail rather rapidly, the MOS devices must have their HCI degradation well understood and characterized. Failure to accurately characterize HCI lifetime effects can ultimately affect business costs such as warranty and support costs, as well as impact marketing and sales promises for a foundry or IC manufacturer.

Oxide breakdown

Oxide breakdowns can be classified as Electrical Over Stress (EOS)/Electro Static Discharge (ESD) induced dielectric breakdown and time-dependent dielectric breakdown (TDDB).

1. EOS/ESD-induced dielectric breakdown. The EOS/ESD-induced dielectric breakdown involves a high voltage being applied across the oxide layer causing a “weak” spot within it to exhibit dielectric breakdown and allow current to flow. This current flow, which is basically due to loss of dielectric isolation at that spot, causes localized heating, which induces the flow of a larger current. A vicious cycle of increasing current flow and localized heating, eventually causes a meltdown of the

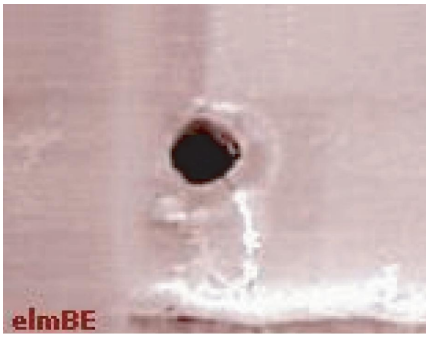


Figure 2.19: An ESD-induced oxide breakdown [9].

silicon, dielectric, and other materials at the "hot spot". This meltdown creates a short circuit between the layers supposedly isolated by the oxide. Figure 2.19 shows an ESD-induced oxide breakdown.

2. Time-dependent dielectric breakdown (TDDB)

   The TDDB is a failure mechanism in MOS devices, when the gate oxide breaks down as a result of long-time application of relatively low electric field. The breakdown is caused by formation of a conducting path through the gate oxide to substrate due to electron tunneling current, when MOS devices are operated close to or beyond their specified operating voltages. As shown in [31], the mean time to failure due to TDDB can be expressed as:

$$t = A \exp(-\gamma E) \exp(\frac{E_\alpha}{kT}) \tag{2.5}$$

   where t is the mean-time to breakdown, A is a constant, γ is the field acceleration parameter, E is the oxide electric field, E_α is the thermal activation energy, k is the Boltzmann's constant and T is the absolute temperature. As can be observed in (2.5), as the oxide electric field and operation temperature increases, the mean-time to breakdown reduces.

2.3 Fault modelling

This section discusses fault modelling approaches for the various failure mechanisms presented in the previous section. As mentioned earlier, a catastrophic fault (short or open circuit) results in a change of circuit topology while a parametric fault does not alter the circuit topology. The first, the second and the third column of Table 2.1 show a brief summary of the failure mechanisms and the corresponding fault models. The fourth column shows when the failure occurs (in production or in the field of operation).

Several fault models are proposed in the literature to model the failure mechanisms shown in Table 2.1. In [32], fault models are classified as structural models, parametric models and behavioural (functional) models. These models are presented in this section.

Table 2.1: Summary of the failure mechanisms and the corresponding fault models.

Location	Failure mechanism	Fault model	When
Whole wafer	Global process deviations	-	P[1]
Individual device	Local geometrical deformations	Parametric fault	P[1]
Individual device	Local process variations	Parametric fault	P[1]
Substrate	Substrate contaminations	Short circuits	P[1]
Random phenomena	Particles	Short & Open circuits	P[1]
Random phenomena	Residues	Short & Open circuits	P[1]
Mask	Dusts	Short & Open circuits	P[1]
Oxide	Pinhole	Short circuits	P[1]
Oxide	Hillock	Short circuits	P[1]
Metal layer or via	Void	Open circuits	P[1]
Package level	Ball lifting	Open circuits	P[1]
Package level	Bond Shorting	Short circuits	P[1]
Package level	Contamination	Short & Open circuits	P[1]
Package level	Die failures	-	P[1]
Metal layer or via	Electromigration	Short & Open circuits	F[2]
MOS devices	NTBI	Parametric fault	F[2]
MOS devices	HCI	Parametric fault	F[2]
Oxide	EOS/ESD breakdown	Short circuits	F[2]
Oxide	TDDB	Short circuits	F[2]

2.3.1 Structural fault model

The structural fault model is used to model the failures which lead to a modification of circuit topology in the case of digital circuits. These effects are represented by stuck-at faults, high-impedance states or bridge faults (e.g. a short between two signal paths). Structural fault models for analog circuits are in essence short and open circuits.

The structural model can be simulated and implemented either of the layout level or of the netlist level. At the layout level, fault modelling consists of injecting missing or extra material on the conduction layers or on the contacts between layers. It should be noted that an injected defect does not systematically lead to a fault [32]. For example, extra metal on the conduction layer does not necessarily affect the functionality of the circuit. At the netlist level, fault modelling consists of representing a physical defect by modifying the circuit topology. Typically, faults are modelled by resistive short and open circuits at the netlist level. A short circuit is typically modelled by a small value

[1]Production
[2]Field

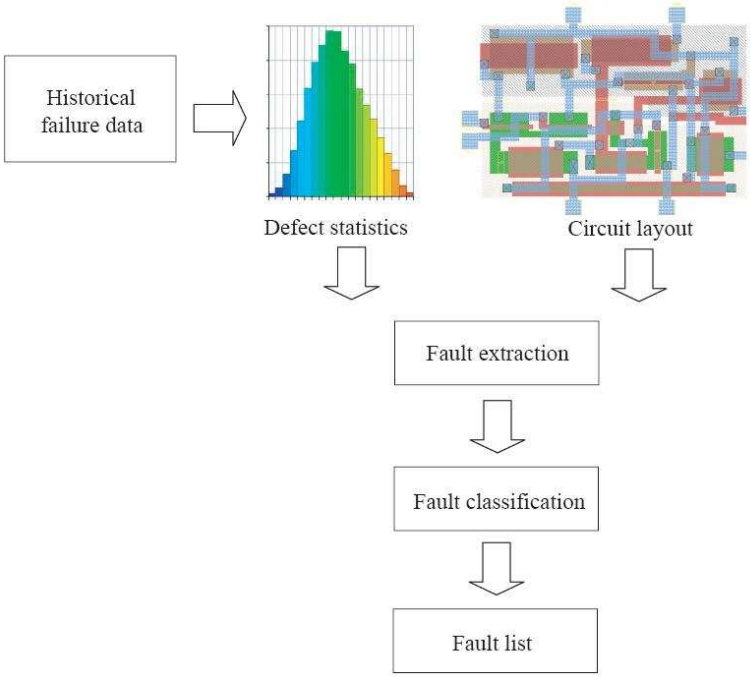


Figure 2.20: An overview of IFA analysis.

resistance (from 1 to 10 Ω) and an open circuit is typically modelled by a large value resistance (from 10M to several G Ω).

Inductive fault analysis (IFA)

Traditionally, structural fault models are developed by considering a probable list of faults that can occur in a given circuit. A short circuit is assumed to occur between two nodes of a component and an open circuit is assumed to occur on the wires. The advantage of this method is its simplicity. The fault list can be obtained by analysing the topology of the circuit. However, the derived fault list does not represent the geometrical reality of defects. For instance, certain faults such as a short circuit between two nodes of an inductor are very unlikely to occur given its distance at the layout.

To solve the problem encountered in traditional structural fault modelling, the Inductive Fault Analysis (IFA) has been proposed in [33]. It is a systematic method for determining what faults are likely to occur in a circuit. It takes into account the circuit fabrication technology, fabrication defect statistics, and physical layout. Figure 2.20 shows a high-level description of IFA analysis. As mentioned in [33], the IFA analysis contains two principal steps:

The first step of the IFA analysis involves statistical defect generation. The information of defect statistics can be obtained from actual experiment data. They consist of two attributes, namely the density of defects per unit area and the probability density function of the defect sizes. The shape of the defects can be assumed to be round or square. Figure 2.21 shows an example of generation of a random defect. The density of defects per unit area can be expressed as a function of the geometrical position of the defect (x, y). The probability density function of the defect size can be expressed as a

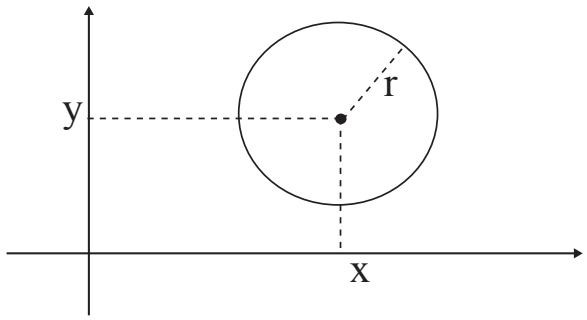


Figure 2.21: An example of generation of a random defect.

function of the radius of the defect $p(r)$. Once these data are obtained, defects can be injected at the layout level using their statistical distribution.

The second step of the IFA analysis involves fault extraction, classification and ranking. The injected defects at step 1 are extracted at the circuit level in this step using a fault extractor. A fault extractor makes use of both the layout description and the fault-free circuit diagram. It extracts the faulty circuit diagram from the modified layout, which incorporates the defect. By carefully examining the extracted faults and interpreting their effects at the circuit level, a classification of circuit faults can be produced. Different types of faults are classified in [33] such as line stuck-at faults, transistor stuck-at faults, floating line faults, and bridging faults. After the fault classification, the faults are then grouped and ranked according to their probability of occurrence. The number of defects which cause a particular circuit fault is indicative of the likelihood of that fault.

In the IFA scenario, the single defect assumption is used. One single defect of a time is generated, analyzed, and translated. Faults caused by simultaneous multiple defects are not likely and therefore are not considered. However, a single defect can impact multiple layers. Hence, the IFA procedure can include both single and multiple faults.

The IFA analysis allows to obtain a list of defects according to their geometric characteristics, which is more realistic than the traditional method by assuming a list of defects. More accurate test metrics can be estimated by injecting defects using the IFA analysis. In [34], a comparative study has shown that the fault coverage can be different for the same test measurements using the traditional fault list construction method and the IFA analysis. This demonstrates the importance to have a realistic fault list.

Defect size and density estimation

As mentioned previously, spot defects are random phenomena occurring with certain stochastic frequency and size. Deriving correctly the density of defects per unit area and the probability density function of the defect sizes is very important to generate defects in IFA analysis. In [10], the average number of faults λ caused by defects is expressed as

$$\lambda = A(x)D \tag{2.6}$$

where D denotes the density of defects per unit area, and $A(x)$ denotes the critical area

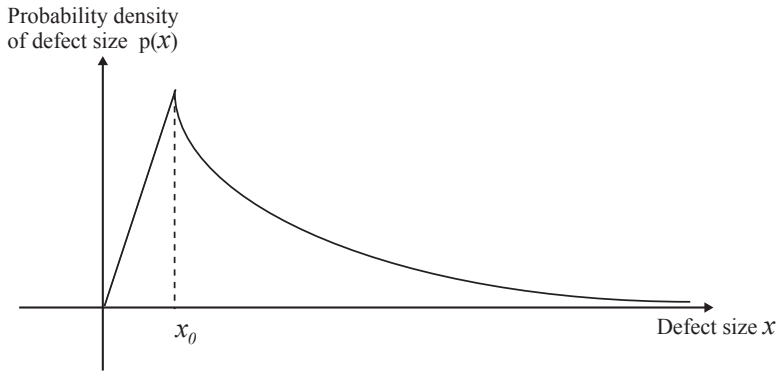


Figure 2.22: Probability density function of defect size [10].

with respect to the defect size x. If the defect is modelled by a circle, x will be the diameter of the circle. The critical area $A(x)$ can be expressed as

$$A(x) = f(\theta(x), p(x), A) \tag{2.7}$$

where $\theta(x)$ denotes the fraction of the total chip area which is sensitive to the defects, A denotes the total chip area, and $p(x)$ denotes the probability density function of defect size x. As can be observed in (2.6), the number of faults caused by defects depends on D, $p(x)$, as well as the circuit topology. In [10], $p(x)$ is estimated based on historical failure data

$$p(x) = \begin{cases} \frac{2(n-1)x}{(n+1)x_0^2} & \text{for} \quad 0 \leq x \leq x_0 \\ \\ \frac{2(n-1)x_0^{n-1}}{(n+1)x^n} & \text{for} \quad x_0 \leq x \leq \infty \end{cases} \tag{2.8}$$

where x denotes the defect size, x_0 and n can be obtained from experimental data in a specified technology. Values of n=2 or n=3 have been obtained in different experiments in [35]. Figure 2.22 shows the probability density function of defect size estimated in [10]. Similar defect size estimation results can be found in [36, 37, 38].

Defect resistance measurements

In the IFA analysis, the geometric characteristics of defects are taken into account. However, defect resistance values are considered in a rather arbitrary manner, i.e., short circuits are injected by considering extra material and open circuits are injected by removing a portion of the material.

Spot defects modelled as a complete open or short circuit in the metal lines are referred to as "hard" since they lead to a complete malfunction of the circuit. However, not all spot defects can be classified as hard defects. In [39], a tunnelling current across the open circuits caused by electromigration was observed, which led to a finite resistance value between the two ends of the open circuits. In [14], the values of open resistances for different metal layers and contacts are estimated. In [13], the measurement of the resistance of short (e.g. bridging) defects is shown. In [40], the behaviour of defects is modelled with S-parameters that are obtained through low-level physical simulations.

Experiences in [14, 13] show that the values of defect resistances for short and open circuits can follow some distribution according to the technology under consideration. A general trend can be observed from these experiments. In particular, we can state that the resistance value of open defects can vary from less than 100 kΩ to several GΩ, whereas the resistance value of short defects can reach 20 k Ω. Open defects with finite resistance and short defects with non-negligible resistance are referred to as "soft" defects since they do not necessarily lead to complete malfunction of the circuit. As a result, the effect of some soft defects could be similar to the effect of parametric faults.

2.3.2 Parametric fault model

As presented in previous sections, failures which do not change the circuit topology are referred to as parametric faults. They can include local geometrical deformation, local process variations, and failures due to aging phenomena such as NBTI and HCI.

In [41], a parametric fault model is proposed by searching the minimum deviation of a parameter which violates at least one specification of the circuit. In order to obtain this deviation, the considered parameter is swept by keeping other parameters at their nominal values until at least one specification is violated. This method is used to evaluate the test metrics in [42, 43, 44]. However, there have been concerns regarding the realistic deviation of a component value, and whether there is sufficient process data to show that these significant parametric deviations actually occur in well-controlled production processes.

In [16], the variation of the threshold voltage V_{th} of MOS devices is modelled by

$$(\sigma V_{th})^2 = (\sigma_G)^2 + 0.5(\sigma_\Delta)^2 \tag{2.9}$$

where σ_G denotes the standard deviation of global inter-die variations of V_{th}, and σ_Δ denotes the local variations (also named mismatch) of V_{th}. As shown in (2.2), σ_Δ is inversely proportional to the square root of the effective area of the transistor. This local effect is random and is due among other things to the statistical distribution of dopant atoms per area. Figure 2.23 shows the modelling of both global and local threshold voltage variations with μ denoting its nominal value.

In [45], a general degradation model taking into account both HCI and NBTI effects is defined as

$$D = D_0 + A_x T_{str}^{\eta_x} \tag{2.10}$$

$$A_x = f(V_{DS}, V_{GS}, V_{th0}, T, W, L, \dots) \tag{2.11}$$

where subscript x represents the degradation mechanism (HCI or NBTI), D represents the degrading transistor parameter (e.g. the threshold voltage V_{th}) and D_0 its initial value. η_x is a time exponent ($\eta_{HCI} \approx 0.5$ and $\eta_{NBTI} \approx 0.16$), A_x is a function of design parameters (e.g. L,W), environmental parameters (e.g. temperature T), and process-related parameters. For HCI degradation,

$$A_{HCI} = C_{HC} \frac{1}{\sqrt{L}} \exp(\alpha_1 E_{ox}) \exp(\frac{E_{\alpha,1}}{kT}) \exp(\alpha_2 V_{DS}) \tag{2.12}$$

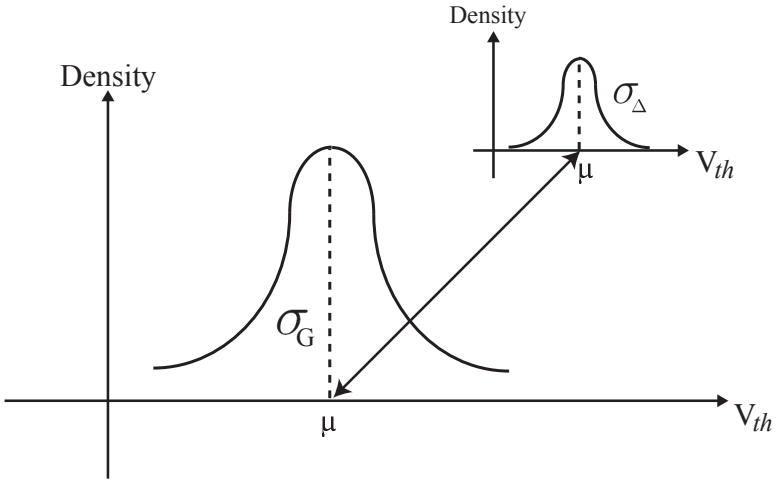


Figure 2.23: Modelling of global and local variations. The resulting V_{th} is obtained after the addition of both contributions.

where C_{HC}, α_1 and α_2 are technology-dependent parameters, L is the transistor length, E_{ox} is the oxide field strength, $E_{\alpha,1}$ is the temperature activation energy, T is the temperature, and V_{DC} is the drain-source voltage. For NBTI degradation,

$$A_{NBTI} = C_{NBTI} \exp(\alpha_3 E_{ox}) \exp(\frac{E_{\alpha,2}}{kT}) \tag{2.13}$$

where C_{NBTI} and α_3 are technology-dependent parameters and $E_{\alpha,2}$ is the temperature activation energy. To model the dynamic effect of time-varying stress voltage on the degradation, an integral equation is used

$$D(t) = \left[\int_0^t (A_x(t))^{1/\eta_x} dt\right]^{1/\eta_x} \tag{2.14}$$

Parametric fault modelling requires a deep understanding of the process variations leading to the parametric deviations. Often process variations are technology related. In the absence of knowledge of parametric deviation mechanisms, an arbitrary large distribution is often assigned to component parameters to model parametric faults.

2.3.3 Behavioural fault model

Behavioural fault models seek to reduce the complexity of fault modelling by considering a high level description of the circuit. Faults are then injected by varying the high-level parameters which of the circuit. In some cases, the high-level parameters are the actual specifications of the circuit [40, 46]. Fault diagnosis at the behavioural is possible when the fault to be detected propagates to one of the high-level parameters.

Since faults are modelled at a behavioural level, fault simulation can be much faster. In [46], a behavioural fault model is proposed by computing the ratio between the input current and the output current of a current mirror. In [40], the physical-level defects of a Low Noise Amplifier (LNA) are simulated and the S-parameters of each considered defect are extracted to construct the behavioural fault model. Then the circuit is simulated

with these fault models in order to evaluate the test strategy. In [47], a hierarchical fault model which contains several abstraction levels has been proposed. At the highest level, the performances of the circuit are considered. At the intermediate level, faults are modelled by varying the performances of a sub-circuit (e.g. the gain of an amplifier) or the values of the passive components. At the lowest level, the threshold value of the MOS devices V_{th} or the ratio W/L is considered.

Behavioural fault models are very useful for a complex system which needs a hierarchical analysis. They are easy to derive from specification or high-level parameters. However, they do not provide insight into the physical fault and it is generally difficult to analyze the root cause of failures using them.

2.4 Conclusion

We have presented in this chapter a state of the art of IC failure mechanisms and fault modelling approaches. Understanding the failure mechanisms throughout the lifetime of an IC is necessary in order to construct a list of realistic faults. Then, different fault modelling approaches have been shown. Since the efficiency of a diagnosis approach is directly related to the fault models, developing realistic fault models is also of paramount importance for the purpose of diagnosis. The next chapter will present the state of the art in fault diagnosis techniques for analog/RF circuits.

Chapter 3

State of the art on analog/RF fault diagnosis

3.1 Introduction

Fault diagnosis consists of finding the root cause of the dysfunction of a circuit. Accurate diagnostic methods are useful to (a) reduce design iterations in IC prototypes, (b) analyze the failure mechanisms from high-volume production data so as to enhance yield for future IC generations, and (c) identify the root-cause of failure in cases where the IC is part of a larger safety-critical system (e.g. automotive, aerospace) so as to improve safety features. Fault diagnosis has become a severe challenge nowadays that calls for immediate solutions. According to anecdotal evidence [48], 35% of car failures are due to the embedded electronics, of which only 60% are diagnosed, the rest being classified as "trouble not found". Amongst the factors that inhibit diagnosis are the limited controllability and observability of internal blocks of ICs, the difficulty to de-embed internal components of blocks (i.e. reverse engineering), the difficulty to deal with unanticipated faults, the limited diagnostic information (only one/few IC samples showing the same erroneous behavior are available), and the fault ambiguity (i.e. different faults having the same influence on the IC behavior) which does not permit case-based reasoning. This section provides the state of the art on manufacturing test and fault diagnosis of analog/RF ICs.

3.2 Manufacturing test approaches

Traditionally, failure analysis (FA) methods consist of optical inspection of defects to identify the root cause of failure. As indicated in [15], developing a test enabled diagnosis approach is very important since the time and the cost required for applying traditional FA methods have become intolerable with the increasing reduction in feature sizes and the high complexity of modern ICs. Different test approaches exist to verify the functionality of analog ICs. They can be broadly categorized as specification test, alternate test and defect oriented test. In specification test, the performances of the DUT are measured and compared to the pre-defined acceptable limits. The alternate test consists of mapping some low cost tests to the specification tests in order to reduce

the test cost. Finally, defect oriented test is developed to detect the presence of a defect within the DUT. Despite of the time and the cost that the specification tests may take and the continuous efforts to replace them by less expensive tests, specification tests remain today the only acceptable test approach for most industrial analog/RF devices in the absence of an acceptable analog fault model, as discussed in [16].

3.2.1 Standard specification testing

The specification test consists of verifying one by one all the performances of the DUT such as gain, slew rate, CMRR (Common-Mode Rejection Ratio) or PSRR (Power Supply Rejection Ratio). A specification is defined by a lower/upper limit. If a performance is out of the pre-defined lower/upper limit, the DUT is declared as faulty.

Ordering and optimization of specification tests

Typical industrial practice in production testing involves performing all specification tests, where if any test is failed, then the die is assigned a failure bin number and testing is terminated. As discussed in [49], the average production testing time varies depending on the order of the tests since testing is terminated as soon as a test is failed. Thus, if tests which are failed must frequently be performed fist, then the average production testing time will be shorter compared to when these are performed last. Suppose a test set has n tests which are ordered from the first position (O_1) to the last position (O_n), requiring test times T_{O_i}, $i = 1, \ldots, n$. The probability P_{O_i} that the i^{th} test is performed is [49]

$$P_{O_i} = \prod_{j=1}^{i-1} Y_{O_j} \tag{3.1}$$

where Y_{O_j} is the yield of the test in the O_j position, given previous tests in positions O_1 to O_{i-1}. Average test time is then defined as

$$AverageTestTime = \sum_{i=1}^{n} T_{O_j} P_{O_j} \tag{3.2}$$

Hence, the pass/fail data for each of the circuit specifications are needed for a number of fabricated chips in order to calculate Y_{O_j} and minimize production testing time. In [50], the Dijkstra's algorithm has been used to optimize the order of specification tests. The Dijkstra's algorithm is a graph search algorithm that solves the single-source shortest path problem for a graph with non-negative edge path costs, producing a shortest path tree. Specifically, the test selection problem is formulated as a shortest path problem in a directed graph, where the computational complexity is dominated by the number of possible subsets of the test set, 2^n. In [51], a fault-driven approach is followed. A set of non-redundant functional tests is built cumulatively by adding at each step the test for which the yield of the currently excluded tests is maximized. The algorithm terminates when a desired fault coverage is reached. In [52], regression models are built to map a set of applied tests to the values of the rest of tests (that will not be applied). Test limits are assigned such that they guarantee the compliance of the unperformed

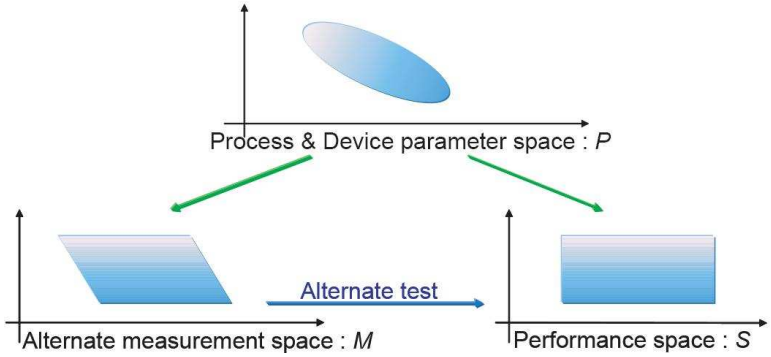


Figure 3.1: An overview of alternate test.

tests to the specifications, within estimated confidence levels. In [53], a multi-objective genetic algorithm has been proposed to search in the power-set of specification tests in order to select appropriate subsets. In [54], a decision tree approach has been proposed to compact the complete test set by eliminating redundant tests. All these approaches require data on defective devices. However, in [55], an analog test ordering approach has been proposed based on a statistical estimation of parametric defect level. A statistical model of n specification tests is obtained by applying a density estimation technique to a small sample of functional devices (obtained from the initial phase of production testing or through Monte-Carlo simulation of the design). The statistical model is next sampled to generate a large population of synthetic devices which will include defective devices. Specification tests can be then ordered according to their impact on defect level by means of feature selection techniques.

3.2.2 Alternate testing

Specification testing of analog circuits is today the only acceptable test approach by the industry. However, it suffers from the drawback of requiring length test times and expensive tester resources required to carry out all specification tests. To address this issue, the alternate test approach has been proposed to replace the specification tests by low-cost tests using non linear regression [56]. Figure 3.1 shows an overview of alternate test.

As show in Figure 3.1, the variation in the DUT performances in space S is not random but a systematic phenomenon caused by variations in the manufacturing process parameters in space P. Similarly, a test stimulus can be selected in such a way that the DUT response to the test stimulus, i.e., the alternate measurement in space M is also governed by the underlying process parameters. Therefore, a statistical tool can be used to capture the relations between the alternate measurements and the specifications, based on measurements made on a large sample of devices. This provides a mechanism for estimating the DUT performances from a set of alternate measurements, without explicitly testing for its specifications.

As shown in [56], the alternate test consists of two phases, namely training and testing phase. In the training phase, the mapping from the alternate measurements

to the specifications is built based on a large number of IC samples using nonlinear regression functions. In the testing phase, alternate measurements are taken for new DUTs. Then, the regression functions built in the first phase are used to predict the specifications without explicitly performing them in the DUTs.

In [57], a variety of built-in sensors has been proposed to extract low cost measurements that are mapped to the performances of the DUT, including non-intrusive sensors such as dummy circuits, process control monitors (PCMs) and sensors electrically connected to the DUT, such as DC probes, envelope detectors and current sensors.

The alternate test approach provides a low-cost solution for verifying the functionality of analog circuits. However, this approach needs to perform signature calibration and outlier detection. When a catastrophic fault such as a spot defect occurs in the DUT, the topology of the DUT has been changed and the mapping between the alternate measurements and the performances in figure 3.1 becomes no longer valid. Thus, outliers should be excluded from the training phase since they are inconsistent with the statistical nature of the bulk of the training data and will adversely affect the fit results. In [58], a defect filter has been proposed based on an estimate of the joint probability density function of the alternate measurements. The construction of the filter does not require a defect dictionary and can accommodate any underlying density without needing any prior knowledge regarding its parametric form.

3.2.3 Defect-oriented testing

The defect-oriented testing aims to detect the presence of a defect within the circuit. Figure 3.2 shows an overview of this approach. The failure mechanisms are evaluated in order to determine a compact test set that can assure product quality while maintaining the test cost low [16, 59]. To accurately represent the failure mechanisms of analog devices, realistic fault models are essential. Then, fault simulation is carried out by injecting each modelled fault at the netlist or layout one at a time. A fault is considered detected if the faulty response differs from the nominal response by a pre-determined margin.

The goal of defect-oriented test is to distinguish the defective and nondefective devices. In [60], a machine-learning-based test method is proposed to allocate a non linear boundary between defective/nondefective devices using a neural system. A confidence level is introduced in order to re-test the devices having an insufficient confidence. Furthermore, by varying the desired level of confidence, it enables the exploration of the tradeoff between test cost and test accuracy.

Test stimuli generation, measurement selection and extraction

Defect-oriented testing requires the selection of measurements sensitive to the failures that are likely to occur. The generation of test stimuli and the extraction of diagnostic measurements are circuit-specific problems. Many proposals can be found in the literature for test stimuli generation in a defect-oriented test approach. In [61], white noise has been used as test stimuli for diagnosing analog filters. Using white noise as a test signal allows diagnosis to be performed through the primary inputs and outputs of the DUT.

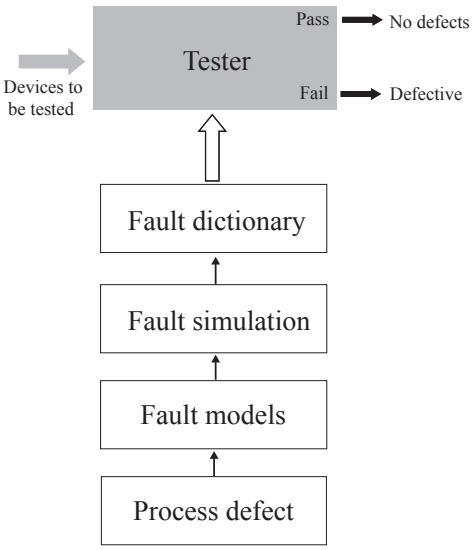


Figure 3.2: Defect-oriented test approach.

In [62], the application of a ramp signal at the power supply of an analog amplifier is investigated. The bias currents existing in the supply bus are used as a fault signatures. They are a function of the operating point as well as the topology of the circuit. For a constant supply voltage, this relationship can be represented as

$$i_{bias} = f(O, T) \tag{3.3}$$

where O is the operating condition of the circuit and T is the topology of the circuit. When O is perturbed by a external source such as a ramp signal on the power supply V_{DD}, the operating condition changes as a function of gate-source V_{gs}, drain-source V_{ds} and threshold voltages V_{th} of each transistor.

$$O = f(V_{gs}, V_{ds}, V_{th}) \tag{3.4}$$

Since the ramp signal forces all transistors to operate across all operation regions, the supply current will differ in time. The time dependency of the supply current can be defined as

$$I_{dd}(t) = f(O(t), T) \tag{3.5}$$

Since most common defects such as short and open circuits change the operating condition of the circuit O and the topology of the circuit T, they can be easily detected by sampling $I_{dd}(t)$. Using the supply current as a signature is an effective method to detect and isolate defects in analog circuits. However, as operation frequency increases, defects which affect the high frequency operation such as an open circuit on a capacitor in an RF circuit may not be detected using supply current signature. In [63], the wavelet decomposition has been used to decompose the response of a DUT and then the Principal Component Analysis (PCA) has been carried out to reduce the dimensionality of the DUT response. In [64], a sensitivity analysis has been proposed to select a set of test frequencies for fault diagnosis. Authors in [64] have also proposed a blind selection to choose test frequencies.

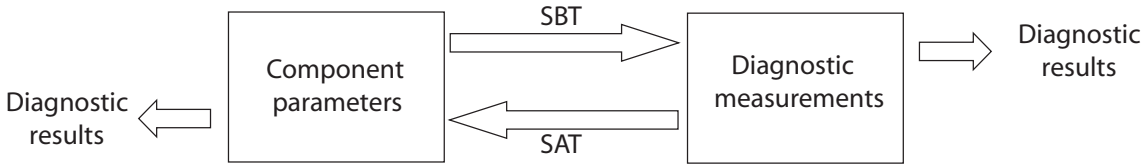


Figure 3.3: A brief description of the SBT and the SAT approach.

As discussed earlier, the main bottleneck of the defect-oriented test approach is the accuracy of the fault model. Indeed, analog fault modelling is an on-going research topic and, certainly, success with this respect will also greatly benefit defect-oriented testing and diagnosis.

3.3 Previous work on fault diagnosis

As shown in [65], analog fault diagnosis methods can be categorized in two principal strategies: simulation before test (SBT) and simulation after test (SAT). In SBT strategy, for a particular DUT, a fault list is obtained at first. Then the corresponding responses of the DUT for all considered faults are recorded. This can be done by examining the DUT or by carrying out fault simulations using a SPICE-like simulator. Faults are then consequently diagnosed by comparing simulated and observed responses. On the other hand, the SAT strategy has been designed to solve for values of component parameters, given a set of measured responses and knowledge of the topology of the DUT. As discussed in [49], the term "simulation" used to describe these two strategies is basically some algorithms which solve for some response parameters given the topology of the DUT and some "input" parameters. For SBT approach, the "input" parameters consist of component parameters of the DUT, i.e., the design parameters, and the response parameters consist of the diagnosis measurements. The SAT approach is used to determine the inverse map, i.e., the "input" parameters consist of the diagnostic measurements, and algorithms are developed to solve the response parameters, i.e., component parameters of the DUT. Figure 3.3 shows a brief description of SBT and SAT approaches and Figure 3.4 shows different diagnosis methods. A detailed description will be shown in the next section.

3.3.1 Simulation before test (SBT)

In this approach, fault simulation is carried out before the test using a list of predefined faults. It is mostly suitable for catastrophic faults or parametric faults with fixed values since an infinity of possible values exist for a parametric deviation. The SBT approach can be further classified into two major approaches, namely rule-based approach and fault dictionary approach. This section gives a detailed description of these approaches.

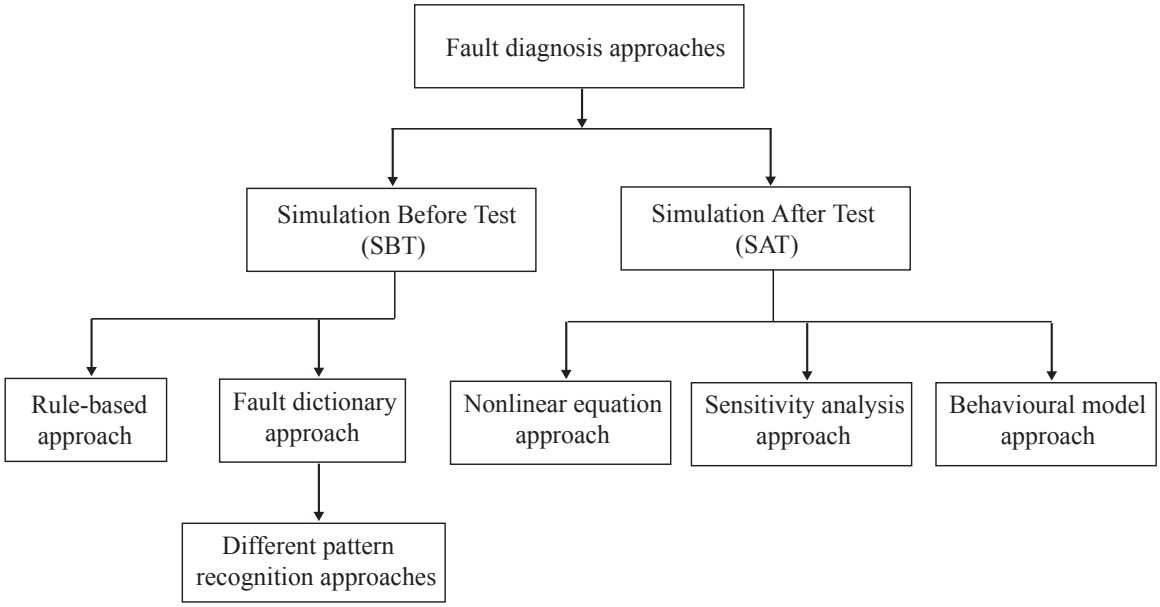


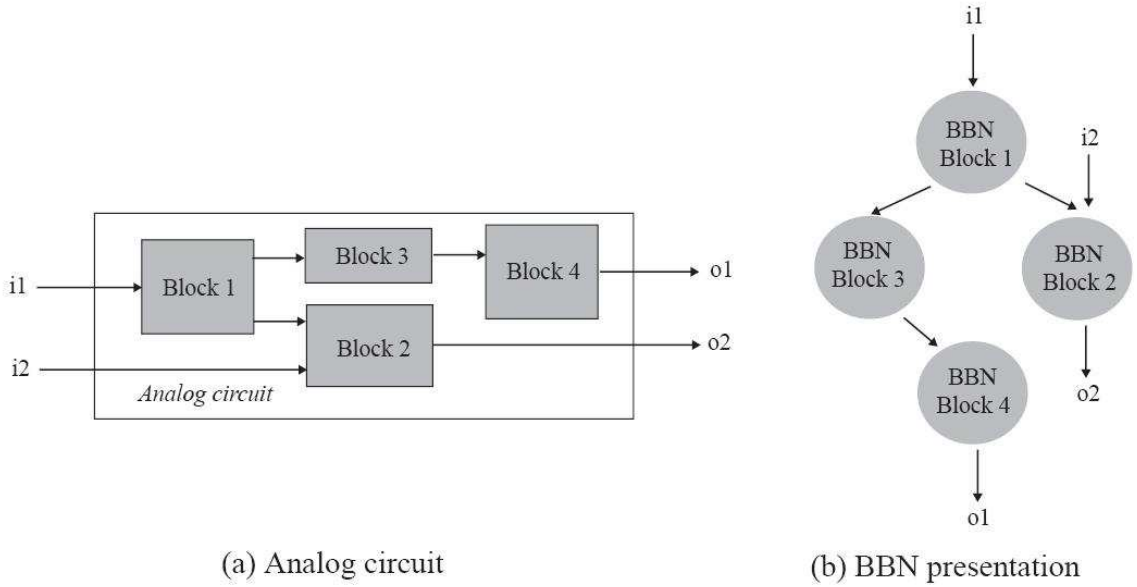
Figure 3.4: A brief classification of different fault diagnosis approaches.

Rule-based approach

Rule-based diagnosis represents the experience of skilled diagnosticians in the form of rules which generally take the form "IF symptom(s) THEN fault(s)". For a particular problem domain, representing the knowledge may require hundreds, or even thousands of rules [66]. The fault (decision) tree approach presented in [66] can also be categorized as rule-based approach since the relationship between the "symptoms" and faults is represented in terms of conditions and rules as well.

In [11], a probabilistic approach based on the Bayesian Belief Network (BBN) is proposed in order to improve the basic rule-based approach. A BBN is a probabilistic graphical model that represents a set of random variables and their conditional dependencies via a directed acyclic graph. A BBN defines various events, the dependencies between them (structure), and the conditional probabilities involved in those dependencies (parameter). A BBN can use this information to calculate the probabilities of various possible causes leading to the actual cause of an event. Figure 3.5 shows an example of BBN modelling of an analog circuit with different BBN blocks. The functionality of each BBN block is modelled by different states (e.g. state 0 denotes good functionality and state 1 denotes failed block). A conditional probability table specifying the dependences between all the states of different BBN blocks can be built automatically or constructed from the knowledge of a domain expert. Authors in [11] proposed to estimate roughly the variables in the conditional probability table initially, then parameter adjustment can be made by gathering production data such as functional test data of failing and/or passing devices and using learning algorithms such as Expectation Maximization method or Conjugate Gradient to determine the conditional probabilities among the dependency parameters.

The advantage of the rule-based diagnosis is its intuitive simplicity. Once the rules are defined, the diagnosis result can be obtained very fast. The disadvantage of this



(a) Analog circuit (b) BBN presentation

Figure 3.5: The BBN modelling of analog circuits for rule-based diagnosis approach [11].

approach is the difficulty to obtain the complete knowledge base of all possible faults. The knowledge base is circuit dependent, i.e., a knowledge base obtained for one type of DUT can not be used to diagnose other DUTs, even a small modification of the DUT's topology can lead to a large modification of the knowledge base. Furthermore, the rule based approach is difficult to apply for large circuits given the complexity of modern ICs. Thus, it can only locate the faulty block in a larger system [66, 11] or an assembly fault (i.e. broken interconnect) [67], but it cannot diagnose faulty components down to the transistor level.

Fault dictionary approach

Figure 3.6 shows a description of fault dictionary approach. A fault dictionary contains a fault list $\{F_j\}_{j=1,...,Q}$ where Q denotes the number of considered faults, and the corresponding diagnostic measurement vectors $\{m_j\}_{j=1,...,Q}$. This fault list can be obtained using historical defect data or an IFA analysis as shown in Chapter 2. The diagnostic measurement vector m can be specification tests, alternate tests, or defect-oriented tests as discussed in the previous section. The fault hypothesis/diagnostic measurement pattern pairs can be generated by sequentially simulating the circuit, inserting each time a single fault in the netlist. The same diagnostic measurement pattern is obtained during diagnosis and is compared to those in the fault dictionary using a similarity measure. The diagnosed fault is the one that pairs up with the most similar diagnostic measurement pattern. This is in essence a pattern recognition (e.g. classification) approach, which can be solved in a deterministic or a probabilistic way. This section provides a detailed description of different fault dictionary methods.

1.k-nearest neighbour (k-NN)

The k-nearest neighbor algorithm (k-NN) is based on closest training examples in

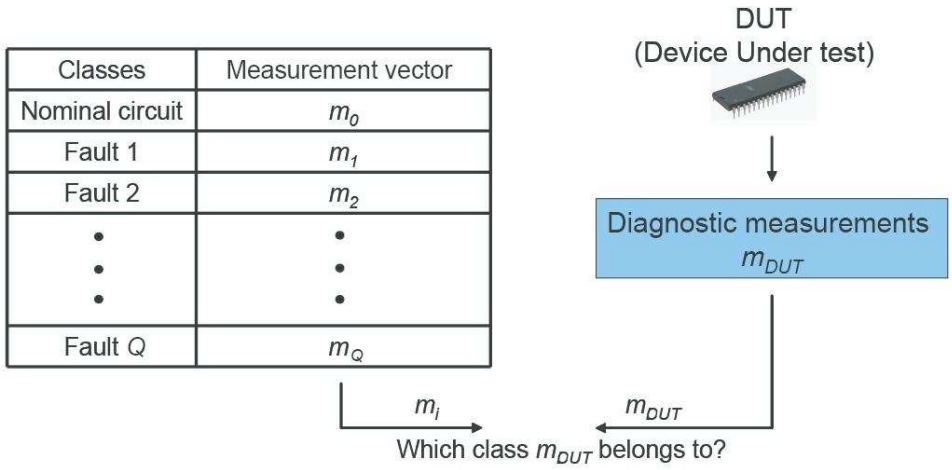


Figure 3.6: Fault dictionary approach.

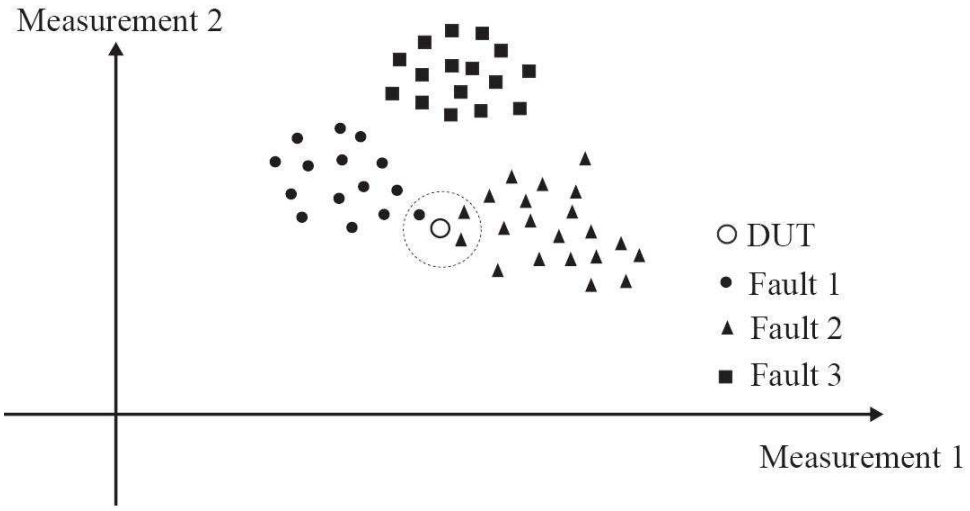


Figure 3.7: The k-NN method in a 2-dimensional diagnostic measurement space.

the feature (diagnostic measurement) space. A DUT is classified by a majority vote of its neighbours, with the DUT being assigned to the class most common amongst its k nearest neighbors, where k is a positive integer number. Figure 3.7 shows the k-NN method in a 2-dimensional diagnostic measurement space.

The training examples are vectors in a d-dimensional feature space, where d denotes the dimension of diagnostic measurements. Each vector has a class label. The training phase of the algorithm requires storing the feature vectors and class labels of the training samples. In the classification phase, k is a user-defined constant, and the DUT is classified by assigning the class which is most frequent among the k training samples nearest to the DUT. As shown in Figure 3.7, if we choose k=3, the diagnosed fault for the DUT will be fault 2 since two samples of fault 2 appear in the first 3 nearest neighbours of the DUT. Setting k=1, the DUT is simply assigned to the class of its nearest neighbour [47].

The advantage of the k-NN method is its simplicity. Distances can be computed using Euclidean distance metric. The drawback of the method is that the choice of k is

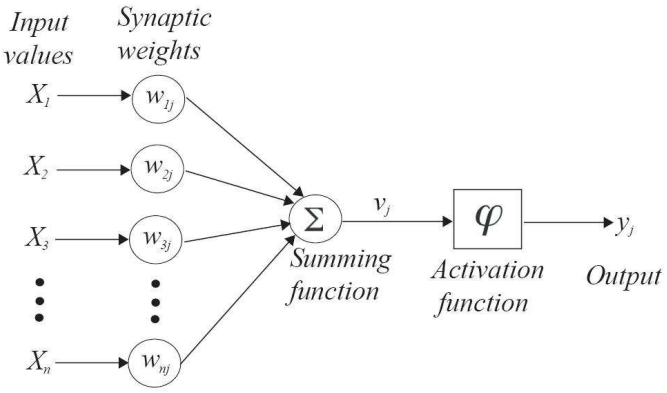


Figure 3.8: A one-layer ANN.

not automatic. The best choice of k depends upon the data; generally, larger values of k reduce the effect of noise on the classification, but make boundaries between classes less distinct. A good k can be selected by a cross-validation method. The accuracy of the k-NN algorithm can be severely degraded by the presence of noise or irrelevant features, or if the feature scales are not consistent with their importance. Furthermore, using the basic "majority voting" classification method, the classes with the more frequent samples tend to dominate the prediction of the new DUTs, as they tend to come up in the k nearest neighbors when the neighbors are computed due to their large number. One way to overcome this problem is to weight the classification taking into account the distance from the DUT to each of its k nearest neighbors.

2. Artificial neural network (ANN)

An artificial neural network (ANN) is a mathematical model or computational model that is inspired by the structure and functional aspects of biological neural networks. A neural network consists of an interconnected group of artificial neurons. In most cases an ANN is an adaptive system that changes its structure based on external or internal information that flows through the network during the learning phase.

An ANN is generally composed of a number of layers. The inputs of each layer are connected with the outputs of the previous layer. Each layer is composed of several "neurons" associated with a weight. At the last layer, all outputs are summed and past through a predefined activation function φ, e.g., a hyperbolic tangent function. Figure 3.8 shows a graphic representation of a one-layer ANN. The output y_j of the ANN can be expressed as

$$y_j = \varphi(\sum_{i=1}^{n} X_i w_{ij}) \tag{3.6}$$

where φ denotes the activation function, X denotes the input vector, n denotes the dimensionality of the input, and w_{ij} denotes the i^{th} weight value of the i^{th} layer.

An ANN is typically defined by three types of parameters: 1) the interconnection pattern between different layers of neurons 2) the learning process for updating the weights of the interconnections, and 3) the activation function that converts a neuron's

weighted input to its output. During the training stage, the weights are updated iteratively with input and output samples in order to minimise training error. In general, there exist two types of training method: supervised learning and unsupervised learning. In supervised learning, each training sample is a pair consisting of an input object and a desired output value (also called the supervisory signal). In unsupervised learning, training samples given to the learner are unlabeled, there is no error or reward signal to evaluate a potential solution.

When ANN is used for fault diagnosis purpose, the input samples of the ANN consist of diagnostic measurement samples and the output samples consist of the corresponding fault classes. In diagnosis phase, the diagnostic measurements of the DUT are used as the input of the ANN and the output value will be the predicted fault class.

The ANN is a machine learning approach used for classification. The advantage of the ANN is their capacity of improving the system by adding new samples in order to update weight values. The drawback is that a large number of training samples may be required to achieve a training precision since any learning machine needs sufficient representative samples in order to capture the underlying structure that allows it to generalize to new cases. Furthermore, the problem of overfitting in the training phase can reduce the generality of the ANN. One way to cope with overfitting is to use a cross-validation method to generalize the trained ANN. In [61, 63], a supervised ANN is used to diagnose catastrophic faults in analog circuits. In [62], an unsupervised ANN is used to diagnose an amplifier using supply current as a fault signature.

3. Support Vector Machine (SVM)

The Support Vector Machine (SVM) is a supervised learning method that analyzes data and recognizes patterns. The standard SVM takes a set of input data and predicts, for each given input, which of two possible classes the input is a member of, which makes the SVM a non-probabilistic binary linear classifier. Given a set of training examples, each marked as belonging to one of two classes, an SVM training algorithm builds a model that assigns new examples into one category or the other.

More formally, a support vector machine constructs a hyperplane or set of hyperplanes in a high dimensional space, which can be used as separation boundary for classification. There are many hyperplanes that might classify the data. One reasonable choice as the best hyperplane is the one that represents the largest separation, or margin, between the two classes. So we choose the hyperplane that maximizes the distance from it to the nearest data point on each side. If such a hyperplane exists, it is known as the maximum-margin hyperplane. Figure 3.9 shows the principle of maximum-margin hyperplane used in SVM.

If in the original space the sets to be discriminated are not linearly separable, the data will be mapped into a much higher-dimensional space using a kernel function k, presumably making the separation easier in that space. Figure 3.10 shows the principle of space mapping using kernel function.

The SVM classifier was originally used to solve binary classification problems. For multi-class classification with Q classes ($Q > 2$), we can reduce the problem into either $\binom{Q}{2}$ or Q distinct binary classification problems and apply either the "one-against-one" or the "one-against-all" strategies [68, 69]. The SVM allocates the separation boundaries

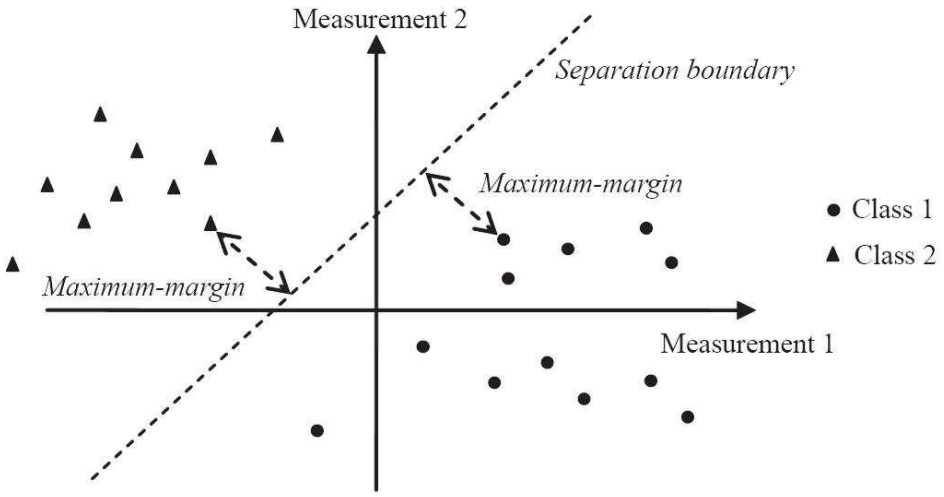


Figure 3.9: Maximum-margin hyperplane used in SVM.

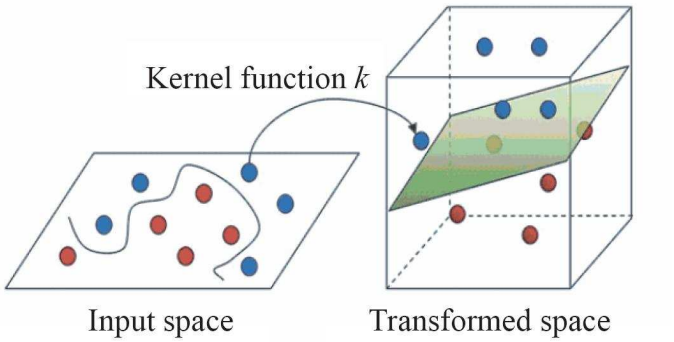


Figure 3.10: Space mapping of SVM using kernel function.

such that they traverse the middle of the distance between the fault clusters. As a result, when the diagnostic measurements are projected in a d-dimensional space, i.e. there will be empty subspaces amidst the fault clusters. This means that SVM will be insensitive to measurement noise or even equipment drifts. SVM can be adapted for non linear regression as well [70].

4. Bayes' rule

In probability theory and applications, Bayes' rule shows how to determine the conditional probability of A given B knowing the conditional probability of B given A and the so-called *prior* or unconditional probabilities of A and B.

Let m be the diagnostic measurement vector of the DUT. E and F are the hypotheses that the DUT is fault free and faulty, respectively. The Bayes' rule is expressed as follows [71]:

$$P(E|m) = \frac{p(m|E)P(E)}{p(m)}, \tag{3.7}$$

$$P(F|m) = \frac{p(m|F)P(F)}{p(m)}, \tag{3.8}$$

where $P(E)$ and $P(F)$ are the *prior* probabilities for hypothesis E and F, $p(m|E)$ and $p(m|F)$ are the conditional probability density function of the diagnostic measurement m given E or F, and $p(m)$ is the *prior* probability density function of the diagnostic measurement m, which is defined as:

$$p(m) = p(m|E)P(E) + p(m|F)P(F), \tag{3.9}$$

As discussed in [71], $p(m)$ is not important as far as decision making is concerned since the denominator term is the same for all fault classes. Suppose that a list of faults can be defined as $\{F_1, F_2, \ldots, F_Q\}$. Thus, for a DUT to be diagnosed, it will most probably have fault F_j if

$$j = \underset{j}{\text{argmax}}\ p(m|F_j)P(F_j), \tag{3.10}$$

where the conditional probability $p(m|F_j)$ can be obtained by Monte Carlo simulation and the prior probability $P(F_j)$ can be obtained by an IFA analysis as discussed in Chapter 2.

The Bayes' rule is a probabilistic diagnosis approach which derives the likelihoods of faults. This allows to analyze the misdiagnosed circuits and the resulting ambiguous groups, which is not possible using a deterministic way. In [71, 72], faults are diagnosed by assuming a Gaussian distribution for $p(m|F_j)$, the mean value μ_j and the variance var_j of $p(m|F_j)$ are estimated by performing a Monte Carlo simulation.

5. Quadratic discriminant analysis

The quadratic discriminant analysis is used in machine learning and statistical classification to separate measurements of two or more classes by a quadric surface. It is a more general version of the linear classifier. Figure 3.11 shows an example of linear and quadratic discriminant analysis in a 2-dimensional diagnosis measurement space. As shown in [73], by assuming that the diagnostic measurement vector m is normally distributed, the probability density function of m for fault j can be expressed as:

$$f_j(m) = \frac{1}{|\sum_j|^{1/2}(2\pi)^{p/2}} \cdot \exp(-\frac{1}{2}((m-u_j)^T \sum_j^{-1} (m-u_j))) \tag{3.11}$$

where m is the diagnostic measurement vector defined above, μ_j is a vector containing the mean value of m over N Monte Carlo simulations, p is the dimensionality of the vector m, and the symbol $\sum_j$ is the covariance matrix of diagnostic measurement vector of fault j. The quadratic discrimination score $d_j(m)$ for fault j is then defined as:

$$d_j(m) = \ln|\sum_j| + ((m-u_j)^T \sum_j^{-1} (m-u_j)) + \ln(p_j) \tag{3.12}$$

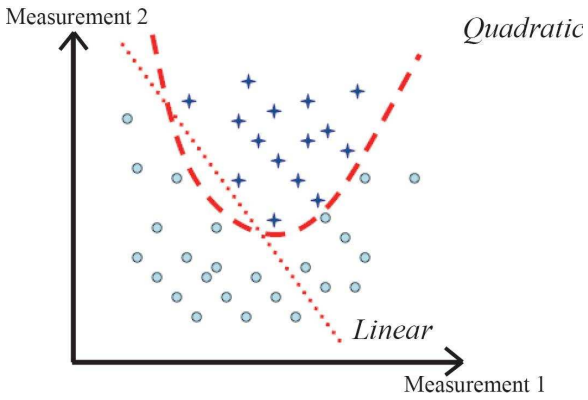


Figure 3.11: Linear and quadratic discriminant analysis in a 2-dimensional diagnosis measurement space.

where p_j is the prior probability of the fault j, which can be obtained by yield simulations or an IFA analysis based on historical fail data. The DUT will most probably have fault j if

$$j = \underset{j}{\operatorname{argmin}}\ d_j(m), \tag{3.13}$$

As discussed in [73], equation 3.12 can also be used to screen faulty circuits that were not modelled in pre-diagnosis analysis. This is done by setting a threshold value on $d_j(m)$ such that if the score is greater than the threshold value, we can conclude that the occurred fault has not been modelled in pre-diagnosis phase.

The advantage of the quadratic discriminant analysis is its simplicity of evaluation. However, care must be taken in computing the term $\sum_j$. Often $\sum_j$ is singular because of the presence of linear dependences of diagnostic measurements. This will lead to a singular and non-invertible covariance matrix. Furthermore, diagnostic measurements are assumed to have Gaussian distribution. However, as discussed in [73], even if the component parameters are normally distributed, nonlinearities in circuit operation may skew the distributions of circuit's performances. Thus, more sophistic method such as non parametric estimation may be needed to estimate $f_j(m)$.

Summary

The SBT approach for fault diagnosis has been presented in this section. In this approach, fault simulation is carried out before the test of the DUT by taking into account its topology. Using realistic fault models is very important to improve the efficiency of the fault simulation. Choosing a set of adequate diagnostic measurements is also important to distinguish different faults. As discussed earlier, different test approaches can be applied to diagnose a DUT. Each test approach has advantages and drawbacks, thus, choosing a test approach is a circuit-specific problem. The SBT approach is often used to diagnose catastrophic faults or parametric faults with fixed values.

3.3.2 Simulation after test (SAT)

As discussed before, the SAT strategy has been designed to solve for values of component parameters, given a set of measured responses and knowledge of the DUT topology. The "input" parameters consist of a diagnostic measurement vector and the response parameters consist of circuit component parameters. Finally, the component which deviates from its tolerance range is considered to be faulty. Thus, the SAT approach is generally used to diagnose parametric faults. This section will discuss different methods used in SAT approach.

Explicit nonlinear equations

As shown in [74], fault diagnosis equations of a circuit or system may be expressed in analytical form. These equations deal with the relationship between external diagnostic measurements and the internal component parameters, which can be expressed as:

$$H(s,r) = y/u, \qquad q = 1,\ldots,N_f \tag{3.14}$$

where H denotes the transfer function of the circuit, s is the Laplace $j\omega_q$ variable which denotes different measurement frequencies, N_f denotes the number of frequencies, r denotes the vector of component parameters to be solved, y denotes the diagnostic measurement vector and u denotes the input vector. The equation in (3.14) can be derived analytically using composite circuit transfer functions [75] or a component connection model [74, 76]. The component connection model defined in [74] describes the components and their connections in a circuit by distinct equations in order to explicitly deal with the relationship between the individual component parameters and the composite system parameters. The component input/output equation is defined as

$$b_i(s) = Z_i(s,r_i)a_i(s), \qquad i = 1,\ldots,m \tag{3.15}$$

where $a_i(s)/b_i(s)$ denotes the i^{th} component input/output, m denotes the number of circuit internal components, and $Z_i(s,r_i)$ denotes the transfer function of the i^{th} component which may take the form of R, Ls, or $1/sC$. For notation brevity, the component equations in (3.15) can be combined into a single block diagonal matrix equation

$$b = Z(s,r)a \tag{3.16}$$

where $b = [b_1,\cdots,b_m]^T$, $a = [a_1,\cdots,a_m]^T$, and $Z(s,r) = \mathrm{diag}(Z_i(s,r_i))$. The connection equations of the whole circuit is then expressed as

$$a = L_{11}b + L_{12}u \tag{3.17}$$

$$y = L_{21}b + L_{22}u \tag{3.18}$$

where u and y represent the vectors of accessible inputs and outputs which are available to the test system, and L_{ij} is the connection matrix which can be obtained by inspection

or computing algorithms for large circuits. By combining (3.15), (3.17) and (3.18), the transfer function matrix observable by the test system between the test input and output vectors u and y can be obtained

$$H(s,r) = L_{22} + L_{21}(1 - Z(s,r)L_{11})^{-1}Z(s,r)L_{12} \tag{3.19}$$

Alternatively, (3.14) can be derived using statistical learning and regression. In [47], (3.14) is obtained by building a non linear equation using statistical simulations such as Monte Carlo simulation. Diagnosis consists of solving the component parameter vector r from the diagnostic measurement y. The solvability of (3.14) is defined as [74]:

$$\delta \quad = \quad m - rank(\frac{dH(s_q,r)}{dr}) \tag{3.20}$$

where m denotes the total number of component parameters to be solved, $rank(\frac{dH(s_q,r)}{dr})$ denotes the rank of the Jacobian matrix $\frac{dH(s_q,r)}{dr}$. In order to solve all parameters in the vector r, $rank(\frac{dH(s_q,r)}{dr})$ should be greater than or equal to m. In [76], an iterative algorithm is proposed to solve (3.14) by taking the measurements y' and solving r' in order to minimise the error $|H(s_q,r') - y'/u|$. In each iteration, the vector r is computed using the Newton-Raphson algorithm:

$$\frac{dH(s_q,r^k)}{dr^k}(r^{k+1} - r^k) \quad = \quad -(H(s_q,r^k) - y'/u) \tag{3.21}$$

where r^k is the k^{th} estimation of the solution of (3.14). In order to solve for r^{k+1} in each iteration, $dH(s_q,r^k)/dr^k$ should be inverted, which implies that $dH(s_q,r^k)/dr^k$ should be non singular.

The advantage of the explicit non-linear equation method is its precision. However, no automated method exists to select diagnostic measurements that satisfy the solvability criterion in (3.20). Furthermore, it is not always guaranteed that the Newton-Raphson scheme will converge to a solution and the estimation is very sensitive to measurement noise. In [75], the parameters which cannot be solved are set to their nominal values, thus, they are not considered in diagnosis phase. Moreover, to derive (3.14), the circuit is supposed to be linear, and the non-linear devices such as transistors are linearized around their nominal operation. In case of gross defects which result in large deviations of circuit performances, the effect of non linearity may result in inaccurate parameter estimation.

Sensitivity analysis

The sensitivity matrix of a circuit describes the relationship between the variations of circuit parameters δr and the variations of diagnostic measurements δv. It can be expressed as:

$$U_{m,n} = \frac{\delta v}{\delta r} = \begin{pmatrix} \frac{\delta v_1}{\delta r_1} & \cdots & \frac{\delta v_n}{\delta r_1} \\ \vdots & \ddots & \vdots \\ \frac{\delta v_1}{\delta r_m} & \cdots & \frac{\delta v_n}{\delta r_m} \end{pmatrix} \tag{3.22}$$

where m denotes the number of parameters to be solved and n denotes the number of diagnostic measurements. In [46], U is derived from the behavioral model of the circuit. In [77], time domain measurements are used to compute U. The equation of the circuit at an arbitrary time point t_n is expressed as

$$Cv'_n + Gv_n = w \tag{3.23}$$

where v denotes the output voltage vector, v'_n represents dv_n/dt, G is the resistive element matrix, C is the reactive element matrix, and w is the input vector. In order to solve (3.23), the time interval $(0, T)$ is divided into $N+1$ discrete points $(0, t_1, t_2, \ldots, T_N)$. At each time point, the solution of (3.23) v_n is determined using difference equations, and the sensitivity of the output with respect to all parameters, $\delta v/\delta r$, where $r = \{C, G\}$, is computed with the solution vector of (3.23). In order to compute the sensitivities $\delta v/\delta r$, both sides of (3.23) are differentiated with respect to r:

$$C\frac{\delta v'_n}{\delta r} + G\frac{\delta v_n}{\delta r} + \frac{\delta C}{\delta r}v'_n + \frac{\delta G}{\delta r}v_n = 0 \tag{3.24}$$

(3.24) can be simplified by denoting

$$s_n = \frac{\delta v_n}{\delta r} \tag{3.25}$$

$$u_n = -(\frac{\delta C}{\delta r}v'_n + \frac{\delta G}{\delta r}v_n) \tag{3.26}$$

Finally the sensitivity equation is derived by combining (3.24) and (3.25)

$$Cs'_n + Gs_n = u_n \tag{3.27}$$

The sensitivity equation (3.27) can also be solved by the difference equations to compute s_n.

For a particular DUT, the difference between the diagnostic measurements v and the nominal value v_{nom} is denoted by $\Delta v = v - v_{nom}$. Then the component parameter deviation vector Δr is calculated as

$$\Delta r = (U^T U)^{-1} U^T \Delta v \tag{3.28}$$

The condition to solve (3.28) is that the invert $(U^T U)^{-1}$ exists, therefore, the sensitivity matrix U should be linearly independent and the number of measurements should be greater than or equal to the number of parameters to be solved: $n \geq m$ [46]. In the presence of fault ambiguity, the matrix $(U^T U)$ is not full rank, i.e., the columns of U are not linearly independent, which results in an ill condition of matrix U. Secondly,

even with numerically full rank, the matrix may still be nearly singular, in which case the solution will be unstable [47]. To solve this issue, the authors in [77] have proposed to add new measurements or add additional components of known values between test nodes during testing in order to increase the rank of matrix U. Other algorithms have been proposed to reduce the columns of matrix U in order to obtain the full rank to solve fault ambiguities [77, 78, 46].

In cases of substantial deviations of r, the sensitivity analysis method is inadequate. To solve this issue, an iterative procedure is implemented that requires to update the sensitivity matrix at each step [77, 79]; however, there is no formal proof that guarantees convergence.

Behavioural model

Behavioural-model-based techniques rely on generating an approximate behavioural model of the circuit. Different abstraction levels can be considered to build the behavioural model. The model constructed from the circuit with its nominal operation is referred to as the reference model. During fault diagnosis, diagnostic measurements of the DUT are compared to those of the reference model. A fault is then detected if a difference is found between the response of the DUT and that of the reference model. The reference model is then perturbed until its response matches the faulty response of the circuit. When a match is found, then a component which may have caused the failure is identified.

The behavioural model can be derived from the transfer function of the circuit [80] or high level performances [46]. In [80], identification consists of estimating the values of different coefficients of the transfer function from the measurements of the DUT. Different methods exist to estimate the behavioural parameters. In [81], the maximum likelihood estimation is used to identify the S-parameters. In [82], the genetic algorithm is used to estimate the small-signal parameters of the RF circuits. The relative sensitivities of small-signal parameters on the circuit performances (i.e., S-parameters) are computed for a wide range of frequencies in order to choose frequency points where there is a change in the sensitivity of parameters to attain diagnostic resolution while avoiding duplication of information. Instead of computing the small-signal parameters from the circuit equation, which is time-consuming, a genetic search algorithm where the small-signal parameters constitute the search variables and the S-parameters constitute the objective function has been used. In each iteration, the difference between the solutions of all S-parameter data and the measured S-parameters constitutes the cost function for the search. A sensitivity-guided weight metric is used in the cost function in order to solve the local minimum problem in the search:

$$C(x) = \sum_{i=1}^{N_p} \sum_{j=1}^{N_f} (P_i(j) - P_m(j))^2 W_i^2(j) \tag{3.29}$$

where

$$W_i(j) = \left(\sum_{k=1}^{N_x} |S_{P_i}^{x_k}| \right) \tag{3.30}$$

where x is the set of small-signal parameters, $C(x)$ is the cost for x, N_p is the number of considered S-parameters, N_f is the frequency chosen from sensitivity analysis, P_i is the set of S-parameters obtained through computation at x, P_m is the set of measured S-parameters, W is the weight associated with each S-parameter, N_x is the number of small-signal parameters, and S_P^x denotes the sensitivity of parameter P to the internal small-signal parameter x computed using perturbation-based simulations.

The main difficulty with this approach is that the search towards a match can be computationally intensive. For a complex system, the construction of an accurate behavioural model may be time consuming. Furthermore, if a fault results in a modification of circuit's topology such as a catastrophic fault, the behavioural model is not valid; this may lead to incorrect predictions.

Summary

Unlike the SBT approach, the computation in SAT approach to solve the response parameters knowing the DUT's topology is carried out after the test of the DUT. The parameters to be solved can be design/process parameters, they can also be high-level behavioral parameters. The testability is the main issue in SAT approach, i.e., whether all considered parameters can be solved accurately within an acceptable time using the available measurements. Several methods have been proposed to improve the testability as presented in this section.

In case of large parameter deviation or a complex circuit with a large number of components, solving parameters can be very time-consuming. Moreover, if a fault has modified the circuit topology such as a spot defect, the approach may not be validated. Thus, the SAT approach is typically used to diagnose parametric faults.

3.4 Summary of diagnosis approaches

This section summarizes different approaches presented in previous sections. Several aspects have been considered when comparing the different diagnosis approaches:

- What is the diagnosis aim? The diagnosis can aim at fault detection which determines if a circuit is functional or faulty, fault location which determines the faulty component, or fault identification which determines the faulty value.
- What approach is used? SBT or SAT?
- What is the fault model? How is it constructed?
- How have the diagnostic measurements been chosen? Are they specification, alternate, or defect-oriented test measurements? Is the choice properly justified or is it ad hoc?
- Has the proposed approach been validated by simulation, by IC prototype, or by industrial circuits?
- How is fault ambiguity resolved?

- Are measurement environment and noise considered?

Table 3.1 summarizes the diagnosis approaches proposed in the literature by answering the aforementioned questions. As can be observed in the table, most SBT approaches are used to diagnose catastrophic faults or parametric faults with fixed deviation values, whereas most SAT approaches are used to diagnose parametric faults. Some authors have validated the proposed approaches by industrial devices [67, 83, 11]. However, the case study shown in [83] is not a large-scale analog IC. In [11], diagnosis is carried out at a rather high abstraction level (i.e. block level). In [67], diagnosis only aims at locating an assembly fault (i.e. broken interconnect) in a large system. To resolve fault ambiguities, more measurements can be added as shown in [77, 72, 71]. Diagnosis results for catastrophic faults are shown in terms of correct classification rate, whereas those of parametric faults are shown in terms of parametric estimation error. It is difficult to compare these results since they depend on several factors: considered fault model, complexity of the case study, etc.

The diagnosis approaches listed in Table 3.1 are not exhaustive since fault diagnosis of analog ICs has been a widely investigated domain for several decades. However, catastrophic and parametric faults are considered separately for most of the proposed approaches, which is not the case in a real defect scenario since the failure mechanisms leading to both catastrophic/parametric faults can occur at any stage of IC production as shown in Chapter 2. Moreover, simple fault models have been considered for most of the proposed diagnosis approaches, i.e., fixed value for short/open defects or arbitrary large distribution for parametric deviation. As discussed in Chapter 2, the resistance value of open defects can vary from less than 100 kΩ to several GΩ, whereas the value of short defects can reach 20 kΩ according to a certain distribution. Since the functionality of analog circuits is highly related to the defect resistance value, it is important to consider its probability in building fault models for diagnosis purposes. To solve these issues, a new approach will be presented in the following chapters.

Table 3.1: Summary of diagnosis approaches for analog circuits.

	Diagnostic approach	Fault model	Diagnostic measurements	Validated circuit	Circuit Complexity	Fault ambiguity resolution	Noise considered?	Diagnosis result (CR[11])	Diagnosis result (PR[12])
[40]	SBT	P[1],C[2]	S[6] + DOF [7]	Simulation	Medium	No	N.C.[10]	25%-100%	1%-99%
[84]	SBT	P	DOF	Simulation	Medium	No	N.C.	N.C.	<6%
[85]	SBT	PF[3]	DOF	Prototype	Medium	No	Yes	95%	N.C.
[47]	SBT+SAT	B[4]	DO [8]	Simulation	High	No	N.C.	100%	0.1% - 8.1%
[86]	SAT	P	DO	Prototype	Medium	N.C.	Yes	N.C.	≈ 2%
[87]	SAT	PF	DO	Simulation	Medium	N.C.	No	N.C.	N.C.
[88]	SBT	C	DO	Simulation	Medium	No	N.C.	61%, 3% [14]	N.C.
[77]	SAT	P	DOF	Prototype	Medium	AM [13]	Yes	N.C.	N.C.
[73]	SBT	C	S	Simulation	Medium	N.C.	No	76% - 100%	N.C.
[89]	SAT	B	DOF	Simulation	Medium	N.C.	Yes	N.C.	1.5%
[67]	SBT	C	S+DOF	Industrial	Medium	N.C.	Yes	100%	N.C.
[83]	SBT	C	DO	Industrial	Medium	No	Yes	90%, 3% [14]	N.C.
[75]	SAT	P	DOF	Simulation	Medium	N.C.	Yes	N.C.	0.3% - 0.6%
[11]	SBT	B	S	Industrial	High	No	Yes	N.C.	N.C.
[46]	SAT	B	S	Prototype	Medium	No	N.C.	N.C.	0.15 LSB
[82]	SAT	B	SF [9]	Simulation	Medium	N.C.	N.C.	N.C.	<3%
[72]	SBT	P	DOF	Simulation	Medium	AM	N.C.	86%	N.C.
[76]	SAT	P	DOF	Simulation	Medium	N.C.	No	N.C.	0-0.9%
[90]	SAT	PF	DO	Simulation	Low	N.C.	No	N.C.	0.004%
[74]	SAT	P	DOF	Simulation	Low	N.C.	No	N.C.	N.C.
[79]	SAT	P	S	Simulation	Low	N.C.	N.C.	N.C.	<3.8%
[62]	SBT	CI[5]	DO	Prototype	Medium	No	Yes	77.8%	N.C.
[61]	SBT	PF	DOF	Simulation	Medium	No	N.C.	97% - 100%	N.C.
[71]	SBT	C	DOF	Simulation	Medium	AM	N.C.	99.97%	N.C.
[91]	SBT	C	DOF	Simulation	Medium	No	No	67% - 100%	N.C.

Abbreviations in Table 3.1

[1]Parametric fault modelled by assigning a large parametric distribution
[2]Catastrophic fault model with fixed value
[3]Parametric fault with fixed deviation value
[4]Behavioral fault model
[5]Catastrophic fault model obtained by IFA analysis
[6]Specification test measurements
[7]Defect-oriented test measurements with feature selection or feature extraction optimization
[8]Defect-oriented test measurements
[9]Specification test measurements with feature selection or feature extraction optimization
[10]Not concerned or not mentioned in the text
[11]Diagnosis result for catastrophic fault or parametric fault with fixed deviation value: Correct classification rate
[12]Diagnosis result for parametric fault: Parametric estimation error
[13]Additional measurements added for resolving ambiguity groups
[14]Classification error for multiple faults

3.5 Conclusion

This chapter presented the state of the art of fault diagnosis. Different test approaches for diagnosis purposes have been discussed, including specification, alternate and defect-oriented test. The choice of diagnostic measurements is a circuit specific problem, which depends on the type and complexity of the DUT, the considered fault model, etc. The existing diagnosis approaches can be classified into two categories: SBT and SAT. While the SBT approach aims to diagnose catastrophic faults or parametric faults with fixed deviations, the SAT approach aims to estimate the parametric deviation in case of parametric faults. The advantages and the main issues with the existing approaches are discussed. A new diagnostic approach aiming at improving these issues will be presented in the following chapters.

Chapter 4

Fault diagnosis based on machine learning

4.1 Introduction

In this chapter, we will present a new fault diagnosis approach for analog integrated circuits. Our approach is based on an assemblage of learning machines that are trained beforehand to guide us through diagnosis decisions. The central learning machine is a defect filter that distinguishes failing devices due to gross defects (catastrophic faults) from failing devices due to excessive parametric deviations (parametric faults). Thus, the defect filter is key in developing a unified catastrophic/parametric fault diagnosis approach. Two types of diagnosis can be carried out according to the decision of the defect filter: catastrophic faults are diagnosed using a multi-class classifier, whereas parametric faults are diagnosed using inverse regression functions. This approach will be shown to single out fault scenarios in an RF Low Noise Amplifier (LNA).

4.2 Proposed diagnosis flow

The proposed fault diagnosis flow relies on an assemblage of learning machines that must be tuned in a pre-diagnosis learning phase. A high-level description of the proposed flow is illustrated in Figure 4.1. The diagnosis starts once a faulty circuit is detected, i.e., the DUT fails at least one of its specifications in production or the DUT fails in the field of operation. The diagnostic measurements specified in the pre-diagnosis phase are then obtained. At first, we can rely on a subset of the standard specification-based tests. If the diagnostic accuracy is not sufficient, the complete specification-based test suite can be used or additional special tests can be crafted to target undiagnosed parameters or to resolve ambiguity groups.

As shown in Figure 4.1, the central learning machine is a defect filter that is trained in the pre-diagnosis phase to distinguish devices with catastrophic faults from devices with parametric faults. Thus, the defect filter enables a unified catastrophic/parametric fault diagnosis approach without needing to specify in advance the fault type. We reuse

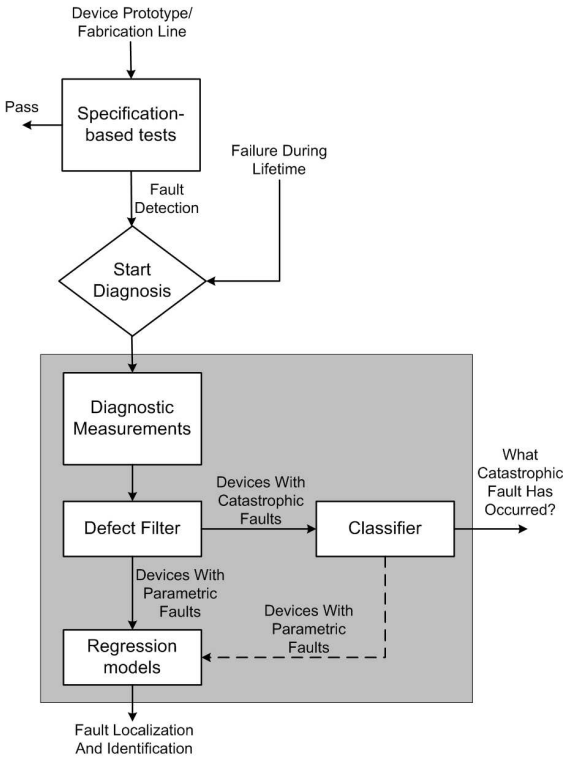


Figure 4.1: Proposed fault diagnosis flow.

here the defect filter proposed in the context of alternate test [58]. This filter relies on a non-parametric estimate $\tilde{f}(m)$ of the joint probability density function $f(m)$, where m is the diagnostic measurements vector. By construction, it is parameterized with a single parameter α, namely $\tilde{f}(m, \alpha)$, which can be tuned in the pre-diagnosis learning phase to control the extent of the filter, i.e. how much lenient or strict it is in filtering out devices. More details about the density estimation approach to construct the defect filter will be given in section 4.2.1.

The defect filter forwards the device to the appropriate diagnosis tier according to the fault type that has been detected. If $\tilde{f}(m, \alpha) = 0$, then the device is inconsistent with the statistical nature of the bulk of the data that was used to estimate the density, thus it is considered to contain a catastrophic fault. A multi-class classifier with Q outputs is used to diagnose catastrophic faults. More details about the diagnosis of catastrophic faults will be given in section 4.2.2.

If $\tilde{f}(m, \alpha) > 0$, the device is considered to contain process variations, i.e. a parametric fault has occurred. For parametric fault diagnosis, we use nonlinear regression functions to predict parametric deviations. More details about diagnosis of parametric faults will be given in section 4.2.3.

The defect filter is always tuned to filter out devices with catastrophic faults. However, this could inadvertently result in some devices with parametric faults being also screened out and forwarded to the classifier. To correct this leakage, the classifier is trained during the pre-diagnosis phase to include detection of devices with process variations as well, i.e. an additional output is added, raising the number of outputs to $Q+1$. Thus, in the unlikely case where a device with a parametric fault is presented to the classifier, the classifier kicks it back to the regression tier.

4.2.1 Defect filter

Why a defect filter

The existing fault diagnosis approaches deal with catastrophic faults and parametric faults separately under certain fault assumptions as presented in Chapter 3. Rule-based and fault dictionary approaches with different pattern recognition methods can be used to diagnose catastrophic faults, whereas for parametric fault diagnosis, explicit non linear equations, sensitivity analysis, and the behavioural model technique can be applied. However, catastrophic and parametric faults can occur at any stage of IC production, as well as in the field, as discussed in Chapter 2. Thus, for a failed DUT, a unified diagnosis approach which makes no assumption on the type of fault is needed when the origin of failure is unknown.

In [58], the defect filter has been used to filter out outliers from the training phase of the regression functions in the context of alternate test. The outliers in an alternate test are devices with physical defects that are induced or enhanced during the IC manufacturing in a random fashion. In the diagnosis context, they are devices with catastrophic faults which can be diagnosed in a SBT approach. On the other hand, devices which are consistent with the statistical nature of the bulk of the data used in the training phase are those with process variations. In the diagnosis context, these are devices with parametric deviations and they are diagnosed in a SAT approach.

Kernel Density Estimation (KDE): A non-parametric estimation approach

As shown in section 4.2, the defect filter is based on the estimate $\tilde{f}(m, \alpha)$ of the joint probability density function $f(m)$, where m is the diagnostic measurement vector and α is a parameter which controls the extent of the filter. For this purpose, we will not make any assumption regarding its parametric form. Instead, we will use non-parametric Kernel Density Estimation (KDE) which allows the observations to speak for themselves. Given a set of N observations of devices under process variations $\{m_1, m_2, \cdots, m_N\}$, where m_i denotes the diagnostic measurement vector of i^{th} observation, the kernel density estimate is defined as [92]

$$\hat{f}(m) = \frac{1}{N \times h^d} \sum_{i=1}^{N} K_e(\frac{1}{h}(m - m_i)) \tag{4.1}$$

where d is the dimensionality of diagnostic measurements, N is the number of observations of devices under process variations, h is a parameter called bandwidth, $K_e(t)$ is the Epanechnikov kernel

$$K_e(t) = \begin{cases} \frac{1}{2} c_d^{-1}(d+2)(1-t^T t) & \text{if} \quad t^T t < 1 \\ 0 & \text{otherwise} \end{cases} \tag{4.2}$$

and $c_d = 2\pi^{d/2}/(d \cdot \Gamma(d/2))$ is the volume of the unit d-dimensional sphere. The kernel density estimate can be interpreted as the normalized sum of a set of identical kernels

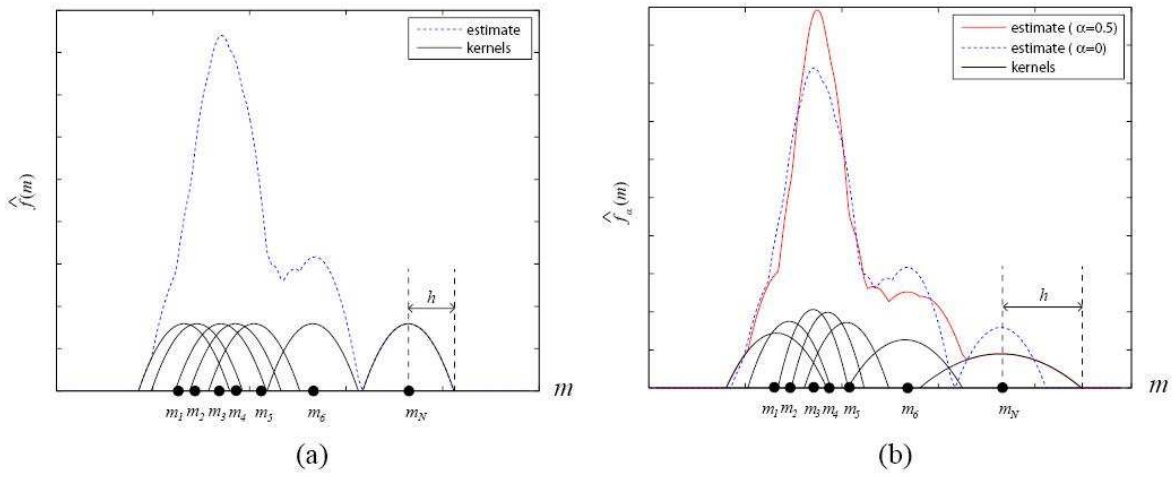


Figure 4.2: KDE method in the 1-dimensional case: (a) estimate in (4.1) where the same kernel is centered on each observation; (b) adaptive estimate in (4.3) where the bandwidth of the individual kernel varies.

centered on the available observations, as shown in Figure 4.2(a) for a 1-dimensional case. The bandwidth h corresponds to the distance between the center of the kernel and the kernel's edge where the kernel density becomes zero.

To control the extent to which the density is nonzero, we can use an adaptive version of the density in (4.1). In particular, we allow the bandwidth h to vary from one observation to another, allowing larger bandwidths for the observations at the tails, as shown in Figure 4.2(b). The adaptive kernel density estimate is defined as [92]

$$\hat{f}_\alpha(m) = \frac{1}{N}\sum_{i=1}^{N} \frac{1}{(h\cdot\lambda_i)^d} K_e\left(\frac{1}{h\cdot\lambda_i}(m-m_i)\right) \tag{4.3}$$

where the local bandwidth factors λ_i are defined as

$$\lambda_i = \{\hat{f}(m_i)/g\}^{-\alpha} \tag{4.4}$$

$\hat{f}(m_i)$ is the pilot density estimate given in (4.1), g is the geometric mean

$$\log g = N^{-1}\sum_{i=1}^{N} \log \hat{f}_j(m_i) \tag{4.5}$$

and α is a parameter which controls the local bandwidth. The larger α is, the larger will be the diagnostic measurement space where the density is nonzero. Figure 4.3 shows the KDE in a 2-dimensional diagnostic measurement space.

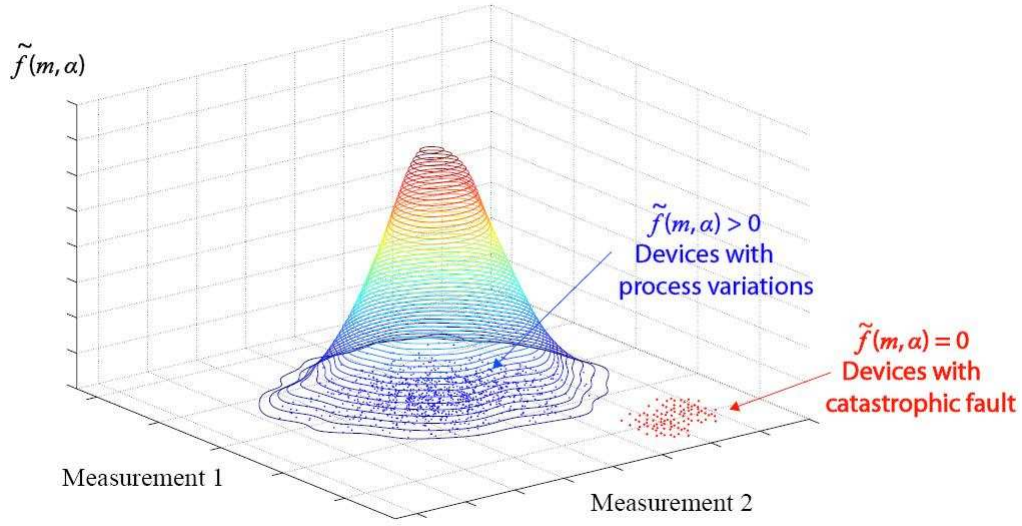


Figure 4.3: Defect filter in a 2-dimensional diagnostic measurement space.

4.2.2 Diagnosis of catastrophic faults: Multi-class classifier

As presented before, for a given diagnostic measurement vector m, if $\tilde{f}(m,\alpha) = 0$ (see red dots in Figure 4.3), then the DUT is considered to contain a catastrophic fault. In this case, the device is forwarded to a classifier that is trained in the pre-diagnosis phase to map any diagnostic measurement pattern to the underlying catastrophic fault. Thus, in this step we follow a fault dictionary approach (see section 3.3.1 of Chapter 3) that employs a multi-class classifier with $Q+1$ outputs, where Q is the number of modeled catastrophic faults in the pre-diagnosis phase.

4.2.3 Diagnosis of parametric faults: Inverse regression functions

If $\tilde{f}(m,\alpha) > 0$ for a given diagnostic measurement vector m of a DUT (see blue dots in Figure 4.3), then the DUT is considered to contain excessive process variations, i.e. a parametric fault has occurred. Figure 4.4 displays the relationships between process variations, performance variations, and alternate measurement variations discussed in section 3.2.2 of Chapter 3. The variations in the DUT performance space S and alternate measurement space M are caused by variations in the manufacturing process parameters and design parameters in space P (shown by green arrows in Figure 4.4). The alternate test approach consists of mapping the low-cost alternate measurements to the performances by means of non-linear regression functions (see blue arrow). In a parametric diagnosis context, the diagnostic measurements (alternate measurements or performances) are known parameters and the process and device parameters are unknown parameters to be predicted. In this work, we predict process and device parameters from alternate measurements or performances by non-linear regression functions, which are

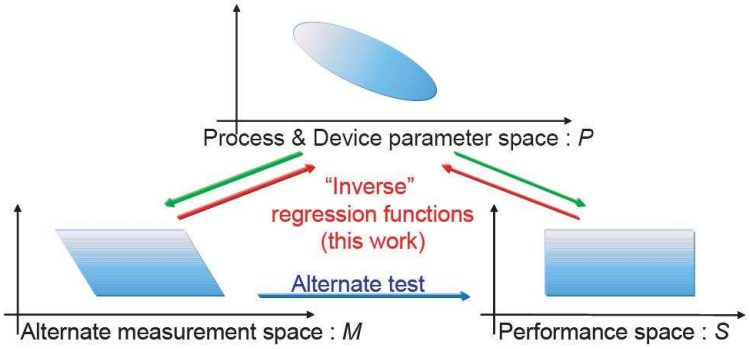


Figure 4.4: Inverse regression function used for parametric estimation.

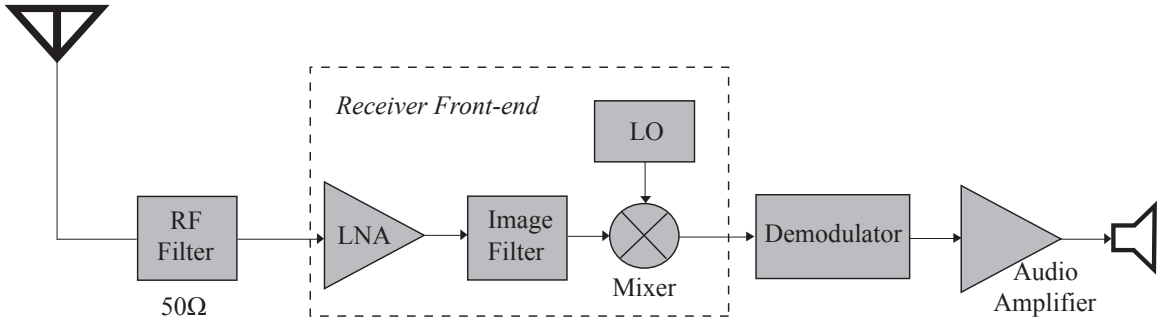


Figure 4.5: A brief description of an RF front-end receiver [12].

named "inverse" regression functions as shown by red arrows in Figure 4.4.

Specifically, we train a set of non-linear regression functions in the pre-diagnosis phase to map the diagnostic measurement pattern (alternate measurements or performances) to the values of all internal circuit parameters of interest. In particular, for n parameters $\{p_j\}_{j=1,\cdots,n}$, we train n regression functions $f_j : m \mapsto p_j$, $j = 1, ..., n$. Unlike prior work on parametric fault diagnosis presented in section 3.3.2, this approach allows an implicit specification of the unknown dependencies between m and all p_j using statistical data and domain-specific knowledge. Thus, it avoids the complications related to an explicit formulation (i.e. diagnosability, convergence, problems with large deviations of p, etc). The main goal is to construct regression models with generalization capabilities, i.e. that can accurately diagnose future devices.

4.3 Case study

4.3.1 Introduction

This section provides a brief description of the case study Low Noise Amplifier (LNA), the fault models used for diagnosis, and the diagnosis tools. An LNA is used to amplify very weak signals captured by an antenna in the beginning of an RF front-end. An RF front-end consists of all the components in the receiver that process the signal at the

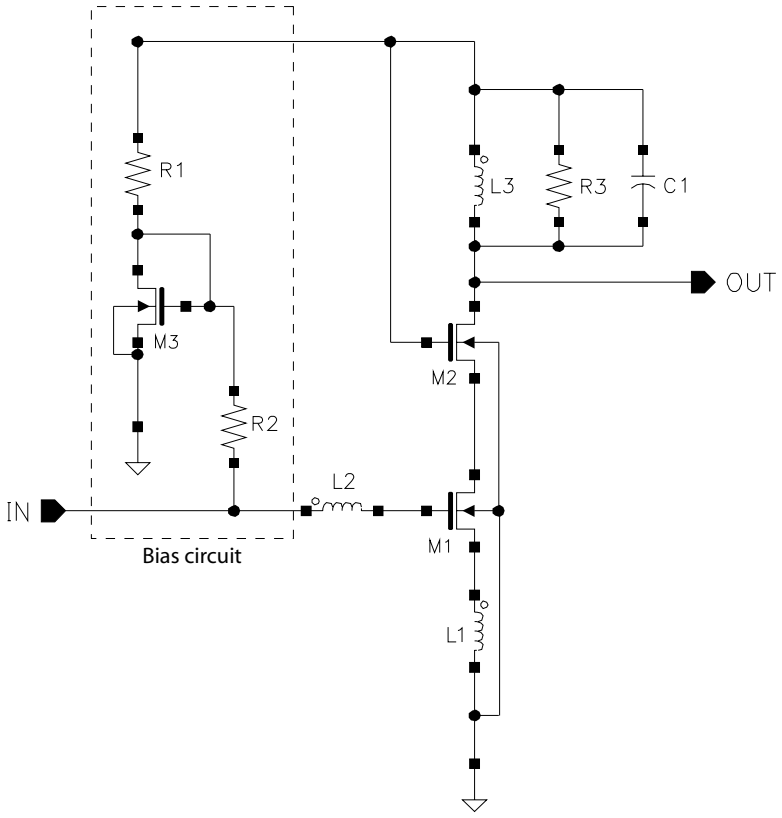


Figure 4.6: Schematic of the LNA under test.

Table 4.1: Performances and specification limits for the LNA under test.

NF (dB)	S_{11} (dB)	S_{12} (dB)	S_{21} (dB)	S_{22} (dB)	1-dB CP (dBm)	IIP_3 (dBm)
≤ 0.7	≤ -8	≤ -35	≥ 11.5	≤ -8.1	≥ -3	≥ 2.8

original incoming radio frequency (RF), before it is converted to a lower intermediate frequency (IF). Figure 4.5 shows a brief description of an RF front-end receiver.

As the first active component in the receiver chain, an LNA should offer sufficient gain and low noise to keep the overall receiver noise figure as low as possible. An LNA should also present an impedance matching, typically at 50 Ω, to the input source and the output load. The input impedance matching is particularly important if a passive filter precedes the LNA, since the transfer characteristics of many filters are quite sensitive to the quality of the termination [93]. The output of the LNA must be equal to 50 Ω so as to drive the image-reject filter with minimum loss and ripple. The characteristic of an LNA is also closely related to the receiver sensitivity and dynamic range.

Figure 4.6 shows the topology of the single-ended LNA under test and the specification requirements are listed in Table 4.1. The LNA is designed for narrow-band applications at 2.4 GHz using the 0.25 μm BiCMOS7RF *ST Microelectronics* technology. The transistors used in this LNA are all CMOS devices as it offers advantages such as low cost, mature process, good thermal conductivity, and excellent integration in the possible future system-on-a-chip (SOC). The transistor M3, together with the resistors

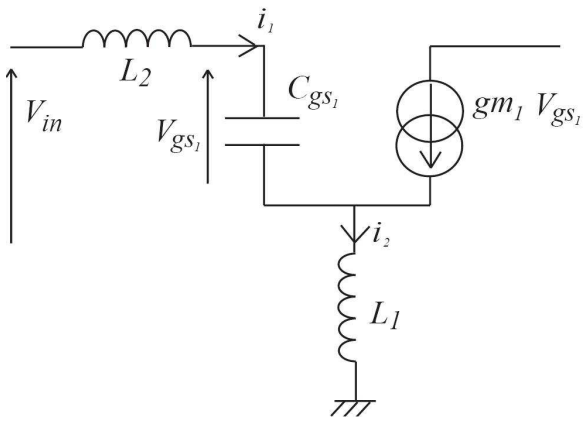


Figure 4.7: Small-signal equivalent circuit of the input stage of the LNA.

R1 and R2 form the bias circuit. M3 essentially forms a current mirror with M1, and its width is some small fraction of M1's width to minimize the power overhead of the bias circuit. The current through M3 is set by the supply voltage and R1 in conjunction with the gate-source voltage V_{gs} of M3. The resistor R2 is chosen large enough to isolate RF signals from the bias block. In a 50-Ω system, values of several hundred ohms to several kilohms can be used for R1 and R2 [93]. Transistor M1, together with inductors L1 and L2 form a common-source input stage of the LNA, M2 is the isolation transistor and the output stage is a RLC network formed by R3, C1 and L3.

4.3.2 Performances of the LNA under test

As the first active component of an RF front-end receiver, the main performances of an LNA include S-parameters which represent input/output impedance return loss (S_{11}/S_{22}), reverse isolation (S_{12}) and gain (S_{21}), Noise Figure, 1-dB compression and Third Intercept Point (IP_3).

S-parameters

The S-parameters include S_{11}, S_{12}, S_{21} and S_{22} expressed in dB. As indicated in the previous section, the input return loss S_{11} is minimized by an impedance match circuit at 50 Ω at the input stage defined by the transistor M1, the inductors L1 and L2. Figure 4.7 shows the small-signal equivalent circuit of the input stage of the LNA, neglecting the gate-drain and source-bulk capacitance of M1. The input impedance can be then computed as

$$Z_{in} = \frac{V_{in}}{I_{in}} = \frac{i_1 \cdot (L_2 s + \frac{1}{C_{gs1} s}) + (i_2) L_1 s}{i_1} \tag{4.6}$$

where

$$i_2 = i_1 + gm_1 \cdot V_{gs1} \tag{4.7}$$

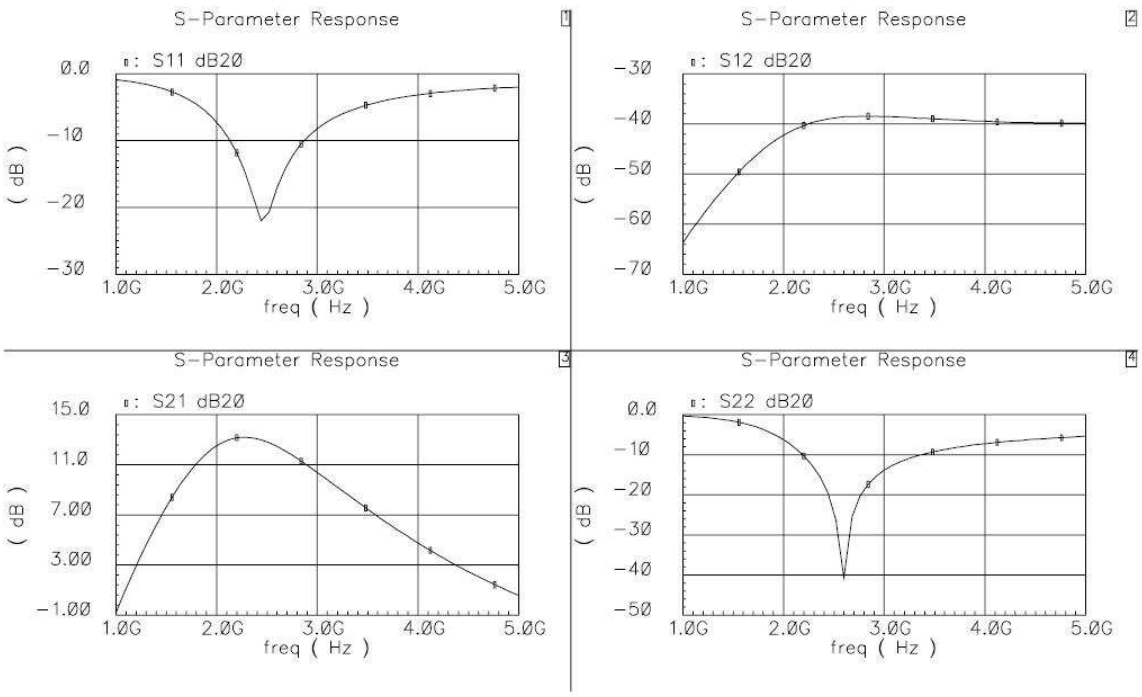


Figure 4.8: Simulation result of S-parameters under nominal condition.

By combining (4.6) and (4.7), the input impedance can be expressed as

$$Z_{in} = (L_1 + L_2)s + \frac{1}{C_{gs1}s} + gm_1\frac{L_1}{C_{gs1}} \tag{4.8}$$

As can be shown in (4.8), the input impedance is equivalent to a RLC network. Thus, proper choice of gm_1, L_1, L_2 and C_{gs1} yields 50-Ω real part. L_1 is the degeneration inductor which controls the real part of the input impedance. Since the input impedance is purely resistive only at resonance, an additional degree of freedom, provided by the inductor L_2, is needed to guarantee this condition. This structure provides a narrowband impedance match. As discussed in [12], at high frequencies, the required value of L_2 becomes comparable with the inductance of the ground bond wire, in this case, multiple bonds or accurate modeling of the wire inductance is needed.

The common-gate transistor M2 plays two important roles by increasing the reverse isolation S_{12} of the LNA. Firstly, it lowers the LO leakage produced by the following mixer. Secondly, it improves the stability of the circuit by minimizing the feedback from the output to the input. The same circuit without isolation transistor M2 would be prone to oscillation [12]. S_{21} represents the gain, which is mainly defined by the common-source input stage of the LNA. The output return loss S_{22} is minimized by an impedance match circuit at 50 Ω in the output stage. The output stage is a RLC network as can be seen in Figure 4.6. Proper choice of R_3, C_1 and L_3 yields 50-Ω real part at resonance.

Figure 4.8 shows the simulation results of the four S-parameters from 1 GHz to 5 GHz with all design parameters at their nominal values. As can be seen, the LNA is well designed for narrow band applications at 2.4 GHz.

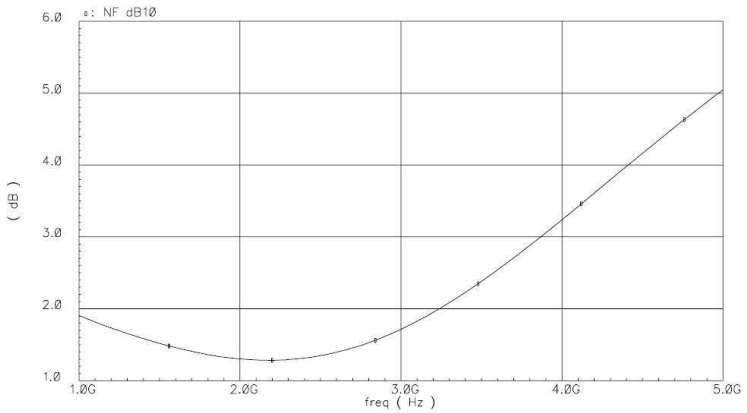


Figure 4.9: Simulation result of Noise Figure under nominal condition.

Noise Figure

Noise Figure (NF) is a measure of degradation of the signal-to-noise ratio (SNR), caused by components in a signal chain. It is defined as

$$NF = 10 \log \frac{SNR_{in}}{SNR_{out}} = SNR_{in,dB} - SNR_{out,dB} \tag{4.9}$$

where SNR_{in} and SNR_{out} are the input and output power signal-to-noise ratios, respectively. $SNR_{in,dB}$ and $SNR_{out,dB}$ are their values in dB. Noise figure is a measure of how much the SNR degrades as the signal passes through a system. For a cascade of stages, the overall noise figure can be obtained in terms of the NF and gain of each stage by Friis' equation:

$$NF_{tot} = 1 + (NF_1 - 1) + \frac{NF_2 - 1}{G_1} + \cdots + \frac{NF_n - 1}{G_1 G_2 \cdots G_{n-1}} \tag{4.10}$$

where NF_i denotes the NF of the i^{th} stage, G_i denotes the gain of the i^{th} stage, and NF_{tot} denotes the NF of all cascade stages. As can be seen in (4.10), the overall NF is dominated by the NF of the first few stages in a cascade structure. Figure 4.9 shows the simulation result of the Noise Figure from 1 GHz to 5 GHz under nominal condition.

1-dB compression

As shown in [12], a non-linear time-variant system can be approximately represented by a third-order expression

$$y(t) = \alpha_1 x(t) + \alpha_2 x^2(t) + \alpha_3 x^3(t) \tag{4.11}$$

where $x(t)$ denotes the input of the system with respect to the time, $y(t)$ denotes the output, and $\alpha_{1,3}$ are the coefficients of different orders. If a sinusoid signal $x(t) = A \cos \omega t$ is applied at the input, then it can be shown that

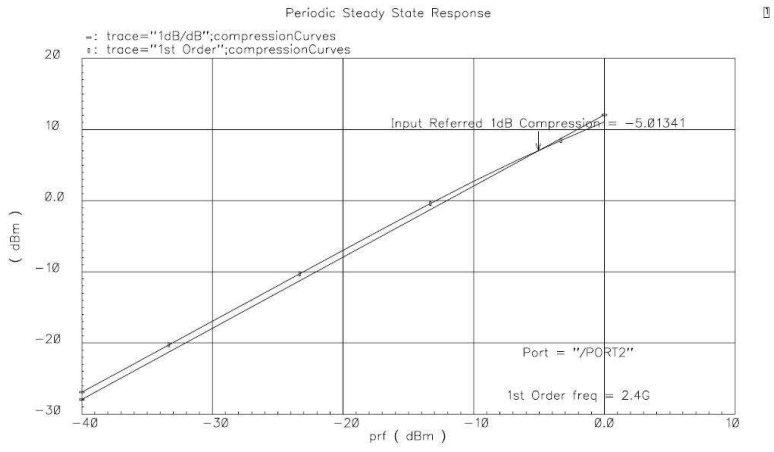


Figure 4.10: Simulation result of 1-dB compression under nominal condition.

$$y(t) = \frac{\alpha_2 A^2}{2} + (\alpha_1 A + \frac{3\alpha_3 A^3}{4}) \cos \omega t + \frac{\alpha_2 A^2}{2} \cos 2\omega t + \frac{\alpha_3 A^3}{4} \cos 3\omega t \tag{4.12}$$

Since the small signal gain of a circuit is usually obtained with the assumption that harmonics are negligible, the gain is then dominated by the term $\alpha_1 A + \frac{3\alpha_3 A^3}{4}$. As shown in [12], in most circuits of interest, the gain approaches zero for sufficiently high input level A if $\alpha_3 < 0$. The 1-dB compression point is defined as the input signal level that causes the small-signal gain to drop by 1 dB. Figure 4.10 shows the simulation result of 1-dB compression point of the LNA under nominal condition. As can be seen in Figure 4.10, the input referred 1-dB compression point is at -5 dBm.

Third-order Intercept Point ($IP3$)

When two signals with different frequencies are applied to a non-linear system, the output in general exhibits some components that are not harmonics of the input frequencies. Assume that $x(t) = A \cos \omega_1 t + A \cos \omega_2 t$ is applied to the non-linear system described in (4.11). It can be shown that third order harmonics at the output are given by $\frac{3\alpha_3 A^3}{4} \cos(2\omega_1 - \omega_2)t$ and $\frac{3\alpha_3 A^3}{4} \cos(2\omega_2 - \omega_1)t$. These harmonics are most important in RF systems [12]. The third-order intercept point ($IP3$) is defined as the input signal level when the third order term $\frac{3\alpha_3 A^3}{4}$ equals to the first order term $\alpha_1 A$. Figure 4.11 shows the simulation result of $IP3$ of the LNA under nominal condition. As can be seen, the $IP3$ of the LNA is at 4.4 dBm.

For a cascade of stages, the overall third-order intercept point A_{IP3} can be obtained in terms of the $IP3$ and gain of each stage [12]:

$$\frac{1}{A_{IP3}^2} = \frac{1}{A_{IP3,1}^2} + \frac{G_1^2}{A_{IP3,2}^2} + \frac{G_1^2 G_2^2}{A_{IP3,3}^2} + \cdots + \frac{G_1^2 G_2^2 \cdots G_{n-1}^2}{A_{IP3,n}^2} \tag{4.13}$$

where $A_{IP3,i}$ denotes the $IP3$ of the i^{th} stage, G_i denotes the gain of the i^{th} stage, and A_{IP3} denotes the $IP3$ of all cascade stages. As can be seen in (4.13), the overall $IP3$ is

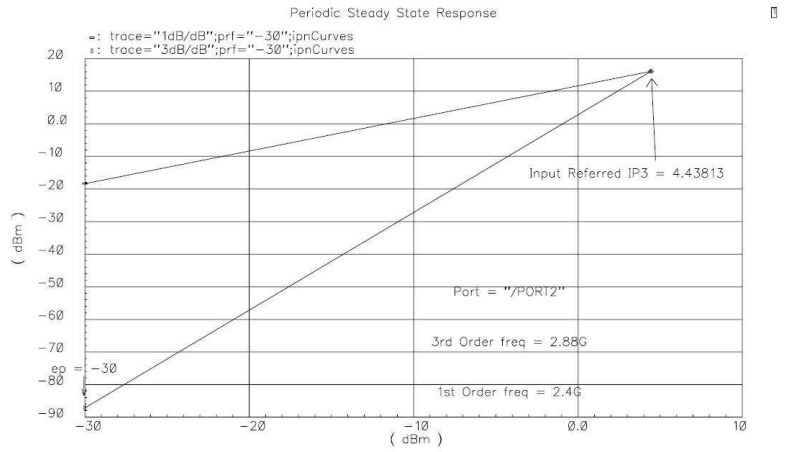


Figure 4.11: Simulation result of IP_3 under nominal condition.

dominated by the $IP3$ of the latter stages if each stage in the cascade has a gain greater than unity.

4.3.3 Fault model

As discussed in Chapter 2, in a production environment, global parametric deviations can be readily detected at wafer-level using process monitors in the scribe lines. Moreover, it is assumed that the root-cause of failure during the lifetime of the IC is localized. Thus, for the purpose of diagnosis, our fault model includes (a) catastrophic faults in the form of short and open circuits and (b) parametric faults that account for location-dependent process deviations. Figure 4.12 shows a description of the fault models used for the LNA.

We model short circuits in passive components and transistor terminals pairs with a 1 Ω resistor. Open circuits in the metal and polysilicon lines are modeled with a 10 MΩ resistor (an open at the gate of M3 is modeled by a broken trace since M3 operates in DC). In total, there are 23 catastrophic faults, which are listed in Table 4.2. In the abbreviation term x_XX_yz, x denotes the catastrophic fault type (x=s for short circuit and x=o for open circuit), XX denotes the affected component, and yz concerns only the transistors and denotes the terminals pair (g=gate, d=drain, and s=source).

We model parametric faults as large deviations in the passive components and in the low-level transistor parameters (i.e. oxide thickness, substrate doping concentration, surface mobility, flatband voltage, etc.). Large parametric deviations in passive components are imposed by simply distorting their fault-free distribution to have a larger standard deviation. With respect to low-level transistor parameters, we noticed in the design kit of *STMicroelectronics* that they are parameterized with a single variable t with nominal value $t = 0$. Thus, denoting these parameters by $q_1, ..., q_k$, the transistor model consists of intricate functions of the form $q_i = f_i(t, q_1, ..., q_{i-1}, q_{i+1}, ..., q_k)$. A Monte Carlo simulation is then enabled by simply varying t uniformly around $t = 0$ with standard deviation σ_t. This observation allowed us to generate realistic faulty transistor models by assigning a larger standard deviation $\beta_t \cdot \sigma_t$, $\beta_t > 1$. Intuitively, deviations in

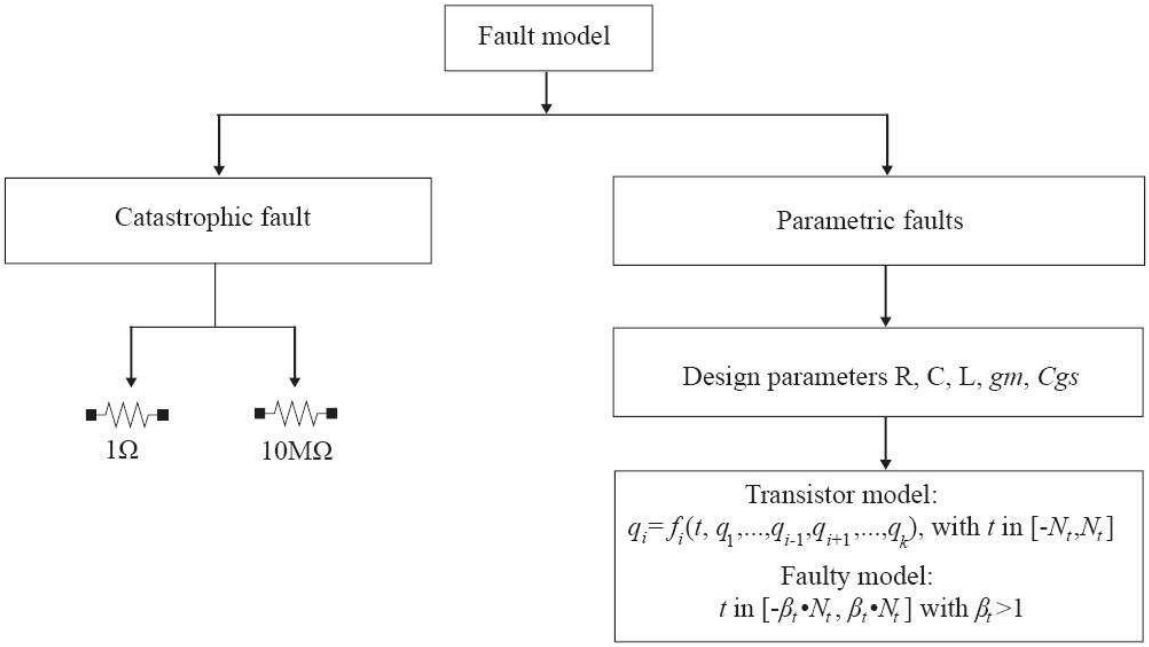


Figure 4.12: Fault models used for the LNA.

Table 4.2: List of catastrophic faults.

Fault	*Faulty Component*
F1	s_M3_gs, s_M3_ds
F2	s_M1_ds
F3	s_M1_gs
F4	s_M1_gd
F5	s_M2_ds
F6	s_M2_gd, s_L3, s_R3, s_C1
F7	s_M2_gs
F8	o_M3_d
F9	o_M3_g
F10	o_M3_s
F11	o_M1_g, o_L2
F12	o_M1_s, o_L1
F13	o_M1_d, o_M2_s
F14	o_M2_g
F15	o_M2_d
F16	s_R1
F17	s_R2
F18	s_L2
F19	s_L1
F20	o_R1, o_R2
F21	o_L3
F22	o_R3
F23	o_C1

Table 4.3: List of circuit parameters under diagnosis.

Parameter	*Nominal value*	*Fault-free distribution*	*Distorted distribution*	*RMS prediction error*
C1	500 fF	-5...5%	-40...40%	3.9%
L1	700 pH	-5...5%	-40...40%	3.2%
L2	8 nH	-5...5%	-40...40%	2.1%
L3	6 nH	-5...5%	-40...40%	2.1%
R1	2 KΩ	-5...5%	-40...40%	25.9%
R2	3 KΩ	-5...5%	-40...40%	22.9%
R3	100 Ω	-5...5%	-40...40%	1%
Cgs_1	347 fF	-20.3...23%	-44.4...27.7%	2.7%
gm_1	84 m	-20.3...42.6%	-94.1...79.7%	3.5%
Cgs_2	358 fF	-13.8...17.7%	-34.5...20.8%	2.6%
gm_2	87 m	-18.8...34.5%	-94...70.6%	3.4%
Cgs_3	52 fF	-19.2...22.4%	-22.1...24.4%	3%
gm_3	10 m	-13.1...16.3%	-26.1...42.3%	11.8%

low-level transistor parameters will be reflected in the small-signal parameters. To this end, we deemed efficient to monitor deviations in g_m and C_{gs}.

The first column of Table 4.3 summarizes the circuit parameters that we diagnose in our experiment (13 in total). The second column lists their nominal values. The third column shows minimum and maximum parameter variations observed over 5000 Monte Carlo simulations using *STMicroelectronics* in-house values for the standard deviations. The forth column shows the corresponding parameter variations after having increased the standard deviations. It should be noted that the distortions that we have imposed in the parameter distributions are illustrative and can be changed to accommodate any fault model of this type.

4.3.4 Diagnosis tools: Classifier and regression functions

We use a support vector machine (SVM) classifier [70] as presented in section 3.3.1 of chapter 3. In contrast to other type of classifiers (i.e. neural networks, nearest neighbors, etc.), SVMs allocate the separation boundaries such that they traverse the middle of the distance between the fault clusters. Now, as will be shown later, our fault clusters are cleanly separated when they are projected in the diagnostic measurement space, i.e. there are large empty subspaces amidst the fault clusters. This means that SVMs will be insensitive to measurement noise or even equipment drifts. In addition, SVMs ensure that complexity is controlled independently of the number of diagnostic measurements. SVMs can be adapted for regression as well [70]. In this experiment, we used the *Kernel-based Machine Learning Lab* package [94] in the *R Project* (www.r-project.org) to implement both the classifier and the regression functions based on SVMs.

4.3.5 Pre-diagnosis learning phase

In pre-diagnosis learning phase, we generate data sets to train and validate the learning machines of the diagnosis flow (i.e. defect filter, classifier, regression functions). We have chosen the four S parameters as our initial diagnostic measurements (a DC diagnostic test will be added later to resolve one ambiguity that we found). Each scattering parameter is sampled at 41 frequency points between 1 GHz and 5 GHz with a step of 100 MHz. Thus, in total, we have $4 \times 41 = 164$ diagnostic measurements.

Training and validation of defect filter

For training and validation of the defect filter, we first generate a data set S_1 which contains 10000 LNA instances obtained by Monte Carlo simulation where all circuit parameters are sampled from their distorted distributions in the fourth column of Table 4.3. The hint here is to model larger component variations in the pre-diagnosis phase than those expected in reality. This way, we minimize the probability that the defect filter will screen out devices with excessive parametric deviations and we ensure that future devices will fall in regions where the regression functions are valid, i.e. in regions where there were enough samples during the pre-diagnosis phase to carry out the regression. In other words, S_1 must be information-rich such that the learning machines can generalize for every possible fault scenario.

We then generate another set S_2 which contains 23 subsets S_{2j}, $j = 1, ..., 23$, corresponding to the 23 fault classes in Table 4.2. Each subset S_{2j} contains 100 LNA instances generated by inserting the catastrophic fault j in the netlist and subsequently running 100 Monte Carlo simulations where the rest of the circuit parameters are sampled from their fault-free distributions. Thus, the size of S_2 is $23 \times 100 = 2300$.

The set S_1 is split in two equal sets S_1^t and S_1^v uniformly at random. Similarly S_2 is split in S_2^t and S_2^v. S_1^t is used to build the defect filter, i.e. to generate the density estimate $\tilde{f}(m, \alpha)$ in (4.3) with $N = 5000$. S_1^v and S_2 are used to validate the defect filter. We tested a defect filter with $\alpha = 0$ (this value of α implements a rather strict defect filter, see [58]) which gave optimal filtering: devices in S_2 have a zero density while devices in S_1^v have a nonzero density.

Training and validation of classifier

The classifier is trained using S_1^t and S_2^t and is validated using S_1^v and S_2^v (S_1 constitutes the "process variations" class). The only misclassification occurred between fault classes F8 and F9. Looking at the LNA schematic, it can be observed that faults F8 and F9 have the same effect: the transistor M3 is off. Thus, these two fault classes can be collapsed in one, resulting in an overall 100% classification rate. This example illustrates that the classifier can help us to identify ambiguous catastrophic faults in the pre-diagnosis phase that we missed out by just looking the schematic with the naked eye.

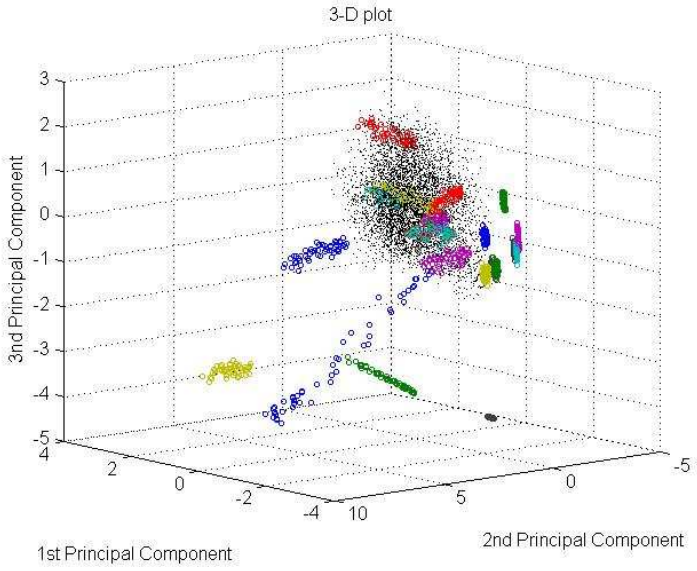


Figure 4.13: Projection of training devices in the top three principal components.

Training and validation of regression functions

The regression models are trained using S_1^t and are validated using S_1^v. The result is shown in the fifth column of Table 4.3 in terms of normalized Root Mean Square (RMS) prediction error, which is defined as

$$\epsilon_j = \frac{\sqrt{\sum_{i=1}^{N}(p_{j,i} - \hat{p}_{j,i})^2/N}}{\overline{p_j}} \tag{4.14}$$

where $p_{j,i}$ is the j^{th} parameter value of the i^{th} device in the validation set S_1^v, N is the total number of devices in S_1^v (i.e. $N = 5000$), and ϵ_j is the normalized RMS error of the j^{th} parameter.

As can be observed in Table 4.3, the regression models can predict accurately multiple parameter variations with the exception of the resistors R_1, R_2 and the transistor M3 in the bias circuit. In retrospect, this could have been anticipated because the bias circuit operates in DC, thus it is not excited by the high-frequency diagnostic measurements. As we will see later, this results in an ambiguity, which calls for additional diagnostic measurements.

To gain some insight about the structure of the data, we perform a Principal Component Analysis (PCA) on the (10000+2300)×164 matrix whose rows correspond to the diagnostic measurement patterns of the devices in S_1 and S_2. Fig. 4.13 shows the projection of these devices in the top three principal components. Fault clusters are represented with different colors, whereas the largely populated "process variation" class is represented with black dots. As can be observed, even in this primitive visualization, fault clusters are cleanly separated.

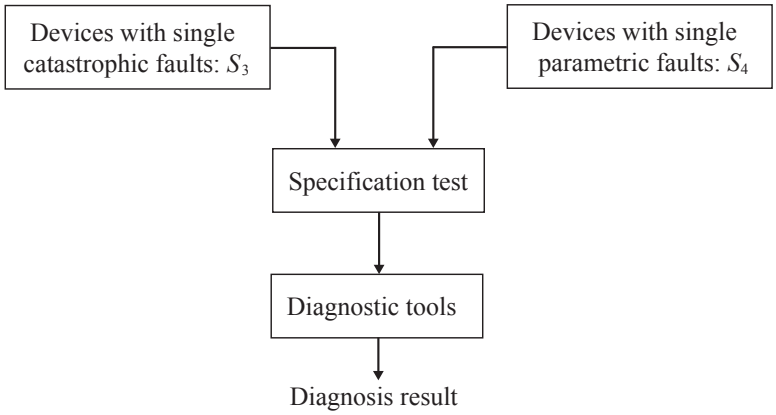


Figure 4.14: Fault injection scenario.

4.3.6 Diagnosis phase

In the diagnosis phase, we generate a fault scenario that may occur to evaluate the generalization of the proposed diagnosis flow. Figure 4.14 shows the fault injection scenario. The set S_3 is generated independently in the same way as S_2. This set corresponds to 23 single catastrophic fault scenarios. The set S_4 contains 20 subsets S_{4j}, $j = 1, ..., 20$, corresponding to the 20 single parametric fault scenarios shown in the first column of Table 4.4. For the passive components, we consider $\pm 30\%$ deviations. For the transistors, we distort the mean value of t in two directions (M_i+ means positive direction and M_i- means negative direction) such that the inflicted (excessive) variations on g_m and C_{gs} are still within the ranges of the fourth column of Table 4.3. Each subset S_{4j} contains 100 LNA instances generated by inserting the j-th single parametric fault and running 100 Monte Carlo simulations where the rest of (unaffected) parameters are sampled from their fault-free distribution. Thus, the size of S_4 is $20 \times 100 = 2000$.

The devices in S_3 and S_4 undergo specification-based testing, according to Fig. 4.1. All devices in S_3 violate at least one specification and as such are labeled as faulty. However, this is not the case for devices in S_4, as shown in the second column of Table 4.4. Faulty devices are next forwarded to the diagnosis phase where they are first subjected to the defect filter. The defect filter fails to characterize correctly a single device with parametric fault L2+30%, which is erroneously screened out and forwarded to the classifier. However, the classifier maps it to the "process variation" class and kicks it back to the regression tier as indicated by the dashed arrow in Fig. 4.1. The rest of devices with catastrophic faults are all correctly classified, thus we conclude that catastrophic fault diagnosis succeeds 100%.

All faulty devices in S_4 are forwarded to the regression tier. The third column of Table 4.4 shows the RMS prediction error of the parameters that deviate in each fault scenario and Fig. 4.15 plots the situation for L2 and R3. Note that the RMS prediction error of the "fault-free" parameters in each scenario is similar to this of Table 4.3 (in general it is even smaller since large errors typically correspond to excessive deviations).

Table 4.4: Single soft fault scenarios.

Single fault scenarios	*Number of faulty circuits /100*	*RMS error of estimated values*
C1+30%	69	1.9%
C1-30%	0	-
L1+30%	74	1.5%
L1-30%	0	-
L2+30%	17	1.9%
L2-30%	81	1.9%
L3+30%	88	1.5%
L3-30%	0	-
R1+30%	0	-
R1-30%	0	-
R2+30%	0	-
R2-30%	0	-
R3+30%	100	0.006%
R3-30%	42	1.3%
M1+	19	cgs_1 : 2.3% gm_1 : 1.2%
M1-	4	cgs_1 : 1% gm_1 : 1%
M2+	0	-
M2-	0	-
M3+	16	cgs_3 : 1.9% gm_3 : 5.1%
M3-	94	cgs_3 : 3.2% gm_3 : 3.1%
Total	604/2000	-

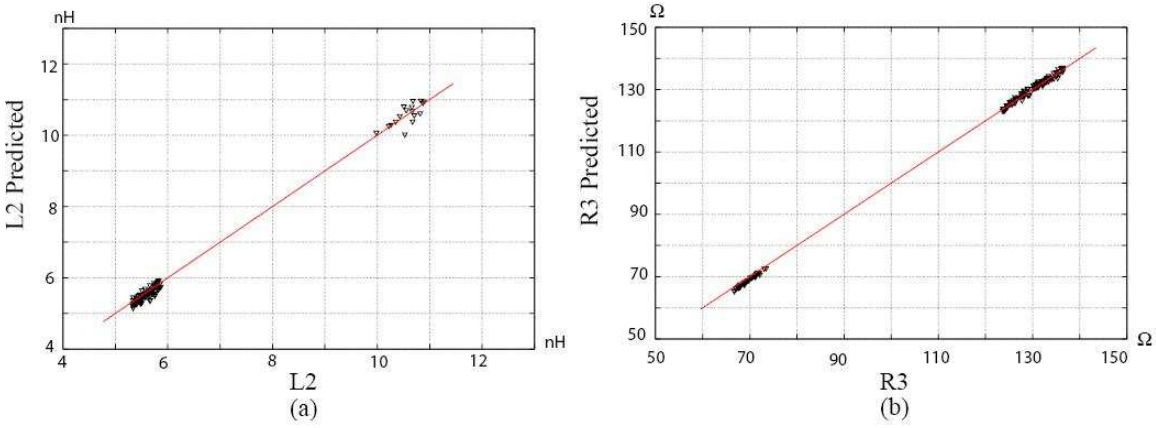


Figure 4.15: Comparison between target and predicted values for (a) L2 (b) R3.

Fault ambiguity analysis

Fault ambiguities have been found in diagnosing parametric faults. For instance, an excessive deviation of the transconductance gm_1 of the transistor M1 can be found at the same time with an excessive deviation of one of the passive components L3, R3 and C1 at the output stage. However, the generated fault scenario is under single fault assumption. Indeed, as the transconductance gm of a MOS device is not an independent parameter, it can be expressed as [95]:

$$gm = k' \frac{W}{L}(V_{gs} - V_{th})(1 + \lambda V_{ds}) \tag{4.15}$$

or

$$gm = \frac{2I_D}{V_{gs} - V_{th}} \tag{4.16}$$

where V_{gs} and V_{ds} are the gate-source and the drain-source voltage of the device, V_{th} is the threshold voltage, W/L is the gate width/gate length, k' is the technology parameter depending on the charge-carrier effective mobility μ_n and the gate oxide capacitance per unit area C_{ox}, λ is the channel-length modulation parameter characterizing the early effect, and I_D is the drain current.

As can be seen in (4.15) and (4.16), the transconductance gm of a MOS device depends on the device (k', W, L, V_{th}, etc.), as well as the bias condition (V_{gs}, I_D, etc.). Thus, an excessive deviation of gm_1 could be induced by (a) a parametric fault in the transistor M1, (b) a deviation in one of the three output passive components L3, R3 and C1 which further affects the drain current I_D of M1, and (c) a deviation in the bias circuit which affects the gate-source voltage V_{gs} of M1. Similar observations can be found for the transistor M2. As a result, a fault in any passive component or in the bias circuit will also impact g_{m1} and g_{m2}.

Also, a parametric fault in M2 does not render the circuit faulty (see zero M2 entries in Table 4.4). Recall from section 4.3.5 that the components of the bias circuit cannot be diagnosed by high-frequency measurements; hence, the predicted deviations of R1, R2, or M3 are not genuine and, thereby, are disregarded. Finally, under the single fault assumption, the probability of two fault scenarios occurring at the same time is negligible.

Based on the predicted values of parameters and the above observations, we define the following diagnosis rules: (a) if g_{m1} and g_{m2} deviate at the same time when a passive component deviates, then the faulty component is the passive component. (b) If both g_{m1} and g_{m2} deviate, then the faulty component is M1 or is located in the bias circuit. The latter rule leads to the only ambiguity so far. Now, note that the LNA fails if a fault within the bias circuit results in a dramatic decrease of the DC bias point of M1 and/or the input impedance of the bias circuit. Thus, this ambiguity can be resolved in part by measuring the gate-source voltage V_{gs3} of M3 (the gate of M3 is not an RF sensitive node). Two follow-up rules to rule (b) above are: (c) if g_{m1} deviates and V_{gs3} is outside its tolerance, then M3 is faulty, (d) if g_{m1} deviates and V_{gs3} is within its tolerance, then the faulty component is M1 or is located in the bias circuit. Using rule (c), we were able to diagnose correctly 49 out of the 16+94=110 circuits with faulty M3.

4.4 Conclusion

In this chapter, we presented a new fault diagnosis method that relies on learning machines to answer the principal questions posed in a branching diagnosis flow. A defect filter detects the type of fault (catastrophic or parametric) and forwards the faulty device to the appropriate tier. Devices with catastrophic faults are diagnosed using a multi-class classifier. If the fault that occurred is parametric, then inverse regression functions are used to predict simultaneously a set of predefined design and transistor-level parameters, in order to locate the faulty parameter and identify its value. In general, some auxiliary circuit-specific fault diagnosis rules are required to resolve ambiguities. This was demonstrated with an LNA example with high overall diagnosis success.

Chapter 5

Bayesian Fault diagnosis based on non-parametric density estimation

5.1 Introduction

In this chapter, a Bayesian fault diagnosis scheme is presented. We focus on spot defect diagnosis since they are considered to be the most common defects in an IC production environment. As shown in Chapter 2, an IFA can be used to generate a list of spot defect locations according to the layout topology of the device and the defect size/geometrical density distribution. However, the resistance value of the injected short/open defect in an IFA is rather arbitrary (short defects are modeled by extra metal material and open defects are modeled by missing material). In this chapter, we will use non-idealized spot defect models for diagnosis purposes by taking into account the defect resistive and capacitive behavior. The likelihoods in the Bayes rule, i.e. the conditional probability density functions of diagnostic measurements given the presence of specific defects, are estimated using non-parametric kernel density estimation. The case study is the LNA presented in chapter 3 with the defects injected at the layout level. The diagnosis decisions and the subsequent defect ambiguity analysis are demonstrated using post-layout simulations.

5.2 Analysis of spot defect behavior

As already discussed in Chapter 2, although both catastrophic and parametric faults can occur at any stage of an IC's lifetime, spot defects turned out to be the most dominant sources of failure in an IC. To this end, we focus on spot defects in this chapter in order to develop an efficient diagnosis approach.

Traditionally, spot defects are modeled as a complete open or short circuit in the metal lines and they are referred to as "hard" since they lead to a complete malfunction of the circuit. However, not all spot defects can be classified as hard defects. In [39], a tunneling current across the open circuits caused by electromigration was observed, which led to a finite resistance value between the two ends of the open circuits. In

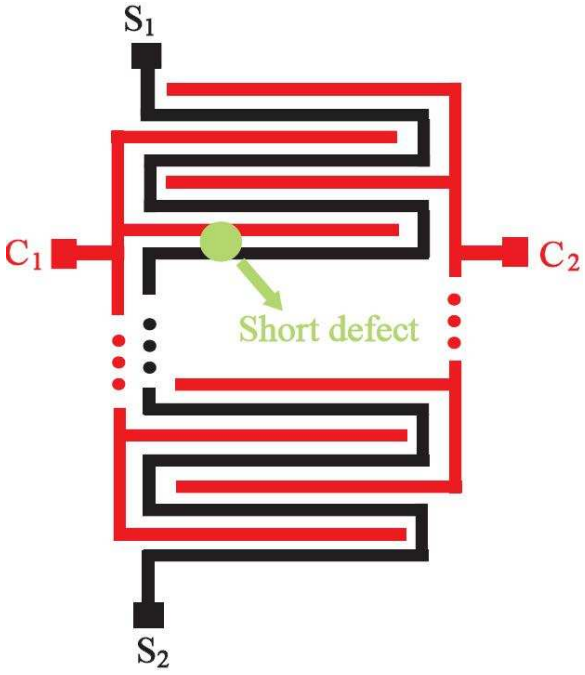


Figure 5.1: Comb-string-comb structure for defect resistance measurement [13].

[59], different types of material in open and short defects have been discussed, including pieces of SiO_2, metal traces, silicon nitrate, polysilicon, and silicide for open defects. For short defects, materials include extra metal, extra polysilicon, etc. Furthermore, different materials in an open defect can result in different values of coupling capacitance. The resistance value of defects vary according to the defect material, e.g. several Ω to several kΩ for short defect resistance values. Then the behavior of defects is modeled with S-parameters that are obtained through low-level physical simulations.

In [13], measurements on defect monitoring wafers are shown in order to evaluate the resistance value of short defects. A defect monitor structure for a CMOS pilot line has been used to measure short defect values. It contains a so-called comb-string-comb structure, shown in Figure 5.1. The string is lying between the two combs and both ends are connected to a bondpad, namely S1 and S2, respectively. Each comb is connected to a bondpad, C1 and C2. Other bondpads can be added to measure any section of the string. A short defect can be detected as a connection between a comb and the string. An open defect can only be detected if present in the string. Resistance measurements between different bondpads are used to detect and measure the resistance value of a defect present in the structure as shown in Figure 5.1. It can be shown that the resistance value of a short defect can be calculated as:

$$R_b = \frac{m}{2} - \beta R_s - R_c \tag{5.1}$$

where

$$m = M_{S1C1} + M_{S2C1} - M_{S2C1} \tag{5.2}$$

Table 5.1: Distribution of short defect resistance R_b [13].

Resistance range	*Percentage*
$R_b < 500\Omega$	69.3 %
$500\Omega \leq R_b \leq 1k\Omega$	26.4%
$1k\Omega \leq R_b \leq 5k\Omega$	2.6%
$5k\Omega \leq R_b \leq 10k\Omega$	0.8%
$10k\Omega \leq R_b \leq 20k\Omega$	0.9%

Table 5.2: Distribution of open defect resistance R_o for one metal layer [14].

Resistance range	*Percentage*
$R_o < 100k\Omega$	6 %
$100k\Omega \leq R_o \leq 1M\Omega$	4%
$1M\Omega \leq R_o \leq 10M\Omega$	5%
$10M\Omega \leq R_o \leq 100M\Omega$	9%
$100M\Omega \leq R_o \leq 1G\Omega$	8%
$R_o > 1G\Omega$	68%

where R_b indicates the resistance value of the short defect, R_s indicates the resistance of one section of the string, R_c indicates the resistance of the contact between the circuit and the probe, β is a location factor, $0 \leq \beta \leq 1$, $\beta = 0$ for location of the defect at the base of the finger and $\beta = 1$ for location of the defect at the tip of the finger, and M_{XY} indicates the resistance measurement from bondpad X to bondpad Y. The same structure can be used to estimate the resistance value of open defects as shown in [14]. Table 5.1 summarizes the distribution of short defect resistance values measured in [13] and Table 5.2 shows the case of open defects for one metal layer measured in [14].

It can be observed from Tables 5.1 and 5.2 that in the worst case, the resistance value of short defects can be as high as 20 kΩ, whereas the resistance value of open defects can be as low as 100 kΩ. Short defects with non-negligible resistance and open defects with finite resistance are referred to as soft defects since they do not necessarily lead to complete malfunction of the circuit. In the limit, the effect of soft defects could be similar to the effect of excessive local process deviations. As shown in Figure 4.13 in Chapter 4, fault clusters with soft defects tend to overlap each other in the diagnostic measurement space unlike the case with hard defects. Thus, a deterministic diagnosis approach can mislead diagnosis decision since it always assigns one fault cluster to the DUT. In this chapter, we will present a probabilistic diagnosis methodology based on Bayes' theorem and non-parametric density estimation.

5.3 Proposed diagnosis approach

The proposed fault diagnosis approach relies on Bayes' theorem and non-parametric kernel density estimation (KDE) to model the resistive behavior of spot defects and derive the probability of occurrence of each defect for a DUT.

5.3.1 Discriminant analysis

Recall from section 3.3.1.4 of Chapter 3 that the probability of a DUT to contain defect F_j is expressed as

$$P(F_j|m) = \frac{p(m|F_j)P(F_j)}{p(m)}, \tag{5.3}$$

where m is the diagnostic measurement vector, $P(F_j)$ is the prior probability for hypothesis F_j, $p(m|F_j)$ is the conditional probability density function of m given F_j, and $p(m)$ is the prior probability density function of m. The DUT will most probably have fault F_j if

$$j = \underset{j}{\text{argmax}}\ p(m|F_j)P(F_j), \tag{5.4}$$

The conditional probability $p(m|F_j)$ can be obtained by Monte Carlo simulation and the prior probability $P(F_j)$ can be obtained by an IFA. Here, for the purpose of simplicity and without loss of generality, we assume that they are equal, i.e. $P(F_i) = P(F_j)$, $\forall i, j$. Under this scenario, a faulty CUT will most likely contain defect F_j if

$$j = \underset{j}{\text{argmax}}\ p(m|F_j), \tag{5.5}$$

In [71], $p(m|F_j)$ is assumed to be normally distributed. The mean value and the variance of m are estimated by performing a Monte Carlo simulation. However, as discussed in [73], even if the component parameters are assumed to be normally distributed, nonlinearities in circuit operation may skew the distributions of circuit performances. Thus, more sophistic methods such as those based on non parametric estimation are needed to estimate $p(m|F_j)$.

5.3.2 Fault diagnosis flow

A high level description of the proposed fault diagnosis flow is illustrated in Fig. 5.2. The pre-diagnosis phase includes defect modeling and defect injection, in order to estimate the densities $p(m|F_i)$. We first generate a list of possible defect locations F_i, $i = 1, \cdots, n$, through a failure analysis. Then, we estimate the probability density function of resistance R and capacitance C associated with each defect. These densities are denoted by $p(R|F_i)$ and $p(C|F_i)$. The density $p(R|F_i)$ is fitted to data using kernel density estimation (KDE) (see section 4.2.1 of Chapter 4) using bounded domain estimation, more detail will be shown in section 5.3.3. The estimation of $p(C|F_i)$ is presented in section 5.3.3.

Once the densities $p(R|F_i)$ and $p(C|F_i)$ are estimated, we can sample them to generate K different scenarios for defect location F_i. In other words, we can generate K different "instances" of the defect F_i, i.e. K different combinations of resistive behaviors. These K defect instances are injected at the layout level during a post-layout Monte Carlo simulation to obtain the corresponding diagnostic measurements m. During the Monte

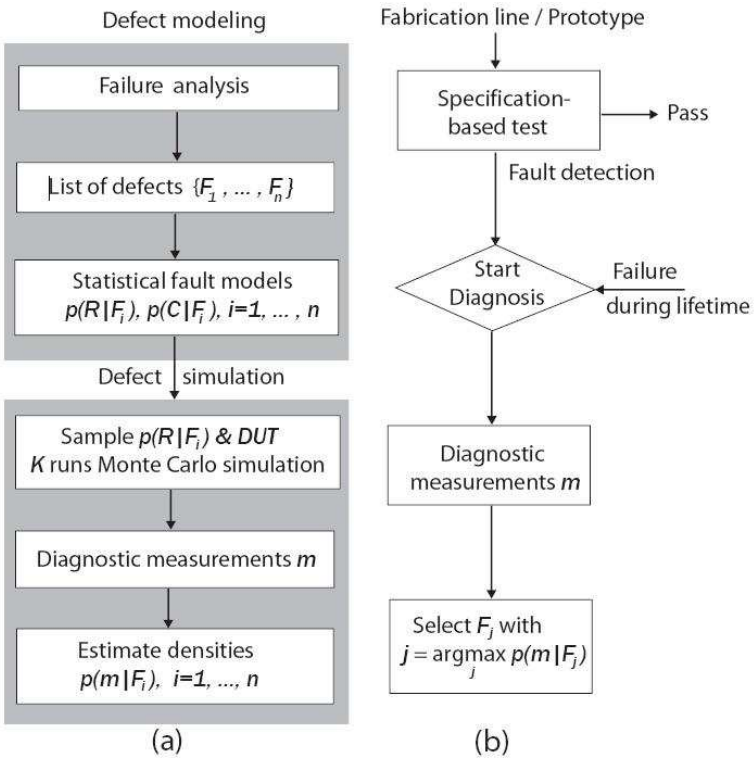


Figure 5.2: Fault diagnosis: (a) extraction of probability density function for the bayesian fault diagnosis framework and (b) fault diagnosis flow.

Carlo simulation, K instances of the circuit and the associated defect are sampled. This simulation includes process and mismatch deviations in the design and random values of the defect parameters R. This way, we collect enough samples to estimate the likelihood $p(m|F_i)$. As before, this estimation is carried out using kernel density estimation (KDE). For example, Figure 5.3 shows the densities $p(m|F_i)$ for three defects in a 2-dimensional diagnostic measurement space.

Once all likelihoods $p(m|F_i)$ are estimated, we can readily use them to diagnose the most probable defect that gave rise to a faulty DUT, given the pattern m, as explained in section 5.3.1 and as shown in the right-hand side of Fig. 5.2.

5.3.3 Fault modeling

Fitting of $p(R|F_i)$: a bounded domain density estimation

As mentioned before, the vast majority of faults in analog ICs that have to be detected during functional or structural testing are caused by local spot defects. Thus, for the purpose of diagnosis, we focus on spot defects to construct fault models. The spot defects are injected at the layout level by analyzing critical defect locations as well as the size of defects. Without doubt, the characteristics of defects change with advances in technology and complete information of defects is usually available only when the technology is already obsolete. Nevertheless, a general trend can be observed, in the sense that a similar distribution of defect values is observed in each technology.

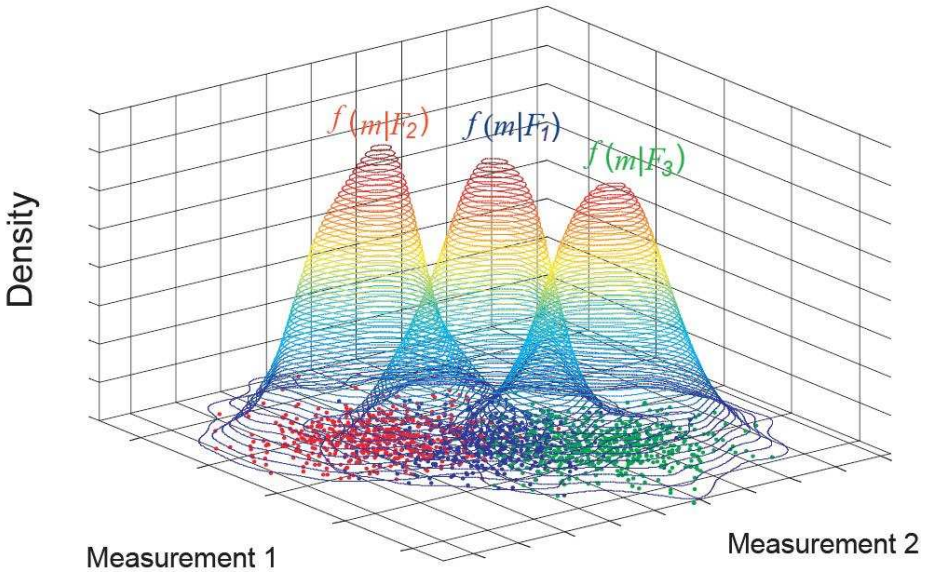


Figure 5.3: KDE method in a 2-dimensional diagnostic measurement space.

It is very often the case that the natural domain of definition of a density to be estimated is not the whole real line but an interval bounded on one or both sides. This is the case for the estimation of the probability density function of defect resistance value $p(R|F_i)$. For example, the resistance R has always a positive value and it is necessary to obtain a density estimate $\hat{p}\,(R|F_i)$ zero for all negative R values. In this work, we use a reflection technique proposed in [92] to carry out bounded domain density estimation. The idea is to have zero density for all negative values of R while keeping the obtained estimates integrated to unity, i.e., $\int_0^\infty \hat{p}\,(R|F_i)\,\mathrm{d}R = 1$. Moreover, the contribution to $\int_0^\infty \hat{p}\,(R|F_i)\,\mathrm{d}R$ of points near zero should be as important as other points well away from the boundary so that the weight of the distribution near zero will not be underestimated.

Specifically, let S_1 denote the original set of resistance samples $\{R_1, R_2, \cdots, R_n\}$. We augment S_1 by adding the reflections of all the points in the boundary, which is zero. The reflected set becomes $\{R_1, -R_1, R_2, -R_2 \cdots, R_n, -R_n\}$. We can name a new set S_2 which denotes the reflected set:

$$S_2 = \{R'_1, R'_2, \cdots, R'_{2n}\} \tag{5.6}$$

Let $p^*(R|F_i)$ denote the density estimated from the set S_2 using Equation (4.3), then an estimate based on the original data set S_1 can be given by

$$\hat{p}\,(R|F_i) = \begin{cases} 2p^*(R|F_i) & \text{for} \quad R \geq 0 \\ \\ 0 & \text{for} \quad R < 0 \end{cases} \tag{5.7}$$

It can be shown that the estimate given in (5.7) will guarantee that $\hat{p}\,(R|F_i)$ is a probability density, i.e., $\int_0^\infty \hat{p}\,(R|F_i)\,\mathrm{d}R = 1$. As discussed in [92], the reflection method can be generalized to the case where the required support of the estimator is a finite interval $[a, b]$. Figure 5.4 shows the estimated probability density function $\hat{p}(R|F_i)$ for short and open defects according to the samples of Tables 5.1 and 5.2.

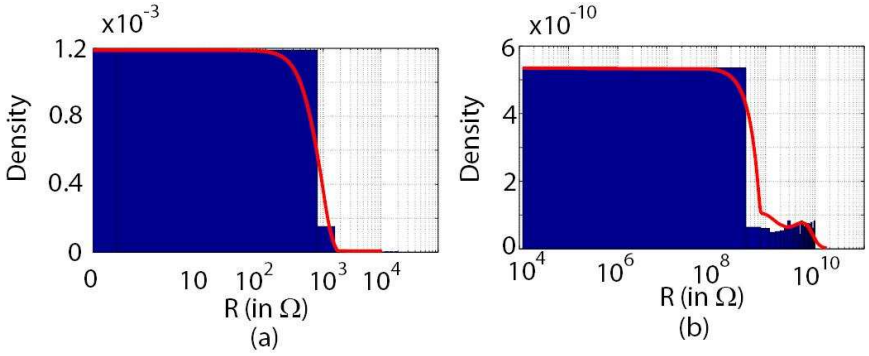


Figure 5.4: Estimated probability density function $\hat{p}(R|F_i)$ for: (a) short defect (b) open defect.

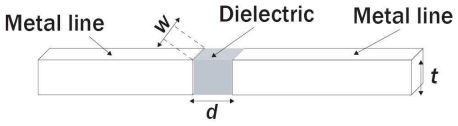


Figure 5.5: Geometry of open defect.

Parasitic capacitance analysis

Parasitic capacitance due to charge coupling between two ends of a metal cut (e.g. open defect) should also be taken into account. The capacitance is expressed as follows (see Figure 5.5):

$$C = \frac{\epsilon \cdot w \cdot t}{d} \tag{5.8}$$

where ϵ is the permittivity of the material between the two ends, w is the width of the metal line, t is the thickness of the metal line, and d is the width of the open defect. In order to evaluate the parasitic capacitance values, we use the permittivity of SiO_2 which is $\epsilon = 3.9 \times 8.85 \times 10^{-12}$ F/m. The value of w depends on the location of each defect and ranges between 1 and 20 μm, and the standard value of t is 2.5 μm for the technology used in the design. We consider the value of d between 0.1 μm and 2 μm [38]. According to (5.8), the parasitic capacitance value between two ends of open defect ranges from 0.04 to 17.2 fF. Thus, the corresponding reactance can be calculated according to $Z_c = \frac{1}{\omega c}$ and lies between 3.8 kΩ and 1.5 MΩ for the interested frequency point at 2.4 GHz. This shows that in the case of RF circuits, we need also to consider the parasitic capacitance created by charge coupling between the two ends of an open defect. As the frequency increases, the reactance of this capacitor will not be negligible. Thus, the open defects are modeled by a resistance in parallel with a capacitance as shown in Figure 5.6. To model the capacitance of open defect and generate the density $p(C|F_i)$, we have assigned a uniform distribution for d from 0.1 μm to 2 μm.

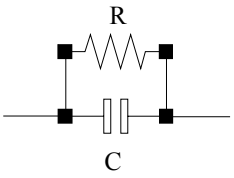


Figure 5.6: Open defect modeling.

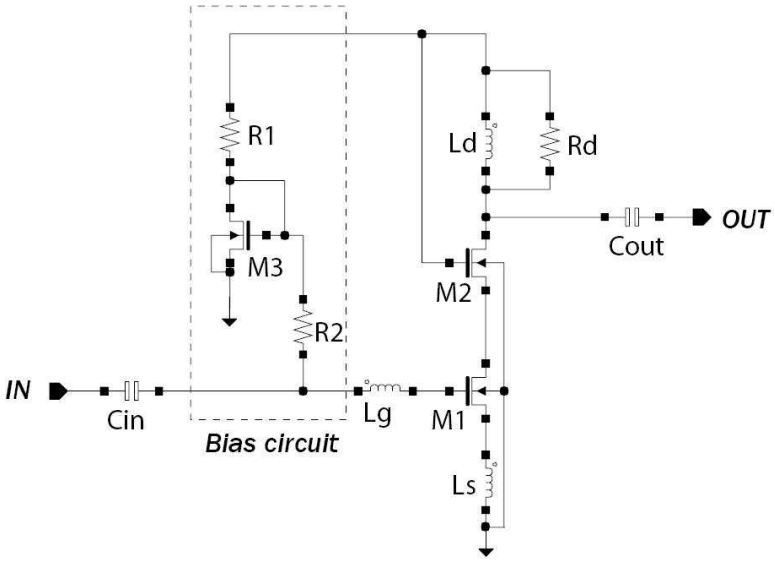


Figure 5.7: Schematic of LNA under test.

5.4 Case Study

5.4.1 Low noise amplifier and its diagnostic measurements

The case study for the non-parametric KDE diagnosis approach is the same LNA shown in Chapter 4 with coupling capacitors integrated in the design, as shown in Figure 5.7. Post-layout fault simulation is carried out with defects modelled as indicated in Section 5.3.3 and injected at the layout level. For this purpose, we use *Calibre* tool of *MentorGraphics* for Design Rule Checking (DRC) and Layout Versus Schematic (LVS). The layout of the LNA is shown in Figure 5.8.

The specification requirements are listed in Table 5.3. With regard to the diagnostic measurements, we chose as our initial diagnostic measurements the four scattering parameters, as well as the noise figure. Each scattering parameter and the noise figure are sampled at 41 frequency points in the range of 1-5 GHz, with a step of 100 MHz. This results in $5 \times 41 = 205$ diagnostic measurements.

5.4.2 Fault modeling phase

In total, 24 fault locations are considered, as shown in the second column of Table

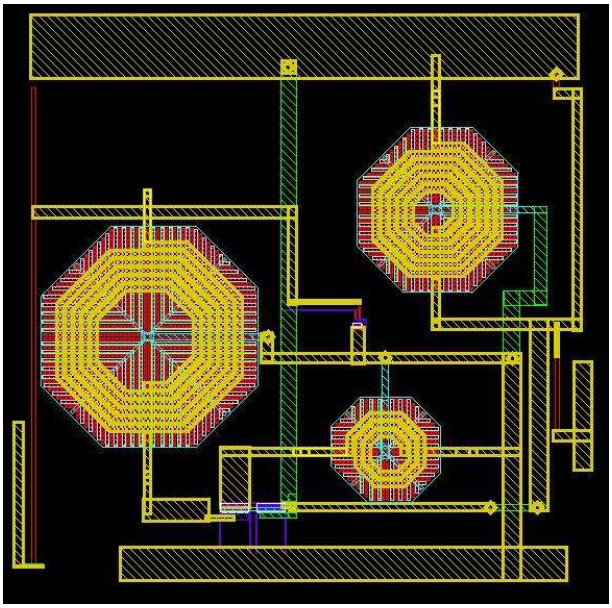


Figure 5.8: Layout of the LNA.

Table 5.3: Specifications of LNA under test.

Performance	*Specification requirements*
Noise Figure (dB)	≤ 2.5
S_{11}(dB)	≤ -12
S_{12}(dB)	≤ -30
S_{21}(dB)	≥ 11
S_{22}(dB)	≤ -12
1-dB Compression(dBm)	≥ -8
IIP_3(dBm)	≥ 2
Stability factor	≥ 1

5.4. This list contains all possible opens and shorts across the circuit components. In the abbreviation term x_XX_yz, x denotes the defect type (x=s for short circuit and x=o for open circuit), XX denotes the affected component, and yz concerns only the transistors and denotes the terminal pairs (g=gate, d=drain, and s=source). The defects which have the same effect on the behavior of the circuit are grouped as a single fault (for example, an open circuit on the drain of M1 and an open circuit on the source of M2 are equivalent). The resistive behavior of the defects is modeled by the densities of Fig. 5.4 and the capacitive behavior by (5.8) using a uniform distribution for the opening width d.

5.4.3 Fault injection phase

The defects are injected at the layout level. Specifically, an open is modeled as a metal trace cut by placing a resistor in parallel with a capacitor across the two edges of the cut. A short is modeled by connecting a resistance between the two implicated

Table 5.4: List of considered defects.

Defect	*Affected component*	*Number of faulty devices/500*	*Number of devices correctly diagnosed (diagnostic rate)*
F1	s_M3_gs s_M3_ds	414	12 (2.9%)
F2	s_M1_ds	130	85 (65.4%)
F3	s_M1_gs	500	463 (92.6%)
F4	s_M1_gd	327	327 (100%)
F5	s_M2_ds	143	78 (54.5%)
F6	s_M2_gd s_Ld s_Rd	500	222 (44.4%)
F7	s_M2_gs	153	44 (28.8%)
F8	s_R1	18	5 (27.8%)
F9	s_R2	88	88 (100%)
F10	s_Lg	491	253 (51.5%)
F11	s_Ls	10	4 (40%)
F12	s_Cin	500	492 (98.4%)
F13	s_Cout	200	186 (93%)
F14	o_M3_g o_M3_d o_M3_s	500	500 (100%)
F15	o_M1_g, o_Lg	500	500 (100%)
F16	o_M1_s, o_Ls	500	492 (98.4%)
F17	o_M1_d, o_M2_s	500	500 (100%)
F18	o_M2_g	500	499 (99.8%)
F19	o_M2_d	500	500 (100%)
F20	o_R1, o_R2	500	2 (0.4%)
F21	o_Ld	500	500 (100%)
F22	o_Rd	500	500 (100%)
F23	o_Cout	500	500 (100%)
F24	o_Cin	500	500 (100%)
Total	-	**8947**	**7252**

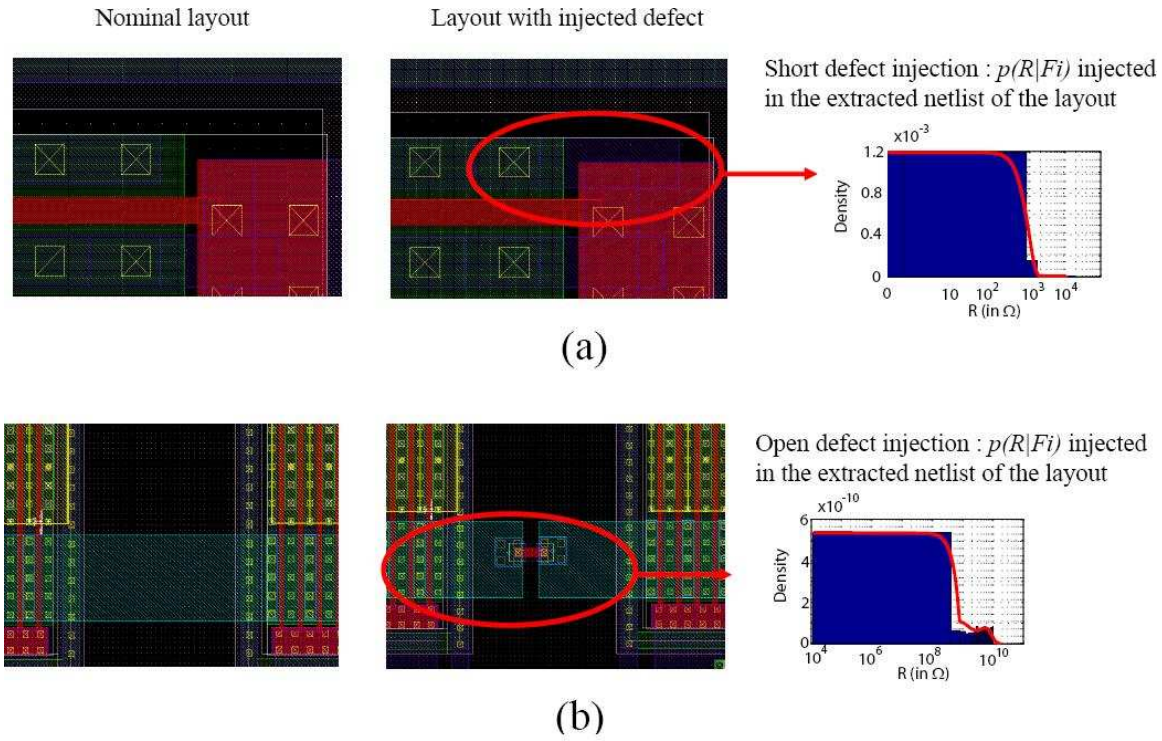


Figure 5.9: Examples of defect resistance injection for (a) F1 and (b) F17.

nodes. Then, the layout is extracted by taking into account all parasitics (e.g. RCc extraction).

To generate the required observations for the estimation of the likelihood $p(m|F_i)$, we generate different defect instances by changing the value of resistance and capacitance in the extracted netlist according to their distributions. Specifically, in a Monte Carlo simulation, the capacitance value in parallel with the open defect is sampled by (5.8) using a uniform distribution for the opening width d between 0.1 μm and 2 μm. For defect resistance value sampling, we sample from the estimated densities $\hat{p}(R|F_i)$ shown in section 5.3.3. Figure 5.9 illustrates examples of defect resistance injection at the layout level for defect F1 and F17. The sampling procedure is shown in Figure 5.10. First, we obtain n samples of defect resistance value from defect characterization test as shown in [13, 14] (see Section 5.2). Then we estimate the probability density function $\hat{p}(R|F_i)$ according to (5.7). Once $\hat{p}(R|F_i)$ has been estimated, we can generate a new set S' containing n' ($n' >> n$) samples from $\hat{p}(R|F_i)$. To this end, we follow the sampling procedure shown in [96]:

Step 1 Consider an observation R'_I from the set S_2 described in section 5.3.3 with I uniformly chosen from $\{1, \cdots, 2n\}$ at random.

Step 2 Generate v to have probability density function $K_e(v)$ in (4.2).

Step 3 Set $R'_s = R'_I + h\lambda'_I v$, where λ'_I is computed using (4.4). If $R'_s < 0$, set $R'_s = -R'_s$.

The acceptance-rejection method [97] is used in Step 2 to sample from the $K_e(v)$. In particular, let $U(v)$ be the probability density function of the uniform distribution in

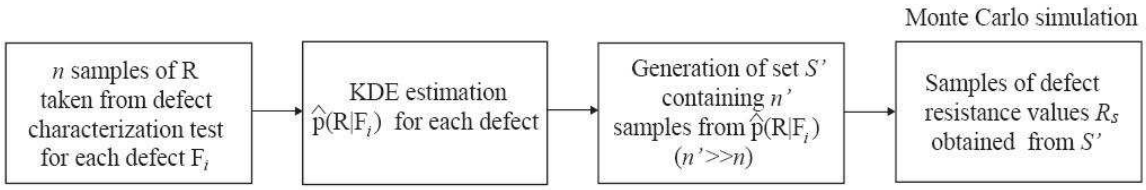


Figure 5.10: Defect resistance sampling procedure in fault simulation.

$[-1,1]^d$ and notice that $K_e(v) \leq c \cdot U(v)$, $c = c_d^{-1}(d+2)/2$, $\forall v \in R^d$. The acceptance-rejection method is as follows:

Step 2a Generate v from U.

Step 2b Generate u from a uniform distribution in [0,1].

Step 2c If $c \cdot u \leq K_e(v)$ accept and v, otherwise return to step 2a.

Steps 1-3 are repeated n' times and the obtained n' samples of R_s' constitute the set S'. Finally, during the Monte Carlo simulation, we sample the set S' uniformly at random to obtain a defect resistance value R_s.

Subsequently, for each instance, we obtain the diagnostic measurement pattern m by post-layout simulations. In this Monte Carlo approach, the parameters of the circuit are sampled from their fault-free distributions. In total, we generate N=500 defect instances corresponding to N=500 observations of pattern m. We repeat the above fault injection step for every fault location F_i, $i = 1, \cdots, 24$. Thus, in total, we simulate $24 \times 500 = 12000$ diagnostic patterns. Using these data, we perform a Principal Component Analysis (PCA), in order to map the original 205 diagnostic measurements onto vectors in a lower dimensional space with cardinality $d' < 205$. We maintained the structure of the data while keeping only 9 principal components, i.e. $d' = 9$.

5.4.4 Diagnosis phase

To evaluate the diagnosis rule established by Equation (5.5) and to examine the resulting fault ambiguities, we generate independently another set of LNA instances. This set contains 500 instances for each defect location, i.e. $24 \times 500 = 12000$ instances in total. Each instance undergoes specification-based testing according to Figure 5.2(b). Diagnosis is applied to those instances which violate one or more of the specifications that are listed in Table 5.3. The number of faulty LNA instances corresponding to each defect is shown in the third column of Table 5.4. As can be seen, open circuits always result in circuit malfunction, whereas the effect of short circuits is not always catastrophic. Such short circuits have resistance values that fall towards the tail of the distribution of Fig. 5.4(a).

Next, we carry out post-layout simulations to obtain the diagnostic measurements for each faulty LNA instance and we use the PCA transformation matrix to obtain the reduced 9-dimensional pattern. Based on the diagnostic measurement of each instance, we diagnose that fault j has occurred, according to Equation (5.5). The number of

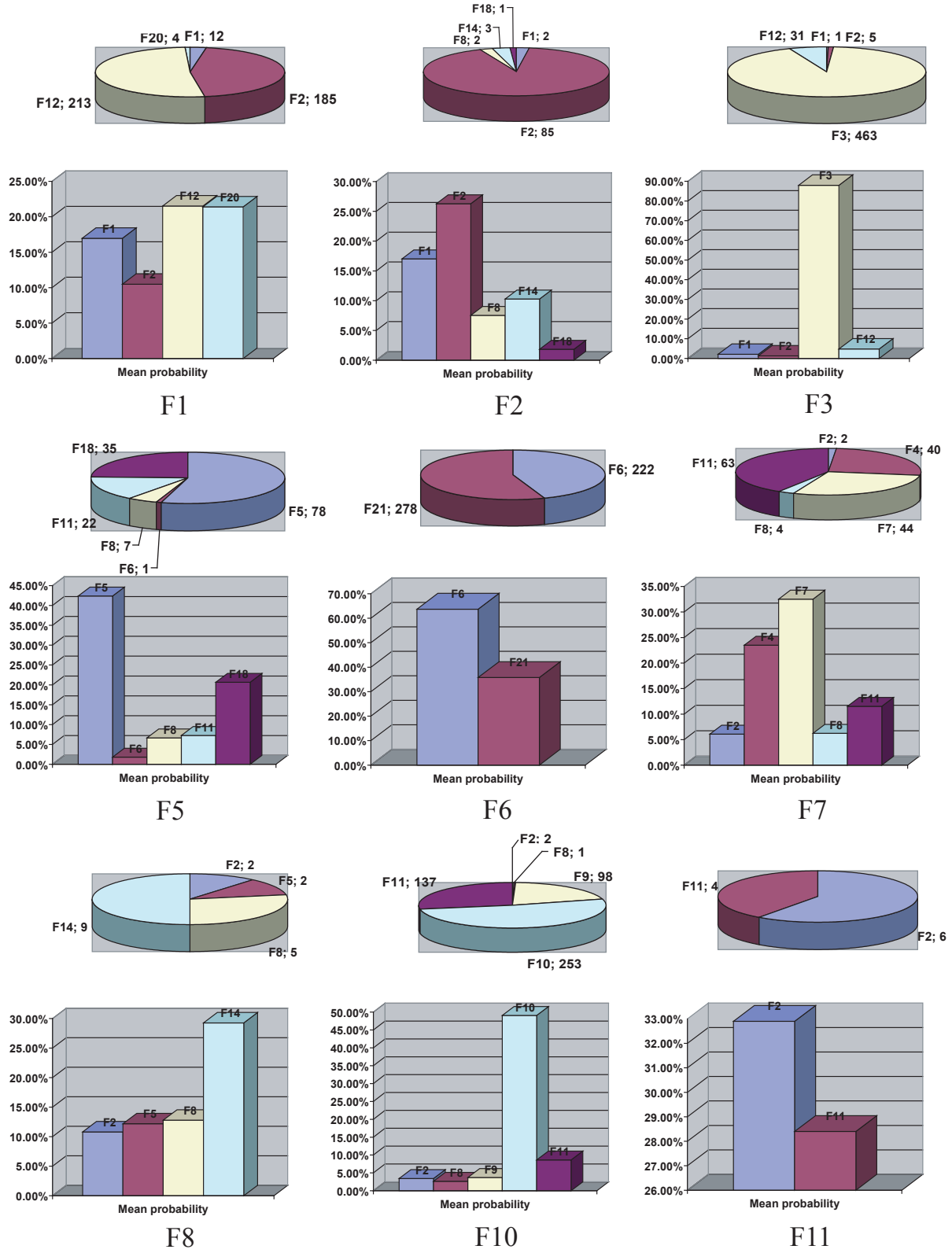


Figure 5.11: Diagnostic decision plot for cases where the diagnostic rate is less than 100%.

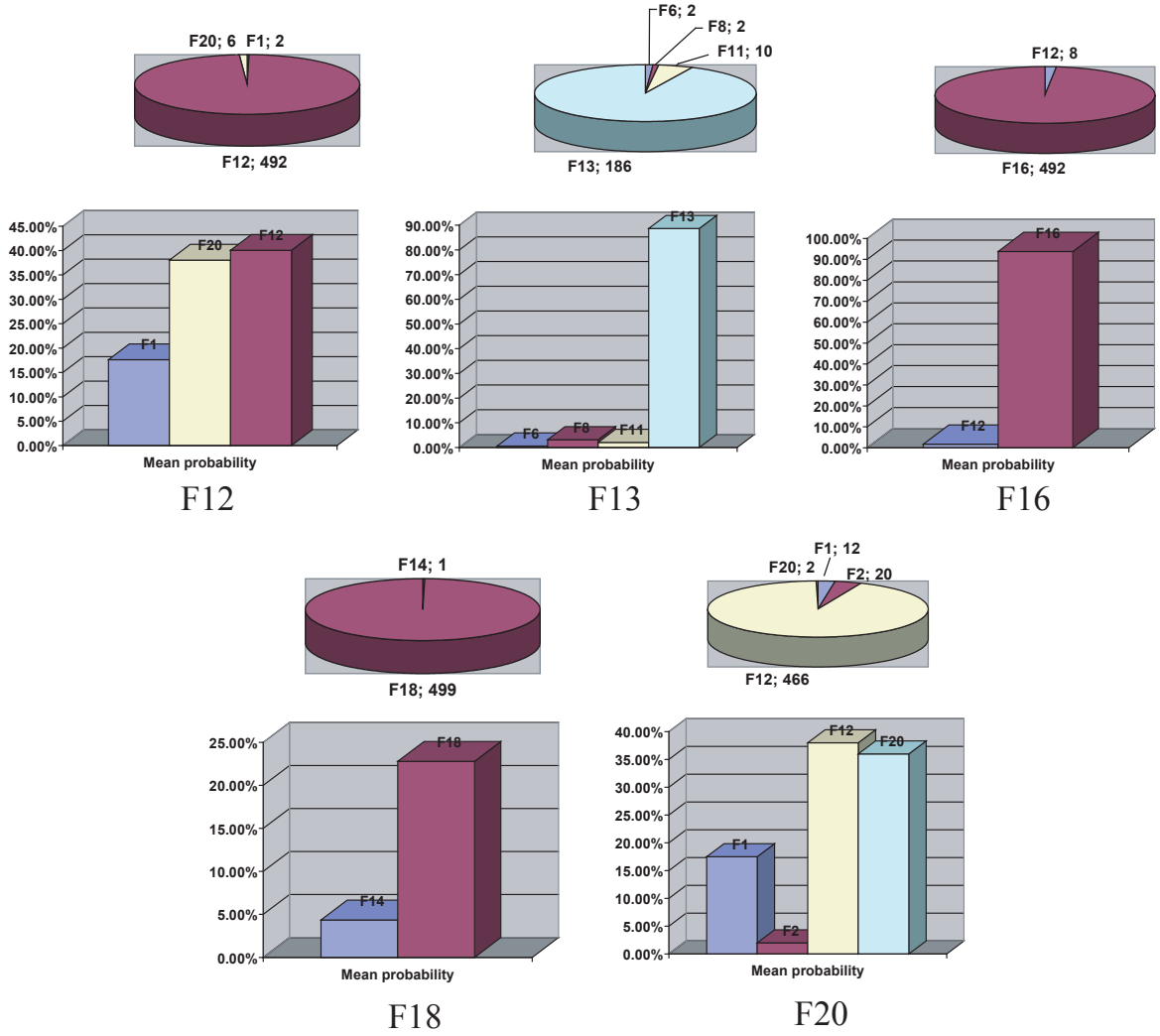


Figure 5.12: Diagnostic decision plot for cases where the diagnostic rate is less than 100% (continued).

devices correctly diagnosed, as well as the diagnostic rate for each defect, are shown in the fourth column of the Table 5.4. As can be seen, the diagnostic rate for most open defects are satisfactory. The defects that are routinely misdiagnosed are opens in the path of the biasing stage. The diagnosis results for the short defects are less satisfactory. The existence of ambiguities (i.e. defects giving similar diagnostic measurements) inevitably leads to erroneous diagnosis decisions.

Figures 5.11 and 5.12 show the diagnostic decisions for all cases where the diagnostic rate is less than 100%. For example, for the case of F_1, only 12 out of 414 instances are correctly diagnosed (e.g. a diagnostic rate of 2.9%). The rest are misdiagnosed as having defects F_2, F_{12} and F_{20}. The pie chart shows the number of instances that are misdiagnosed to each of these defect classes.

Furthermore, it is worthwhile to analyze the mean relative probability for each defect over all 500 recorded patterns m defined by

$$f(F_j) = E\left[\frac{p(m|Fj)}{\sum_{i=1}^{n} p(m|Fi)}\right] \tag{5.9}$$

The result is shown in the bar plots in Figures 5.11 and 5.12. These plots offer more insight about defect ambiguities. For example, even if finally only 2.9% of defects F_1 are correctly diagnosed, the score $f(F_1)$ is close to the scores $f(F_2)$, $f(F_{12})$, and $f(F_{20})$, which means that defect F_1 should be suspected. In all cases for which the diagnostic rate is less than 100%, the actual defect always ranks among the three most likely ones. Once the ambiguities are analyzed, we can return to the LNA schematic to understand their origin and, thereby, to enhance the set of diagnostic measurements in order to resolve these ambiguities.

5.5 Conclusion

In this chapter, we presented a fault diagnosis flow for analog circuits that relies on the Bayes rule to assign occurrence probabilities for potential defects. We model spot defects as short and open circuits, yet we study a variety of resistive and capacitive behaviors for each defect location. Furthermore, we generalize our approach by modeling the various probability densities in the analysis, i.e. the likelihoods in the Bayes rule and the defect distributions, using nonparametric kernel density estimation. The proposed defect diagnosis flow is demonstrated on an RF LNA using post-layout simulations.

Chapter 6

Experimental results

6.1 Introduction

In this chapter, we present the experimental results of diagnosis approaches presented in Chapter 4 and 5. The case study is an industrial, large-scale device designed by NXP semiconductors and it is produced in high-volume. We focus on diagnosis of spot defects, in particular short circuits since they are considered to be the most common defects for this case study [98]. Diagnosis of faulty devices has been already carried out by NXP using traditional failure analysis (FA) methods by observing failures by their optical characteristics. However, as discussed in Chapter 1, these tedious methods are inadequate given the high complexity of this case study. Thus, developing low-cost test based diagnosis approaches in order to determine the root cause of failure or to guide appropriately the aforementioned classical FA methods and reduce the required time-to-diagnose is crucial to expand safety features.

For this real case study with an industrial device, we have encountered the problem of missing values in fault simulation and DUTs. This problem obliges us to apply missing data analysis and subsequently discard simulated defects or diagnostic measurements if necessary. More detail about missing value analysis will be shown in Section 6.2.2.

Furthermore, the diagnosis approaches presented in Chapter 4 and 5 require statistical fault simulation to obtain enough samples of modelled defects in order to train diagnosis tools. However, statistical fault simulation is impractical for our case study since the time needed for carrying out this simulation is intolerable given the complexity of the device. With insufficient simulation samples of defects, diagnosis result could be misleading. In order to enhance diagnosis, we propose to use multiple classifiers and combine their scores, rather than using a single classifier as shown in Chapter 4 and 5. Finally, experimental results show that combination of classifiers can efficiently improve diagnosis.

6.2 Proposed approach

Diagnosis of failed parts is very important for the case study since it is used in

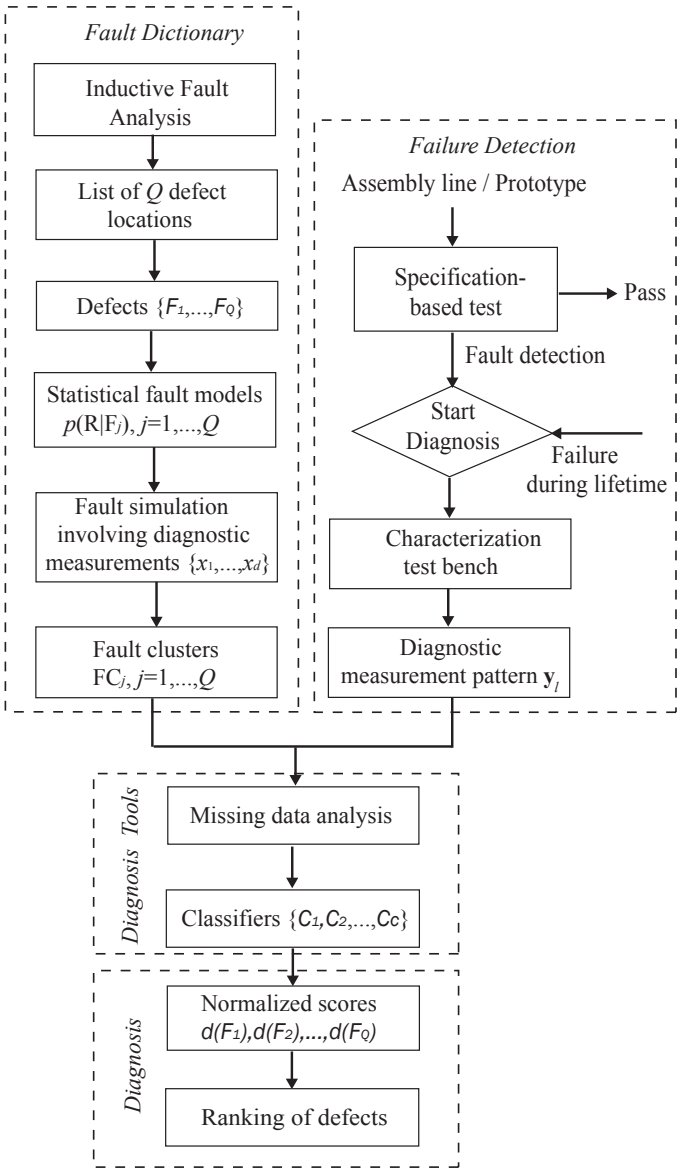


Figure 6.1: Proposed fault diagnosis flow.

automobile systems. As discussed in Chapter 4 and 5, diagnosis of local spot defects in analog circuits can be viewed as a probabilistic pattern recognition task. As presented in Chapter 5, the flow starts by examining possible defect scenarios through an IFA which results in a list of probable defects. Based on the diagnostic measurement pattern of the DUT, these defects are ranked according to their probability of occurrence. As discussed before, the reality of this real, large-scale study has forced us to study the problem of missing values in the simulation data and in the diagnostic measurement pattern of the DUT. Finally, the scores from different classifiers are combined to obtain an average score for each defect. One can consider the ranking to guide a classical FA to identify much faster the true defect.

Fig. 6.1 shows a high level description of the proposed flow. We have added missing data analysis and classifier combination in our flow compared to that presented in Figure

5.2. These additional analyses, which are not considered in Chapter 5, are necessary for diagnosing a real case study. As before, the first step takes place off-line and involves fault simulation to construct the fault dictionary. In particular, the list of Q probable defect locations is generated through an IFA. This list is believed to represent the totality of the defects that may occur in practice. A defect F_j, $j = 1, \cdots, Q$, is modeled by either a short-circuit or an open-circuit that has a certain resistance value R. The resistance can take values according to a distribution $p(R|Fj)$ that is fitted based on characterization data as shown in Chapter 5.

At this point we choose the diagnostic measurements that we will employ in the diagnosis analysis. Given the list of probable defects F_j, the densities $p(R|Fj)$, and a set of d diagnostic measurements, we perform fault simulation in order to construct the fault dictionary. Formally, let

$$\mathbf{x}_i^j = \left[x_{i,1}^j, x_{i,2}^j, \cdots, x_{i,d}^j\right] \tag{6.1}$$

denote the simulated diagnostic measurement pattern for the j-th defect that has a resistance value R_i sampled from $p(R|Fj)$. With this notation, $x_{i,k}^j$ denotes the k-th diagnostic measurement for the j-th defect that has resistance value R_i. For n resistance values, we obtain the j-th fault cluster

$$FC_j = \left\{\mathbf{x}_1^j, \cdots, \mathbf{x}_n^j\right\}. \tag{6.2}$$

In other words, the j-th fault cluster consists of n points allocated in the space of diagnostic measurements, where each point corresponds to the diagnostic measurement pattern of the j-th defect for a specific resistance value. If the diagnostic measurement pattern is sensitive to the resistance value, then the j-th fault cluster will be sparse and may overlap with other fault clusters, thus resulting in defect ambiguity. We can first use the standard tests that are performed on a characterization bench as diagnostic measurements and add more measurements if necessary to resolve defect ambiguity. It is also possible to enhance each fault cluster with more points that represent process spread. This is recommended if we can afford the extra simulation effort. In particular, for each resistance value, we can perform n' Monte Carlo simulations by allowing the circuit parameters to vary according to their fault-free distributions in the process design kit. In this case, each fault cluster consists of $n \cdot n'$ points. The fault clusters FC_j, $j = 1, \cdots, Q$, compose the fault dictionary.

The fault dictionary is put aside so that it can be readily used for diagnosing a faulty device. In particular, any prototype or any device in the assembly line that have been detected to violate one or more specifications, as well as any device that has failed in the field of operation and is a customer return, are forwarded to the diagnosis phase. To perform the diagnosis, we obtain the same d-dimensional diagnostic measurement pattern defined in the first preparatory step. The diagnostic measurement pattern of the real l-th faulty device is denoted by

$$\mathbf{y}_l = [y_{l,1}, y_{l,2}, \cdots, y_{l,d}]. \tag{6.3}$$

The diagnosis phase consists of constructing the diagnosis tools and subsequently using them to perform diagnosis of the faulty device. To construct the diagnosis tools, we need to first deal with the problem of missing data in the vectors FC_j, $j = 1, \cdots, Q$,

and $\mathbf{y}_l$. Specifically, fault simulation for some diagnostic measurements might not converge or it might result in untrustworthy values that do not comply with the range of values expected to be seen in practice. If this scenario occurs for the k-th diagnostic measurement of the j-th defect with resistance value R_i, then the value $x_{i,k}^j$ is considered to be missing. Similarly, if a diagnostic measurement $y_{l,k}$ on a real device hits the instrument limits, then it is considered to be missing. In this step, the vectors FC_j, $j = 1, \cdots, Q$, and $\mathbf{y}_l$ are cleaned up from the missing values. The missing value analysis will be discussed in detail in Sections 6.2.2 and 6.2.5.

The diagnosis tools include a set of c classifiers $\{C_1, C_2, \cdots, C_c\}$ that are trained using the fault dictionary. The selected classifiers are described in detail in Section 6.2.3. Based on the pattern $\mathbf{y}_l$ of the faulty device, each classifier assigns a probability score to each of the modeled defects instead of just making a deterministic judgement about which one of the defects is present in the faulty device. Furthermore, the classifiers are combined to assign a single probability score $d(F_j)$ to each of the defects. In practice, this has been shown to improve the classification accuracy [99, 100]. The combination method is discussed in Section 6.2.4. The output of the diagnosis phase is the ranking of the defects according to their probability of occurrence in the faulty device. This information can be used to guide the tedious search in the traditional FA flow to identify faster the defect that has occurred.

6.2.1 Normalization

Two different diagnostic measurements can take ranges of values that differ by many orders of magnitude. On the other hand, a diagnosis tool always involves the notion of distance between the pattern $\mathbf{y}_l$ of the faulty device and the fault clusters FC_j. Therefore, we need to normalize the diagnostic measurements to have similar mean and variance, such that we avoid having the distance measure being dominated by a few diagnostic measurements while being practically insensitive to variations in the rest of the diagnostic measurements. In this work, we have chosen to scale each diagnostic measurement in the range [-1,1]. In particular, the lower and upper specification limits of the diagnostic measurements are mapped to -1 and 1, respectively. In the rest of this chapter, we keep the notation of Section 6.2, however the reader should be aware that the diagnostic measurement pattern is assumed to be normalized.

6.2.2 Missing value analysis

The injection of a defect in the device netlist might render the system of equations during circuit simulation unsolvable. Therefore, it is highly likely that there exist diagnostic measurements that are unattainable for specific defects and specific resistance values. In other words, it is highly likely that there are missing values in the fault clusters FC_j due to convergence problems in circuit simulation. Furthermore, there might exist diagnostic measurements for which simulation is inaccurate since the test environment has not been modelled appropriately. This may result in large deviations between fault

simulation and test measurements. These values are also labeled as missing in the fault clusters FC_j.

The problem of missing values also concerns the real diagnostic measurement pattern $\mathbf{y}_l$. Indeed, a diagnostic measurement might hit the instrument limit, in which case its value is artificially "forced" to equal the instrument limit. In this case, we can only use the pass/fail information provided by the diagnostic measurement and we should consider the absolute value as missing.

Let z_k denote a value of the k-th diagnostic measurement. According to the notation in Section 6.2, $z_k = \{x^j_{i,k}, y_{l,k}\}^{j=1,\cdots,Q}_{i=1,\cdots,n}$. In this work, we apply the Not Missing At Random (NMAR) mechanism [101] which states that z_k is considered to be missing if $|z_k| > n_{th}$, where n_{th} is a threshold value. Notice that the fact that each diagnostic measurement is scaled in the range [-1,1] allows us to use a single threshold n_{th}. The definition of the value of n_{th} is not a simple task due to the discrepancy between the simulation environment and the characterization test bench. One can choose to incorporate the load board configuration, the test hardware, the test instrument limits, etc., in the simulation environment [102, 103], but this is time consuming given the complexity of the characterization measurements, if at all possible. For this purpose, we follow the suggestion in [101] and we consider a variety of missing models, that is, many different values of n_{th} are tested. We will revisit this issue in Section 6.2.5.

The proposed approach to account for the missing data is as follows:

1. If $y_{l,k}$ is missing, then the k-th diagnostic measurement is excluded from the analysis.

2. If $x^j_{i,k}$ is missing but the same element is available for other resistance values of the j-th defect, then $x^j_{i,k}$ is replaced by the mean value of the available elements. This approach is called mean imputation [101]. For example, if $x^j_{h,k}$ is available for $h = 1, \cdots, i-1, i+1, \cdots, n$, then $x^j_{i,k}$ is replaced by $\frac{1}{n-1}\sum_{h\neq i} x^j_{h,k}$.

3. Let

$$A^j = \begin{bmatrix} \mathbf{x}^j_1 \\ \mathbf{x}^j_2 \\ \vdots \\ \mathbf{x}^j_n \end{bmatrix} \tag{6.4}$$

denote the matrix corresponding to the j-th fault cluster FC_j and let

$$A = \begin{bmatrix} A^1 \\ A^2 \\ \vdots \\ A^Q \end{bmatrix}. \tag{6.5}$$

The matrix A is scanned and each time an element $x^j_{i,k}$ is found to be missing and it cannot be replaced using mean imputation in step 2), then either the j-th defect or the k-th diagnostic measurement is excluded from the analysis. This

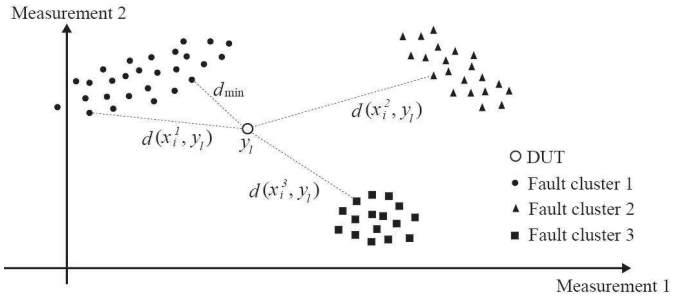


Figure 6.2: Euclidean distance method in a 2-dimensional diagnostic measurement space.

approach is called listwise deletion [101]. To decide whether to exclude the defect or the diagnostic measurement we count the number of defects for which the k-th diagnostic measurement is missing, denoted by N_{def}^k, as well as the number of diagnostic measurements that are missing for the j-th defect, denoted by N_{meas}^j. If

$$\frac{N_{def}^k}{Q} > \beta \times \frac{N_{meas}^j}{d}, \tag{6.6}$$

where β is a user-defined coefficient, then we exclude the k-th diagnostic measurement, otherwise we exclude the j-th defect. Setting β small, more diagnostic measurements will be excluded, whereas, setting β large, more defects will be excluded.

To conclude, missing values force us to exclude either diagnostic measurements or defects from the analysis. In the former case, we remove information that may be useful for performing diagnosis. In the latter case, we are bound to obtain misleading diagnosis results if the defect that is present in the faulty device is inadvertently excluded from the analysis.

6.2.3 Classification methods

As already mentioned, numerous classifiers, ranging from simple to more elaborate ones, can be employed to diagnose local spot defects. In this section, we describe in detail the classifiers that we use in this work and we show how they assign to each defect a normalized score between [0,1]. In Section 6.2.4, the normalized scores are combined to obtain a unified approach that improves the diagnosis accuracy as opposed to using a single classifier.

Euclidean distance

As shown in Fig. 6.2, this method relies on the distances between the patterns $\mathbf{y}_l$ and $\mathbf{x}_i^j$, $i = 1, \cdots, n$, $j = 1, \cdots, Q$. We consider the Euclidean distance to determine pattern proximity

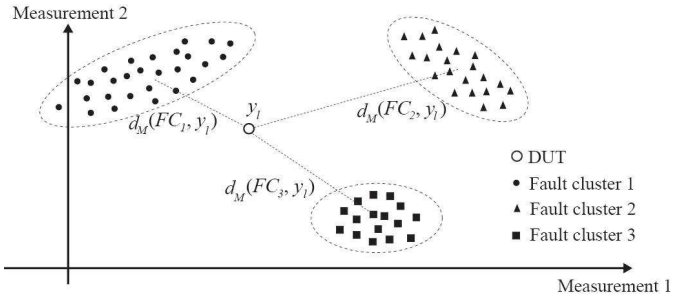


Figure 6.3: Mahalanobis distance method in a 2-dimensional diagnostic measurement space.

$$d(\mathbf{x}_i^j, \mathbf{y}_l) = \sqrt{(x_{i,1}^j - y_{l,1})^2 + \cdots + (x_{i,d}^j - y_{l,d})^2}. \tag{6.7}$$

We define the minimum distance as

$$d_{min} = \min_{i,j} d(\mathbf{x}_i^j, \mathbf{y}_l) \tag{6.8}$$

which allows us to scale the distances between [0,1]

$$d^{'}(\mathbf{x}_i^j, \mathbf{y}_l) = d_{min}/d(\mathbf{x}_i^j, \mathbf{y}_l). \tag{6.9}$$

The pattern $\mathbf{x}_i^j$ with the shortest distance from the pattern $\mathbf{y}_l$ is mapped to 1. We assign a score to each defect F_j by computing the average normalized distance $d^{'}(\mathbf{x}_i^j, \mathbf{y}_l)$ over all resistance values $i = 1, \cdots, n$

$$d_1(F_j) = \frac{1}{n} \sum_{i=1}^{n} d^{'}(\mathbf{x}_i^j, \mathbf{y}_l). \tag{6.10}$$

Mahalanobis distance

This method considers the Mahalanobis distance between the pattern $\mathbf{y}_l$ and each fault cluster FC_j, $j = 1, \cdots, Q$. As shown in Fig. 6.3, this form of distance represents the difference between the pattern $\mathbf{y}_l$ and the mean of the fault cluster FC_j, normalized by the within-cluster covariance which is a measure of the spread of the cluster around the center of its mass

$$d_M(FC_j, \mathbf{y}_l) = \sqrt{(\mathbf{y}_l - \mathbf{u}_j)^T \times S_j^{-1} \times (\mathbf{y}_l - \mathbf{u}_j)}, \tag{6.11}$$

where $\mathbf{u}_j = [u_{j,1}, \cdots, u_{j,d}]$ is the mean vector with

$$u_{j,k} = \sum_{i=1}^{n} x_{i,k}^j, \tag{6.12}$$

S_j is the covariance matrix shown in (6.13), and $E[\cdot]$ denotes the expected value computed over all resistance values $i = 1, \cdots, n$. This method favors fault clusters for which the

$$S_j = \begin{pmatrix} E[(x_{i,1}^j - u_{j,1})(x_{i,1}^j - u_{j,1})] & \cdots & E[(x_{i,1}^j - u_{j,1})(x_{i,d}^j - u_{j,d})] \\ E[(x_{i,2}^j - u_{j,2})(x_{i,1}^j - u_{j,1})] & \cdots & E[(x_{i,2}^j - u_{j,2})(x_{i,d}^j - u_{j,d})] \\ \vdots & \cdots & \vdots \\ E[(x_{i,d}^j - u_{j,d})(x_{i,1}^j - u_{j,1})] & \cdots & E[(x_{i,d}^j - u_{j,d})(x_{i,d}^j - u_{j,d})] \end{pmatrix} \tag{6.13}$$

distance between their center of mass and the pattern $\mathbf{y}_l$ is small and penalizes fault clusters for which this distance is large compared to their spread. By defining the minimum Mahalanobis distance as

$$d_{Mmin} = \min_j d_M(FC_j, \mathbf{y}_l), \tag{6.14}$$

we assign a score to each defect F_j between [0,1]

$$d_2(F_j) = d_{Mmin}/d_M(FC_j, \mathbf{y}_l), \tag{6.15}$$

where, as before, the highest score is given to the most probable defect.

Non-parametric kernel density estimation (KDE)

As already shown in Chapter 5, a faulty DUT will most likely contain defect F_m if

$$f_m(\mathbf{y}|F_m) > f_j(\mathbf{y}|F_j), \quad \forall j \neq m. \tag{6.16}$$

This method relies on the estimation of the densities $f_j(\mathbf{y}|F_j)$, $j = 1, \cdots, Q$ using the available observations $\mathbf{x}_i^j$, $i = 1, \cdots, n$, contained in the j-th fault cluster FC_j. We will re-use the KDE method presented in Chapter 5 to estimate $f_j(\mathbf{y}|F_j)$. Recall that the kernel density estimate is defined as (see section 4.2.1 of Chapter 4)

$$\hat{f}_j(\mathbf{y}|F_j) = \frac{1}{n \times h^d} \sum_{i=1}^{n} K_e(\frac{1}{h}(\mathbf{y} - \mathbf{x}_i^j)) \tag{6.17}$$

where h is a parameter called bandwidth, $K_e(t)$ is the Epanechnikov kernel

$$K_e(t) = \begin{cases} \frac{1}{2} c_d^{-1}(d+2)(1 - t^T t) & \text{if} \quad t^T t < 1 \\ 0 & \text{otherwise} \end{cases} \tag{6.18}$$

and $c_d = 2\pi^{d/2}/(d \cdot \Gamma(d/2))$ is the volume of the unit d-dimensional sphere. The kernel density estimate can be interpreted as the normalized sum of a set of identical kernels centered on the available observations, as shown in Fig. 4.2 (a) for the 1-dimensional case. The bandwidth h corresponds to the distance between the center of the kernel and the kernel edge where the kernel density becomes zero.

We use an adaptive version of (6.17). In particular, we allow the bandwidth h to vary from one observation $\mathbf{x}_i^j$ to another, allowing larger bandwidths for the observations that

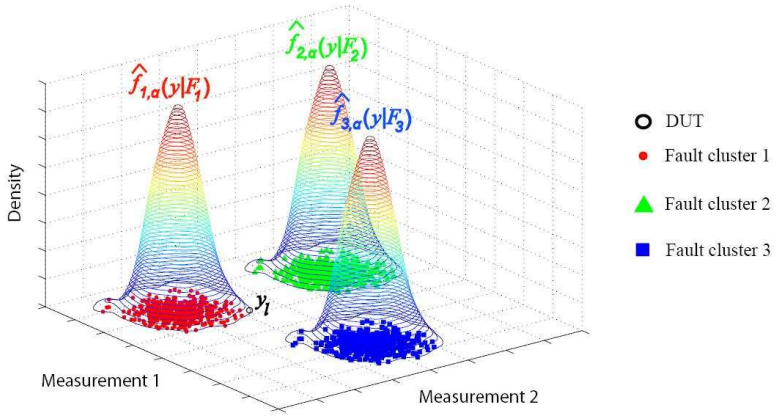


Figure 6.4: KDE method in a 2-dimensional diagnostic measurement space.

lie at the tails of the distribution, as shown in Fig. 4.2(b). The adaptive kernel density estimate is defined as [92]

$$\hat{f}_{j,\alpha}(\mathbf{y}|F_j) = \frac{1}{n}\sum_{i=1}^{n} \frac{1}{(h\cdot\lambda_i)^d} K_e(\frac{1}{h\cdot\lambda_i}(\mathbf{y}-\mathbf{x}_i^j)) \tag{6.19}$$

where the local bandwidth factors λ_i are defined as

$$\lambda_i = \{\hat{f}_j(\mathbf{x}_i^j|F_j)/g\}^{-\alpha}, \tag{6.20}$$

$\hat{f}_j(\mathbf{x}_i^j|F_j)$ is the pilot density estimate given in (6.17), g is the geometric mean

$$\log g = n^{-1}\sum_{i=1}^{n} \log \hat{f}_j(\mathbf{x}_i^j|F_j) \tag{6.21}$$

and α is a parameter which controls the local bandwidths. The larger α is, the larger will be the diagnostic measurement space where the density $\hat{f}_{j,\alpha}(\mathbf{y}|F_j)$ is nonzero. An example of densities $\hat{f}_{j,\alpha}(\mathbf{y}|F_j)$ for three defects in a 2-dimensional diagnostic measurement space is shown in Figure 6.4.

Given a DUT with pattern $\mathbf{y}_l$, we assign a normalized score between [0,1] to each defect

$$d_3(F_j) = \frac{\hat{f}_{j,\alpha}(\mathbf{y}_l|F_j) - \hat{f}_{min}}{\hat{f}_{max} - \hat{f}_{min}}, \tag{6.22}$$

where

$$\hat{f}_{min} = \min_j \hat{f}_{j,\alpha}(\mathbf{y}_l|F_j) \tag{6.23}$$

$$\hat{f}_{max} = \max_j \hat{f}_{j,\alpha}(\mathbf{y}_l|F_j). \tag{6.24}$$

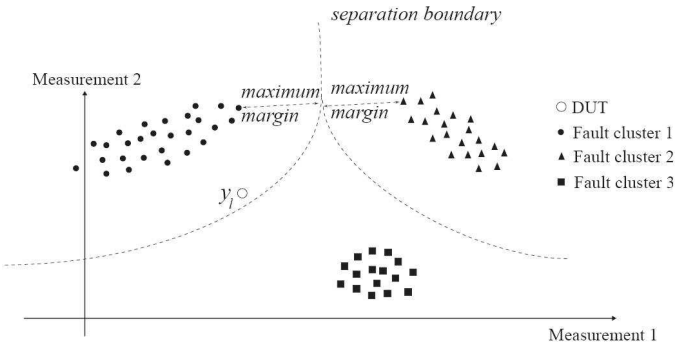


Figure 6.5: SVM method in a 2-dimensional diagnostic measurement space.

As before, the defect that achieves the highest density $\hat{f}_{j,\alpha}(\mathbf{y}_l|F_j)$ is mapped to 1. Furthermore, if $d_3(F_j)$ is zero for every defect, then the pattern $\mathbf{y}_l$ is considered to be "foreign" to all fault clusters. In this case, we can conclude that the defect that has occurred had not been modeled in the fault dictionary. Thus, unlike the other methods that always assign a score to each defect, the non-parametric KDE method is the only one that in theory can identify an "unexpected" defect. This is a very important attribute of the KDE method.

Support vector machine (SVM)

This method aims to allocate nonlinear boundaries in the space of diagnostic measurements to separate the Q fault clusters. In particular, we use SVMs [70] to learn the boundaries that traverse the middle of the Euclidean distance between the Q fault clusters. This is shown in Fig. 6.5 for a 2-dimensional diagnostic measurement space.

The SVM classifier was originally used to solve binary classification problems. For multi-class classification with Q fault clusters ($Q > 2$), we can reduce the problem into either $\binom{Q}{2}$ or Q distinct binary classification problems and apply either the "one-against-one" or the "one-against-all" strategies. Experiments on large problems show that the "one-against-one" strategy is more suitable for practical use [68]. In this approach, the classification is carried out by a max-wins voting strategy, where each binary classifier assigns the DUT to one of two fault clusters, then the vote for the assigned fault cluster is increased by one vote, and finally the fault cluster with the largest number of votes determines the fault cluster to which the DUT belongs to.

This method assigns normalized scores between [0,1] to each defect according to

$$d_4(F_j) = N_j/N_{max}, \tag{6.25}$$

where N_j denotes the number of classifiers that assign the pattern $\mathbf{y}_l$ to defect F_j and

$$N_{max} = \max_j N_j. \tag{6.26}$$

Pass/fail verification method

This method simply examines the similarity of the patterns $\mathbf{y}_l$ and $\mathbf{x}_i^j$ by verifying the pass/fail information for each diagnostic measurement. Formally, we consider the specification indicator $I_{i,k}^j$, such that (a) $I_{i,k}^j = 1$ if both $\mathbf{y}_l$ and $\mathbf{x}_i^j$ comply with the specification of the k-th diagnostic measurement or if both $\mathbf{y}_l$ and $\mathbf{x}_i^j$ fail the specification of the k-th diagnostic measurement and (b) $I_{i,k}^j = 0$ if only one of $\mathbf{y}_l$ and $\mathbf{x}_i^j$ complies with the specifications of the k-th diagnostic measurement. The normalized score between [0,1] for defect F_j is defined as

$$d_5(F_j) = \frac{1}{n}\sum_{i=1}^{n}\frac{1}{d}\sum_{k=1}^{d} I_{i,k}^j. \tag{6.27}$$

6.2.4 Classifier combination

As suggested by practitioners in the field of pattern recognition [99, 100], the overall classification accuracy can be improved by combining the response of different classifiers. Various combination methods have been proposed in the literature, including averaging, weighted averaging, majority vote, fuzzy integral, etc. [99, 100]. We have chosen the averaging method by reason of its simplicity and its capacity of providing a score for all defects without any further training.

Given y_l, the score of all considered classifiers for all F_j can be organized in a matrix DP [100]

$$DP(y_l) = \begin{pmatrix} d_1(F_1) & \cdots & d_1(F_j) & \cdots & d_1(F_Q) \\ \vdots & & \vdots & & \vdots \\ d_i(F_1) & & d_i(F_j) & & d_i(F_Q) \\ \vdots & & \vdots & & \vdots \\ d_c(F_1) & \cdots & d_c(F_j) & \cdots & d_c(F_Q) \end{pmatrix} \tag{6.28}$$

where c is the number of considered classifiers, Q is the number of fault classes, and $d_i(F_j)$ is the normalized score of the i^{th} classifier for the j^{th} fault class. The score of class F_j for a total number of c classifiers is calculated as

$$d_{com}(F_j) = \frac{1}{c}\sum_{i=1}^{c} d_i(F_j). \tag{6.29}$$

Notice that for the pass/fail verification method the notion of missing values does not apply since this method considers only the pass/fail information and not the actual diagnostic measurement values. Therefore, for the pass/fail verification method all defects and diagnostic measurements are considered in the analysis. For all other methods, a defect F_j that is eliminated from the analysis due to missing values is given a zero score, that is, $d_i(F_j) = 0$ for $i = 1, \cdots, 4$.

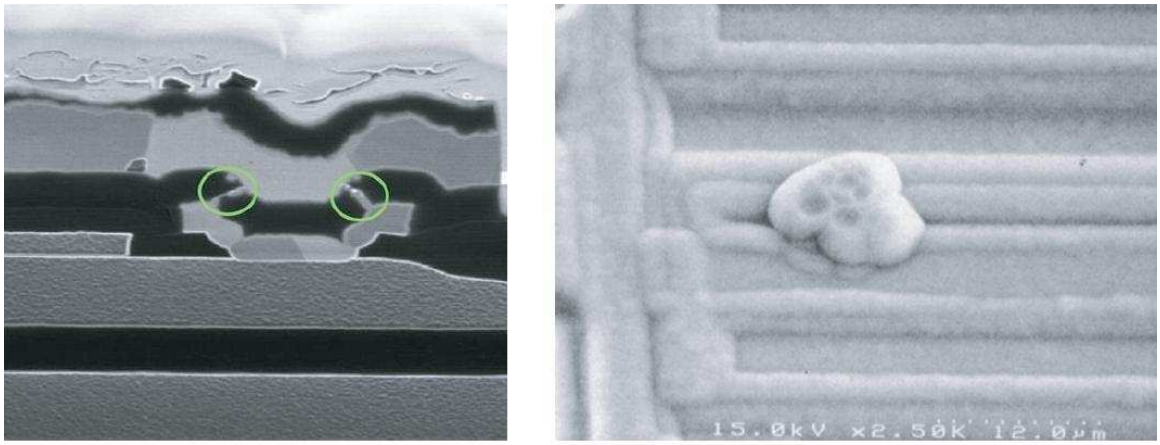


Figure 6.6: (a) FIB image of the short-circuit defect diagnosed in DUT 18 and (b) SEM image of the short-circuit defect diagnosed in DUT 26.

6.2.5 Missing model combination

As suggested in Section 6.2.2, it is more appropriate to consider several missing models in solving the NMAR problem. To this end, we consider p different values of n_{th}. The final score for defect F_j is given by

$$d_{final}(F_j) = \frac{1}{p}\sum_{i=1}^{p} d^i_{com}(F_j), \tag{6.30}$$

where $d^i_{com}(F_j)$ denotes the score for defect F_j when considering the i-th value of n_{th}, $i = 1, \cdots, p$.

6.3 Case study

6.3.1 DUT and Data Sets

Our case study is a Controller Area Network (CAN) transceiver designed by NXP Semiconductors. This device is produced in high-volume and constitutes an essential part in the electronic system of automobiles. It is deployed in a safety-critical application, thus it has to meet stringent specifications and demands practically zero test escapes. Therefore, it is of vital importance to diagnose the sources of failure, in order to achieve better quality control and, when possible, improve the design such that similar failures do not emerge in the field during the lifetime of the operation.

We have at hand a set of 29 devices from different lots that failed at least one of the specifications during production test. The classical (tedious) FA was carried out by NXP for all these devices and it was found that they contain a short-circuit (e.g. bridge) defect. For example, Fig. 6.6(a) shows a Focused Ion Beam (FIB) image of the bridge

Table 6.1: Number of deleted defects and diagnostic measurements for different values of β and n_{th}.

	$n_{th} = 50$	$n_{th} = 80$	$n_{th} = 100$
β value	Deleted defects /measurements	Deleted defects /measurements	Deleted defects /measurements
0.1	9/58	9/57	9/55
0.3	23/36	25/34	23/33
0.5	36/31	43/24	37/25
0.8	72/23	58/20	55/20
1	78/18	64/19	74/19
1.5	100/15	92/13	105/10
2	127/10	110/10	117/8

defect observed in DUT 18 and 6.6(b) shows a Scanning Electron Microscope (SEM) image of the bridge defect observed in DUT 26. For the purpose of the experiment, we assume that the actual defects that have occurred in each of these devices are unknown and we set out to diagnose them by applying the proposed flow. We consider d =97 diagnostic measurements, including DC voltage, DC current, and timing measurements.

As discussed in the introduction, short-circuit defects were considered initially to be the most common defects for this type of device [98]. Both IFA and fault simulation have been carried out by NXP. The IFA resulted in a list of Q =923 probable short-circuit defects. Subsequently, fault simulation was carried out involving the same d =97 diagnostic measurements. Each short-circuit is modeled with n =3 different resistance values (e.g. $\{5\Omega, 50\Omega, 200\Omega\}$). Thus, in total $3 \times 923 = 2769$ simulations were carried out to generate the fault clusters that we use to build the diagnosis tools.

6.3.2 Missing Values Analysis

The problem of missing values was encountered in this data set. We believe that this problem will turn up for every real, large-scale study that involves a complex device and a large set of diagnostic measurements. The number of defects and diagnostic measurements that need to be deleted from the analysis in order to account for missing values depend on (a) the coefficient β and (b) the range of thresholds n_{th} that should be considered to account for the discrepancy between the Automatic Test Equipment (ATE) and the simulation environment. The parameters β and n_{th} can be defined based on the available simulation and real data without needing to know the actual defect that has occurred. Table 6.1 shows the ratio of deleted defects and diagnostic measurements for various combinations of β and n_{th}. We have chosen $\beta = 0.3$ for the rest of analysis since the total number of deleted defects and diagnostic measurements is the minimum regardless the value of n_{th}.

6.3.3 Difficulties with classifiers

In Section 6.2.3 we described several classifiers which can be utilized for the purpose of diagnosis. However, as it will be explained in this section, the Mahalanobis distance and the SVMs turned out not to be applicable in this case study. We chose nevertheless to include them in the list of possible classifiers, in order to demonstrate that standard, popular, and well-documented approaches may not always be well-suited within the context of a real, large-scale case study.

Mahalanobis distance

The covariance matrix S_j of some fault classes is non-invertible due to the fact that (a) some diagnostic measurements are constant across all bridge resistance values and (b) there exist correlations among diagnostic measurements. If (a) we remove the constant diagnostic measurements and, thereafter, (b) we perform a Principal Component Analysis (PCA) to transform the remaining diagnostic measurement space into an orthogonal space of reduced dimensionality that nevertheless retains the variance in the data, then we end up eliminating the vast majority of diagnostic measurements, to the point where most information available for diagnosis is lost. This suggests that the Mahalanobis distance method should be abandoned for our case study.

Support Vector Machine (SVM)

The SVM classifier did not produce trustworthy diagnosis results since the training set of 3 observations for each defect (corresponding to the 3 bridge resistance values) is too small for such a high input dimensionality (e.g. 97 diagnostic measurements) and such a high number of fault clusters (e.g. 923). This method could have been useful only if the simulation effort was increased to include data for a larger number of bridge resistance values. This is not practical however when we seek to build very quickly diagnosis tools that serve to pinpoint a number of candidate defects, in order to guide appropriately the decisions in a classical failure analysis and save time.

6.3.4 Diagnosis Results

We combine 3 classifiers, namely the Euclidean distance, the non-parametric KDE with $\alpha = 50$, and the pass/fail verification method, and we obtain the normalized combined scores using (6.29). The experiment is repeated for 6 different values of n_{th} (e.g. $\{50, 60, 70, 80, 90, 100\}$) and the final averaged scores are computed using (6.30). Table 6.2 shows the 5 most highly ranked defects according to their scores for each of the 29 failed devices. The first column shows the DUT number, the second column shows the actual defect that is present, the third column shows the ranking of defects, and the fourth column shows the corresponding (rounded) final scores. Table 6.3 shows the summary of the diagnosis results for the proposed combination method. The second column shows how many times the true defect appears to be the first choice in the ranking, the

Table 6.2: Diagnosis Results.

DUT	True defect	Defect ranking	Normalized scores
1	107	**107** 90 920 114 347	0.924 0.923 0.923 0.923 0.923
2	320	**320** 341 126 374 111	0.948 0.867 0.833 0.827 0.822
3	125	47 616 **125** 681 360	0.914 0.839 0.838 0.837 0.837
4	101	**101** 117 459 50 388	0.831 0.829 0.826 0.817 0.817
5	216	**216** 666 192 516 120	0.831 0.795 0.792 0.788 0.785
6	300	524 608 744 294 789	0.900 0.890 0.862 0.855 0.850
7	20	**20** 126 24 27 111	0.889 0.866 0.862 0.850 0.849
8	27	**27** 111 126 446 341	0.891 0.856 0.837 0.834 0.834
9	104	111 **104** 465 721 126	0.848 0.844 0.839 0.823 0.822
10	21	310 682 524 789 608	0.867 0.858 0.855 0.855 0.851
11	101	**101** 117 459 50 388	0.831 0.829 0.826 0.818 0.817
12	19	**19** 541 106 562 595	0.810 0.794 0.780 0.780 0.780
13	19	**19** 541 562 595 106	0.799 0.791 0.788 0.771 0.771
14	140	401 **140** 457 40 919	0.936 0.912 0.911 0.910 0.910
15	20	**20** 24 126 27 111	0.887 0.865 0.862 0.853 0.849
16	101	**101** 117 459 50 388	0.831 0.829 0.826 0.817 0.817
17	107	**107** 90 920 114 347	0.924 0.923 0.923 0.923 0.923
18	31	117 **31** 50 388 622	0.901 0.888 0.882 0.881 0.880
19	101	252 305 366 363 31	0.883 0.857 0.846 0.844 0.843
20	19	**19** 541 106 562 595	0.821 0.794 0.793 0.780 0.780
21	156	524 608 744 789 682	0.903 0.893 0.872 0.872 0.866
22	20	**20** 126 24 27 111	0.882 0.870 0.867 0.864 0.853
23	107	**107** 90 920 114 347	0.924 0.923 0.923 0.923 0.923
24	22	**22** 19 541 338 106	0.826 0.808 0.808 0.795 0.795
25	107	**107** 90 920 114 347	0.924 0.923 0.923 0.923 0.923
26	380	666 192 516 676 457	0.910 0.906 0.905 0.904 0.903
27	376	383 456 112 34 196	0.924 0.920 0.830 0.826 0.824
28	28	666 192 516 355 676	0.910 0.907 0.898 0.896 0.896
29	300	524 608 744 475 215	0.896 0.896 0.866 0.864 0.862

Table 6.3: Comparison of diagnosis results using different classifiers as well as their combination.

Diagnosis method	First choice	First three choices	First five choices
Euclidean distance	10	11	19
Non-parametric KDE	7	7	11
Pass/fail verification	10	15	16
Combination method	17	21	21

third column shows how many times the true defect is included in the first three choices in the ranking, and the fourth column shows how many times the true defect is included in the first five choices in the ranking. As can be observed, the proposed combination method diagnoses correctly 17 out of the 29 failed devices (the true defect matches with the first choice), and includes 4 more devices in the first three choices.

Table 6.3 also shows a comparison of the proposed method with diagnosis approaches that employ a single classifier. The second, third, and fourth lines, correspond to the Euclidean distance, the non-parametric KDE, and the pass/fail verification methods, respectively. As can be observed, the combination method provides the best diagnosis result which justifies our choice to average the scores of different classifiers.

By comparing the diagnosis predictions to the true defect existing in each DUT, we identify the defects that we are unable to diagnose. We were unable to diagnose correctly defects 21, 28, 156, 300, 376, 380, and in one case defect 101. The patterns of these defects and the patterns of the defects that are diagnosed in their place turn out to overlap. We were unable to resolve this ambiguity with the available diagnostic measurements.

6.3.5 A comparison study

As discussed before, several parameters can impact diagnosis result such as missing value multiplication coefficient β, the parameter α which controls the local bandwidth in KDE method, i value in pass/fail verification method. These parameters are chosen in order to obtain an optimal diagnosis result. How the diagnosis results are affected when these parameters change? This section provides a comparison study of diagnosis results with different values of these parameters. The diagnosis result presented in this section are in essence the result in the last line of Table 6.3 with different parameter settings including the missing value multiplication coefficient β, the parameter α in KDE method, and i value in pass/fail verification method.

Diagnosis result with different β values

As presented in section 6.3.2, different fault classes and measurements are discarded using different values of β. Table 6.4 shows the diagnosis results shown in the last line of Table 6.3 with different values of β.

As can be observed in Table 6.4, $\beta = 0.3$ provides an optimal diagnosis result since the number of correct predictions is always the maximum. This observation justifies the choice for $\beta = 0.3$ when the total number of fault classes and measurements to be discarded is the minimum as shown in Table 6.1.

Diagnosis results with different values of α

As presented in section 4.2.1, α is a parameter which controls the local bandwidth of kernel density in KDE method. The larger α is, the larger will be the diagnostic

Table 6.4: Diagnosis results with different values of β.

β value	First choice	First three choices	First five choices
0.1	9	15	15
0.3	17	21	21
0.5	12	16	20
0.8	11	19	20
1	9	18	19
1.5	6	18	19
2	10	19	20

Table 6.5: Diagnosis results with different values of α.

α value	First choice	First three choices	First five choices
10	11	19	21
20	17	19	21
30	17	19	20
40	17	21	21
50	17	21	21

measurement space where the density is nonzero. Table 6.5 shows the diagnosis results in the last line of Table 6.3 with different values of α.

As can be observed in Table 6.5, diagnosis results remain very similar with different values of α. This observation shows that since the scores of all fault classes obtained using KDE method are very close to each other, they do not have a significant impact on the final scores obtained by classifier combination.

Diagnosis results with different values of i in pass/fail verification method

As presented in section 6.2.3, different values of i can be used in pass/fail verification method. Table 6.6 shows the diagnosis results in the last line of Table 6.3 with different values of i.

As can be observed in Table 6.6, diagnosis results are very similar with different values of i. This shows that the pass/fail behaviour remains the same regardless the value of defect resistance.

Table 6.6: Diagnosis results with different values of i.

i value	First choice	First three choices	First five choices
1	18	19	21
2	18	20	21
3	17	21	21

6.4 Conclusion

In this chapter, we presented the experimental results of diagnosis approaches presented in Chapter 4 and 5. The case study is an industrial, large-scale device designed by NXP semiconductors and it is produced in high-volume. We focus on diagnosis of spot defects, in particular short circuits since they are considered to be the most common defects for this case study. We have added the analysis of missing data and combination of classifiers in our diagnosis flow compared to that presented Chapter 5. These analyses are necessary for diagnosing a real case study. The combination of classifiers pinpoints to a subset of defects that are the most likely to have occurred in the DUT. The ranking of defects can be subsequently used to speed up a classical failure analysis method by placing the emphasis on the locations of the chip where the defect has probably occurred.

We showed that by combining classifiers we obtain an improved diagnosis accuracy as opposed to using a single classifier. In particular, we are able to diagnose correctly 21 out of 29 failed devices which is considered to be a successful result.

Chapter 7

Conclusions and future work

7.1 Conclusions

Fault diagnosis of ICs has grown into a special field of interest in semiconductor industry. At the design stage, diagnosing the sources of failures in IC prototypes is very critical to reduce design iterations in order to meet the time-to-market goal. In a high-volume production environment, diagnosing the sources of failures can assist the designers in gathering information regarding the underlying failure mechanisms. In cases where the IC is part of a larger system that is safety critical (e.g. automotive, aerospace), it is important to identify the root-cause of failure and apply corrective actions that will prevent failure reoccurrence and, thereby, expand the safety features.

The aim of this thesis was to develop a methodology for fault modelling and fault diagnosis of analog/mixed circuits. In general, failures in analog ICs can lead to two types of faults: catastrophic faults and parametric faults. In this thesis, a new approach has been proposed to diagnose the type of the defect (parametric or catastrophic) that is responsible for the malfunction of a circuit, localize it on the die, and identify its value.

The principal contributions of this thesis are:

1. Development of a unified catastrophic/parametric diagnosis approach using machine learning. The central learning machine is a defect filter that distinguishes failing devices due to gross defects (catastrophic faults) from failing devices due to excessive parametric deviations (parametric faults). Then two types of diagnosis can be carried out according to the decision of the defect filter. One one hand, catastrophic faults are diagnosed using a multi-class classifier. On the other hand, parametric faults are diagnosed using inverse regression functions to predict simultaneously a set of predefined design and transistor-level parameters, in order to locate the faulty parameter and identify its value.

2. Realistic fault models have been used for the purpose of diagnosis. A list of spot defect locations has been obtained according to the layout topology of the device through an Inductive Fault Analysis (IFA). The resistive behavior of spot defects has also been taken into account for constructing fault models using non-parametric density estimation.

3. Use of probabilistic method to diagnose the most probable defect that gave rise to a faulty DUT. We assign occurrence probabilities for potential defects for a faulty DUT by pinpointing to a subset of defects that are the most likely to have occurred. Deriving occurrence probabilities also allows us to analyze the misdiagnosed circuits and the resulting ambiguous groups. This is not possible using the standard fault dictionary approach since it provides a deterministic diagnosis decision.

4. The diagnosis problem has been discussed by taking into consideration the realities of an industrial, large-scale case study. The methodology has been demonstrated on data provided by NXP Semiconductors in order to determine the root cause of failure or to guide appropriately the classical Failure Analysis (FA) methods and reduce the required time-to-diagnose. The device under consideration is a Controller Area Network used in automobile systems which demands high quality control due to the reliability requirements of the application wherein it is deployed.

7.2 Future work

In term of future work, we are planning the following:

1. Construction of more accurate fault models. The characteristics of defects can differ from one technology to another. For one specific technology, new defect characterization analysis must be carried to build the appropriate fault models and avoid unanticipated faults in diagnosis phase.

2. Optimization of test stimuli to improve diagnostic accuracy and further resolve fault ambiguity. As shown in Chapter 4, some auxiliary circuit-specific test stimuli and fault diagnosis rules can be used to resolve ambiguities.

3. A more elaborated method to handle the missing values. Missing values are encountered when analyzing an industrial, large-scale case study as discussed in Chapter 6. Mean imputation method has been used in order to estimate the diagnostic measurement value that is missing. However, this method might not work well if the diagnostic measurements are very sensitive to the resistance value. The second available option is to remove the diagnostic measurement or the defect from this analysis, however, this is even less attractive. Also, the definition of missing value using a threshold value is not a simple task due to the discrepancy between the simulation environment and the characterization test bench. A more accurate missing value definition could be considered by taking into account the load board configuration, the test hardware, the test instrument limits, etc.

Chapter 8

Résumé en français

8.1 Introduction

8.1.1 Introduction

L'intégration sur une même puce des fonctions numériques, analogiques et radio fréquences (RF) est un des challenges actuels du développement des systèmes de communication du futur.

Outre la complexité de conception de tels systèmes, une attention toute particulière doit être apportée à la sûreté des circuits. Il est très important de vérifier le fonctionnement d'un circuit intégré (IC) dans la conception, durant la fabrication et lors de l'utilisation chez les clients, qui est le rôle du test.

Avec la complexité croissante des circuits intégrés, le test de circuits est devenue un défi sérieux aujourd'hui dû à une accessibilité et une observabilité limitée des blocks internes des ICs. Selon la période où le test est effectué, il peut être classé en test de caractérisation, test de production et test en fonctionnement normal (*test in field*). Le but de test de caractérisation est de vérifier lors de la phase de conception le défaut de conception, la robustesse du circuit par rapport aux variations de process. Le test de production vérifie les spécifications du circuit et détecte les défauts de fabrication qui peuvent affecter le fonctionnement des circuits. Il inclus le test fonctionnel, le test structurel et le test paramétrique. Finalement, le test en fonctionnement normal (*test in field*) permet de vérifier le fonctionnement du circuit dans son application finale. Les défauts peuvent se produire dans n'importe quel période d'une vie d'un IC, l'analyse de défauts est donc essentiel pour réduire le temps de mise sur le marché (*the time to market*) d'un circuit, améliorer le rendement et assurer la sûreté du circuit.

8.1.2 Motivation

Le diagnostic de fautes est essentiel pour atteindre l'objectif de temps avant mise sur le marché des premiers prototypes de circuits intégrés. Une autre application est dans l'environnement de production. Les informations de diagnostic sont très utiles pour les concepteurs de circuits afin d'améliorer la conception et ainsi augmenter le rendement

de production. Dans le cas où le circuit est une partie d'un système d'importance critique pour la sûreté (e.g. automobile, aérospatial), il est important que les fabricants s'engagent à identifier la source d'une défaillance dans le cas d'un retour client pour ensuite améliorer l'environnement de production afin d'éviter la récurrence d'un tel défaut et donc améliorer la sûreté.

8.1.3 Objectifs

L'objectif principal de cette thèse est de développer une approche de modélisation et de diagnostic de fautes pour les circuits analogiques/RF. En général, il existe deux types de défauts dans les circuits analogiques : fautes catastrophiques et fautes paramétriques. Les fautes catastrophiques incluent les circuit-ouverts, les court-circuits ainsi que d'autres changements topologiques dans un circuit. Les fautes paramétriques représentent les fautes qui ne changent pas la topologie du circuit et elles ont uniquement un impact sur les valeurs des paramètres. L'approche de modélisation de fautes doit prendre en comptes tous types de fautes d'une façon générale en utilisant des méthodes statistiques. Ensuite, une approche de diagnostic doit être développé pour analyser le mécanisme de défauts. Les fautes catastrophiques et paramétriques ont été traitées séparément dans la littérature, l'approche proposée dans cette thèse doit considérer tous types de défauts.

Cette thèse se déroule en collaboration avec NXP Pays-bas dans le cadre du projet européen CATRENE CT302-TOETS., l'approche de diagnostic proposée doit être validée par les données de circuits défectueux de NXP.

8.2 État de l'art sur la modélisation de fautes de circuits intégrés

8.2.1 Introduction

Dans la production, le rendement (*yield*) d'un circuit intégré représente la proportion des circuits fonctionnels est il est défini comme suit :

$$Yield = \frac{N}{M} \tag{8.1}$$

où N représente le nombre de circuits qui passent le test et M représente le nombre total de circuits fabriqués. Un défaut peut se produire dans n'importe quelle étape de production. Une connaissance profonde sur le mécanisme physique de défauts est essentielle pour construire des modèles de fautes réalistes. En plus, l'efficacité d'une approche de diagnostic est directement liée à la précision du modèle de fautes. Aujourd'hui les modèles de fautes dans les circuits numériques sont bien définis et largement utilisés dans l'outil de conception CAO [17]. Pourtant, la modélisation de fautes analogiques est encore un challenge à cause de la nature continue de l'opération de circuits analogiques, la non-linéarité, la sensibilité des performances aux variations de process, etc.

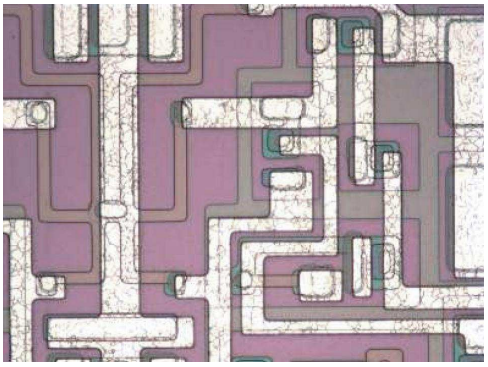


Figure 8.1: Exemple de non-alignements des masques [2].

8.2.2 Mécanismes de défauts dans les circuits analogiques intégrés

Lors de la conception d'un IC, les défauts dans les premiers prototypes peuvent être dus aux défauts de conception, l'imprécision de modèles de simulation, etc. Ce type de défauts peut être corrigé progressivement dans les itérations de conception. Dans un environnement de production, plusieurs facteurs peuvent engendrer la perte de rendement. En général, les mécanismes de défauts peuvent être classés en variations de process globales, variations de process locales, défauts *spot* (*spot defect*) et phénomène de vieillissement. Nous allons présenter par la suite les différents mécanismes de défauts.

Variations de process globales

Dans une technologie immature, les défauts peuvent être engendrés par une erreur grave dans les paramètres de contrôle, le *layout*, les équipements, etc. Les sources majeures de ces variations sont [18]:

1. Les erreurs humaines et les défaillances des équipement.

2. Instabilité dans les conditions du process, en terme de changement de valeurs de n'importe quelle paramètre physique. Par exemple, un écoulement turbulent de gaz utilisé pour la diffusion et oxydation peut engendrer des variations des paramètres de process tel que la concentration du dopage et l'épaisseur d'oxyde de grille. Ensuite, les variations de ces paramètres de process peuvent perturber les paramètres des composants tel que la tension de seuil V_{th} des transistors MOS.

3. Instabilité du matériel. Ce sont des variations de matériels dans les procédures de fabrication tel que les paramètres physiques des compositions chimiques.

4. Les non-alignements des masques. Ce sont des erreurs dans la formation de lithographie qui déforment la géométrie d'un circuit. La figure 8.1 montre un exemple de non-alignements des masques.

Il est à noter que dans la production d'un IC, des structures spéciales sont mises en place pour détecter les variations de process globales. Ces structures de test sont conçues pour avoir des performances sensibles aux paramètres de process spécifiés. Le test de ces structures est connu sous le nom de Moniteur de Contrôle de Process (*Process Control*

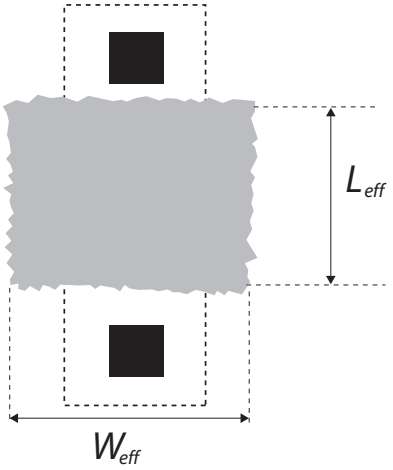


Figure 8.2: Variations locales sur L_{eff} et W_{eff}

Monitor (PCM)). Si un de ces tests PCM est échoué, le wafer sera considéré comme défectueux et il sera rejeté. Par conséquence, les variations de process globales ne sont pas considérées dans le contexte de modélisation de fautes et diagnostic.

Variations de process locales

Contrairement aux variations de process globales, les variations de process locales affectent les composants de chaque puce individuellement. En générale, ces variations peuvent perturber certains paramètres de process locaux mais elles ne changent pas la topologie du circuit. Exemples de ce type de variations sont :

1. Les déformations géométriques locales. Ce sont des effets de process qui engendrent une variation d'emplacement de frontière des différentes régions d'un IC. Les déformations géométriques peuvent être latérales ou verticales comme montrés dans [18]. Exemples de déformations latérales incluent variations de longueur effectif L_{eff} ou largeur effectif W_{eff} d'un transistor MOS [19]. La figure 8.2 montre l'impact de déformations géométriques sur L_{eff} et W_{eff}.

Comme montré dans [20], la variance de tension de seuil $\sigma^2(V_{th})$ est inversement proportionnelles au terme $L_{eff} \times W_{eff}$

$$\sigma^2(V_{th}) \propto \frac{1}{L_{eff} \times W_{eff}} \tag{8.2}$$

2. Les variations des paramètres de process locales. Exemple de ce type de variations incluent les variations sur la concentration de dopage. Ces variations peuvent être globales comme mentionné dans la section précédente, elles peuvent également être locales dû à la non- uniformité de la densité de distribution du dopage ionique [20]. Elles peuvent entraîner des variations dans la tension de seuil V_{th} des transistors MOS.

Défauts *spot*

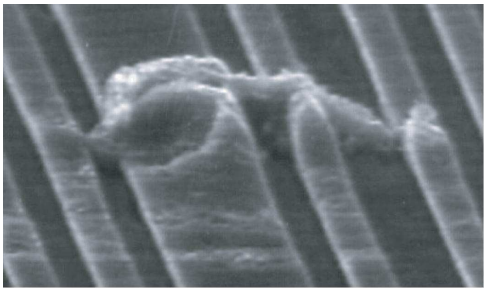


Figure 8.3: Un court-circuit entre les lignes de conduction causé par un particule [3].

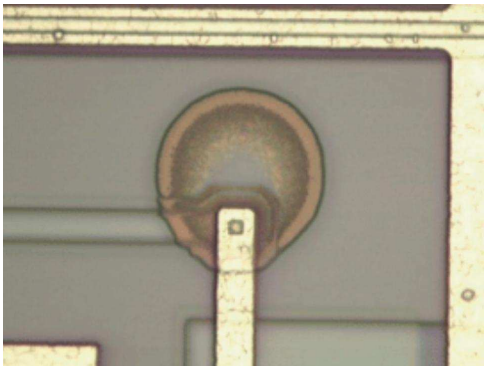


Figure 8.4: Un circuit-ouvert dans le contact causé par un résidu [3].

Les défauts *spot* sont souvent causés par des particules ou des résidus dans la fabrication et affectent soit les couches individuelles, soit les interconnections entre deux couches. Selon [10], les défauts *spot* sont des phénomènes aléatoires avec certain probabilité d'occurrence. Dans cette section, nous allons donner une description détaillée des défauts *spot*, ils peuvent être :

1. Particules, contamination dans l'environnement de la fabrication des ICs. Ils peuvent être des contaminations dans le substrat [5], des particules dans les couches de métaux (voir Figure 8.3), des résidus dans les process de fabrication (voir Figure 8.4), des poussières sur le masque, etc.

2. Défauts liés aux process de fabrication. Exemples de ce type de défauts incluent le *pinhole*, le *hillock*, le vide (*void*, voir Figure 8.5)

3. Défauts lié à la mise en boîtier du circuit. Ce type de défauts apparaît lors de la phase de mis en boîtier d'un IC. Ils peuvent être un circuit-ouvert dans un fil de connexion ou un court-circuit entre les fils, la contamination, un défaut sur le *die*, etc.

Comme les défauts *spot* entraînent un changement de la topologie du circuit, ils sont considérés comme des fautes catastrophiques. Selon plusieurs références, [25, 26, 5, 27], les défauts *spot* sont les sources majeures de défaillance dans les circuits intégrés.

Phénomène de vieillissement

Défauts peuvent également être introduits après la fabrication dans l'application finale de circuits intégrés à cause du phénomène de vieillissement. Ce type de défauts

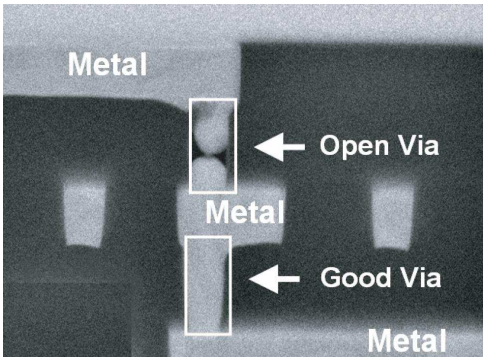


Figure 8.5: Exemple d'un circuit-ouvert sur le via causé par un vide [6].

inclut :

1. L'électromigration. Elle est définit comme le déplacement d'atomes dans un conducteur induit par un flux d'électron. L'approche conventionnelle utilisée pour assurer un degré de fiabilité suffisant reste encore actuellement basée sur le modèle empirique mis au point dans [28] :

$$\frac{1}{MTF} = AJ^2 \exp(-\frac{\varphi}{kT}) \tag{8.3}$$

où MTF représente la durée de vie moyenne avant défaillance (MTTF : *Mean Time To Failure*), A est une constante déterminée empiriquement, J est la densité de courant en Ampère par centimètre carré, φ est l'énergie d'activation, k est la constante de Boltzmann et T est la température.

2. L'instabilité de polarisation négative de température (NTBI: *Negative Bias Temperature Instability*). C'est un phénomène qui se produit dans les transistors PMOS stressés avec la tension de polarisation de grille négative à température élevée. Il peut entraîner une baisse de tension de seuil V_{th} dans les transistors PMOS.

3. L'injection de porteuses " chaudes" (HCI : *Hot Carrier Injection*). Le phénomène HCI se produit quand un électron ou un "trou" gagne suffisamment d'énergie cinétique pour être injecté du canal de conduction dans l'oxyde de grille. La présence de ces porteuses dans l'oxyde de grille durant une période prolongée peut entraîner des déviations dans le paramètres de transistors tel que la tension de seuil V_{th}.

4. Le claquage d'oxyde. L'exemple de ce type de défauts inclut La décharge électrostatique (DES). Une DES est un problème grave dans les circuits intégrés car elle peut créer un courant non négligeable dans une couche diélectrique, qui entraîne un court-circuit.

8.2.3 Modélisation de fautes

Plusieurs types de modèles de fautes sont proposés dans la littérature. Dans [32], Ils sont classés en trois catégories : modèle structurel, modèle paramétrique et modèle comportemental. Cette section montre une description détaillée des modèles de fautes.

Modèle structurel

Le modèle structurel de faute consiste à représenter un défaut qui entraîne un changement de la topologie d'un circuit. Ce modèle est largement utilisé dans les circuits numériques pour représenter des collages à 0 ou collages à 1. Un collage à 0 (1) consiste à relier un point d'un circuit à la masse (la tension d'alimentation). Il est suffisant pour représenter la plupart de fautes dans les circuits numériques.

L'avantage de la modélisation structurelle est qu'elle est simple à mettre en oeuvre. Les modèles sont souvent des composants déjà existent dans le simulateur. Cette méthode est généralement utilisée pour modéliser les fautes catastrophiques dans les circuits analogiques. Cependant, il est difficile à modéliser les fautes paramétriques avec ce modèle, car il existe un nombre infini de possibilités des déviations paramétriques.

Modèle paramétrique

La modélisation paramétrique est souvent l'attribution de la distribution d'une valeur d'un paramètre au-delà de son intervalle de tolérance. Contrairement au modèle structurel, le modèle paramétrique modélise les défauts qui ne changent pas la topologie du circuit.

Dans [41], un modèle de faute paramétrique est proposé en recherchant la déviation minimum d'un paramètre qui permet de violer au moins une des spécifications du circuit. Pour trouver une telle faute, il faut varier le paramètre en question d'un certain pourcentage jusqu'à ce qu'au moins une des spécifications soit violée tandis que les autres paramètres restent fixés à leurs valeurs nominales. Cette méthode est utilisée pour évaluer les métriques de test dans [42, 43, 44].

La modélisation paramétrique est une méthode non déterministe, elle permet de couvrir une large plage de déviations de paramètres de circuit. L'avantage est que toutes les possibilités des valeurs dans l'intervalle de variations considérées peuvent être représentées par le modèle. Mais ce modèle ne prend pas en compte la possibilité réelle de déviations de composants en assumant généralement une variation plus large que ses tolérances. Pourtant, certaines variations des paramètres assumées par le modèle se produisent rarement dans la réalité.

Modèle comportemental

Le modèle comportemental de fautes est une description de haut niveau des performances d'un circuit ou d'un sous circuit. L'injection d'une telle faute consiste à dévier les performances d'un circuit ou d'un sous circuit. Puisque les fautes sont modélisées au niveau performances, la simulation du modèle est plus rapide.

La modélisation comportementale est très utile pour un système complexe où une analyse hiérarchique est nécessaire. Dans l'industrie, les modèles comportementaux sont utilisés comme la base de développement de procédures de test [32]. Mais l'efficacité de cette méthode dépend beaucoup de la qualité du modèle, il faut un modèle très complet et précis pour pouvoir décrire le défaut physique. En plus, le modèle comportemental ne

contient pas d'informations sur les causes originales de fautes (déviation de paramètres du design ou défauts physique au niveau process), il ne permet pas d'effectuer un diagnostic profond sur les circuits défaillants. Il est souvent utilisé pour évaluer les métriques de test pour les circuits analogiques.

8.3 État de l'art sur le diagnostic de circuits analogiques

8.3.1 Introduction

Le diagnostic consiste à trouver la cause du mauvais fonctionnement d'un circuit défaillant. Selon le but du diagnostic, on peut distinguer : la détection, la localisation et l'identification de fautes. La détection de fautes consiste à détecter qu'une faute existe dans le circuit, la procédure de diagnostic s'arrête une fois l'existence d'une faute est détectée. La localisation de fautes consiste à localiser l'endroit d'une faute sur le circuit. L'identification de fautes consiste à identifier la valeur d'un paramètre (par exemple une déviation de la valeur d'un paramètre du circuit au-delà de son intervalle de tolérance) qui engendre la faute.

La technique de diagnostic peut aussi être classé selon la méthode utilisée. Généralement, il existe deux méthodes de diagnostic : simulation avant test (SBT : *Simulation Before Test*) et simulation après test (SAT : *Simulation After Test*).

8.3.2 Simulation avant test (SBT)

Dans cette approche, les simulations sont effectuées avant le test de circuits. Une fois le circuit est testé, la décision de diagnostic peut se faire rapidement. Il existe deux méthodes dans SBT : la méthode basée sur règles et la méthode de dictionnaire de fautes.

Méthode basée sur règles (*Rule-based method*)

La méthode basée sur règles représente les informations de diagnostic sous forme de règles comme "SI symptôme(s) ALORS fautes". Plusieurs centaines, voire des milliers de règles sont nécessaires pour construire la base de connaissances [66]. Dans la phase de diagnostic, le moteur d'inférence cherche dans la base de connaissances les règles appropriées pour trouver la solution du problème.

L'avantage de cette méthode est sa simplicité. Pour diagnostiquer un circuit défaillant, une fois les règles sont définies, la solution peut être obtenue rapidement. L'inconvénient de cette méthode est la difficulté d'obtenir une base de connaissances suffisante qui inclut toutes les fautes éventuelles. En plus, la construction de la base de connaissance dépend du circuit, une base de connaissance pour un circuit ne peut pas être utilisée pour un autre circuit, même un petit changement de la structure du circuit pourrait entraîner un grand changement de la base de connaissances. Cette méthode est souvent utilisée pour localiser les fautes dans les systèmes plus larges [66, 11] ou les fautes d'assemblage [67], mais elle ne peut pas diagnostiquer les fautes au niveau transistor.

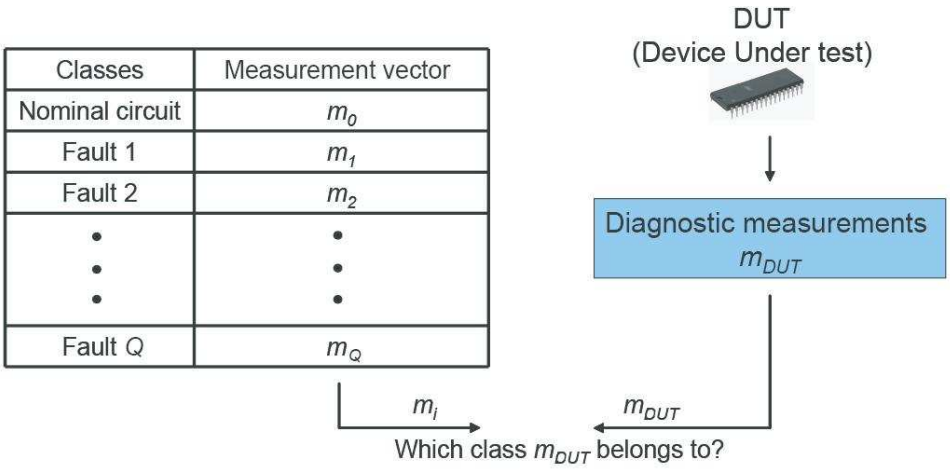


Figure 8.6: Méthode de dictionnaire de fautes

Méthode de dictionnaire de fautes

La figure 8.6 montre le principe de la méthode de dictionnaire de fautes. Cette méthode construit un dictionnaire qui contient l'ensemble de fautes $\{F_j, j = 1, 2, ..., n\}$ et de mesures de diagnostic $\{m_j, j = 1, 2, ..., n\}$ correspondantes. Ils sont obtenus à partir des simulations en générant chaque fois une faute F_j dans le *netlist* du circuit. Dans la phase de diagnostic, les mêmes mesures m_i sont prises et elles sont comparées avec celles stockées dans le dictionnaire. La faute sera celle dont les mesures sont plus similaires que celles du circuit sous test. La méthode de dictionnaire de fautes est donc une approche de reconnaissance de formes (e.g. classification). Plusieurs méthodes de classification ont été proposées dans le passé comme la recherche des plus proches voisins, les réseaux de neurones, machine à vecteurs de support (SVM), théorème de Bayes, le classificateur quadratique, etc.

8.3.3 Simulation après test (SAT)

Dans cette méthode, les simulations sont effectuées après le test du circuit. L'analyse consiste à identifier certains paramètres du circuit à partir des mesures de diagnostic. Il existe différentes méthodes de SAT pour l'identification des paramètres : technique basée sur l'analyse analytique des équations du circuit, technique basée sur l'analyse de la matrice de sensibilité, technique basée sur le modèle comportemental du circuit.

Technique basée sur l'analyse analytique des équations du circuit

Pour un circuit linéaire et invariant par décalage temporel (*linear time-invariant circuit*) ou un circuit non linéaire polarisé autour de son fonctionnement nominal, les relations entre les paramètres internes du circuit et ses performances (ou les mesures de diagnostic) peuvent être exprimées sous forme d'une série des équations non linéaires :

$$H(s_q, c) \;=\; p, \qquad q \;=\; 1, \ldots, N_f \tag{8.4}$$

où s_q est la variable *Laplace* $j\omega_q$ qui correspond aux différentes fréquences du test, N_f est le nombre de fréquences du test, c est le vecteur des paramètres du circuit à estimer et p est le vecteur des performances ou des mesures de diagnostic. Les équations (8.4) peuvent être obtenues par l'analyse analytique du circuit avec le modèle de composant de connexion (*component connection model*) [74, 76]. Résoudre l'ensemble des équations (8.4) consiste à prendre les mesures de diagnostic p' aux différentes fréquences s_q et estimer les paramètres du circuit c. Dans [74], les solutions de (8.4) n'ont pas été calculées mais la solvabilité de (8.4) a été calculée à l'aide du théorème des fonctions implicites. Les auteurs ont défini la testabilité δ comme le nombre des paramètres arbitraires dans c :

$$\delta \;=\; m - rank(\frac{dH(s_q, c)}{dc}) \tag{8.5}$$

où m est le nombre total des paramètres à résoudre. Un algorithme a été développé pour choisir un ensemble de fréquences du test s_q pour minimiser δ afin d'augmenter la solvabilité des équations (8.4). Dans [76], les auteurs ont proposé une procédure pour résoudre les équations (8.4), elle consiste à prendre les mesures p' dans le circuit sous test et estimer les paramètres c' pour minimiser $|H(s_q, c') - p'|$. Les valeurs de c' ont été obtenues par l'algorithme de *Newton-Raphson* :

$$\frac{dH(s_q, c^k)}{dc^k}(c^{k+1} - c^k) \;=\; -(H(s_q, c^k) - p') \tag{8.6}$$

où c^k est la k-ième estimation de la solution des équations (8.6) et p' représente les mesures prises dans le circuit sous test. Pour résoudre l'équation (8.6), il faut inverser $dH(s_q, c^k)/dc^k$ à chaque itération, donc $dH(s_q, c^k)/dc^k$ doit être une matrice inversible.

L'analyse analytique est une technique explicite pour estimer les paramètres du circuit c à partir des mesures de diagnostic p dont l'avantage est sa précision. Mais pour un circuit plus large, l'analyse pourrait devenir très long et complexe. La testabilité doit être aussi vérifiée. Les résultats d'analyse de testabilité montrent que ce n'est pas toujours le cas où tous les paramètres sont testables. Dans [75], les paramètres non testables sont forcés d'avoir leurs valeurs nominales et ils ne sont pas considérés dans la phase de diagnostic. En plus, la convergence de l'algorithme de *Newton-Raphson* n'est pas toujours garantie.

Technique basée sur l'analyse de la matrice de sensibilité

La matrice de sensibilité U représente le rapport entre les variations de paramètres du circuit δc et les variations de performances du circuit δp :

$$U_{m,n} = \frac{\delta p}{\delta c} = \begin{pmatrix} \frac{\delta p_1}{\delta c_1} & \cdots & \frac{\delta p_n}{\delta c_1} \\ \vdots & \ddots & \vdots \\ \frac{\delta p_1}{\delta c_m} & \cdots & \frac{\delta p_n}{\delta c_m} \end{pmatrix} \tag{8.7}$$

où m est le nombre de paramètres du circuit à identifier et n est le nombre de performances (ou de mesures de diagnostic). Pour un circuit sous test, ces mesures sont prises et elles sont comparées avec les valeurs nominales, leur différences constituent le vecteur Δp. Ensuite, les déviations de paramètres du circuit Δc sont calculées à partir de Δp et U en inversant la matrice de sensibilité U :

$$\Delta c = (U^T U)^{-1} U^T \Delta p \tag{8.8}$$

La condition pour résoudre l'équation (8.8) est que $(U^T U)^{-1}$ existe. Cela implique que le nombre de mesures n doit être supérieur ou égal au nombre de paramètres m : $n \geq m$. En plus, avec la présence d'ambiguïté de fautes, les colonnes de la matrice U ne sont pas linéairement indépendantes et U devient mal conditionnée, la solution de (8.8) n'est pas stable. Certaines méthodes pour résoudre le problème d'ambiguïté sont proposées dans [77]. Par exemple, des nouvelles mesures peuvent être rajoutées pour augmenter le rang de la matrice U. Les auteurs dans [77] ont aussi proposé un algorithme qui réduit le nombre de colonnes de U afin d'avoir une matrice de plein rang. Nous trouverons des algorithmes similaires dans [78, 46].

La matrice de sensibilité est utilisée pour estimer les variations de paramètres dans le cas de fautes paramétriques. Pourtant, cette méthode ne peut estimer que les petites variations de paramètres. Dans [79], une matrice de sensibilité incrémentale est proposée afin d'estimer les larges déviations de paramètres. L'application de cette méthode pour un circuit plus complexe est difficile. Dans [77], un algorithme itératif est proposé pour mettre à jour la matrice de sensibilité en cas de larges déviations de paramètres, mais la convergence n'est pas toujours garantie.

Technique de modèle comportemental

La technique de modèle comportemental consiste à générer un modèle approximatif du circuit. Différents niveaux d'abstraction peuvent être envisagés pour construire le modèle. Ensuite pour un circuit sous test, les mesures sont prises et comparées avec les performances du modèle. S'il existe une différence entre les performances du circuit et celles du modèle, alors la présence d'une faute est détectée. Le diagnostic consiste à ajuster les paramètres du modèle pour que ses performances soient identiques que celles du circuit sous test. Les paramètres qui ont été déviés dans le modèle indiquent l'origine de fautes.

En général, le modèle est représenté sous forme de fonction de transfert, l'identification consiste à estimer les coefficients de la fonction de transfert. Différentes méthodes d'identifications sont proposées. Dans [81], la méthode de l'estimation du maximum de vraisemblance (*maximum likelihood estimation*) est utilisée pour déterminer les coefficients de paramètres S d'un circuit multi-ports avec la présence du bruit. Dans [82],

les paramètres de petits-signaux des transistors d'un amplificateur sont estimés par un algorithme génétique a partir des mesures de paramètres S.

Théoriquement, si le modèle du circuit est précis, toutes les fautes peuvent être diagnostiquées. La difficulté principale de cette méthode est que le temps de calcul pour aboutir à une solution pourrait être très long dans la phase d'identification. En plus, si une faute a changé la topologie du circuit (e. g. une faute catastrophique), le modèle ne sera plus valable et la solution d'identification pourrait être fausse.

8.4 Diagnostic de fautes basé sur l'apprentissage automatique

Nous présentons dans cette section une méthodologie pour le diagnostic des fautes dans les circuits analogiques basée sur l'apprentissage automatique. La clé de la méthodologie proposée est un filtre de défauts qui sépare les circuits défaillants dus aux fautes catastrophiques et les circuits défaillants dus aux fautes paramétriques. Ensuite, deux types de diagnostic pourront être envisagés selon la décision du filtre de défauts : les fautes catastrophiques seront diagnostiquées en utilisant un classificateur et les fautes paramétriques seront diagnostiquées en utilisant les fonctions de régression "inverses". L'efficacité de la méthodologie proposée a été démontrée par un cas d'étude : Un amplificateur faible bruit (*low noise amplifier : LNA*)

8.4.1 Méthodologie proposée

La méthodologie du diagnostic que nous proposons est constituée par un ensemble de machines d'apprentissage automatique qui doit être entraîné dans la phase de pré-diagnostic. La figure 8.7 montre la description de la méthodologie proposée.

Le diagnostic commence par obtenir les mesures de diagnostic précisé dans la phase de pré-diagnostic. Nous pouvons considérer une partie de test de spécification au début. Si la précision de diagnostic n'est pas suffisante, le test de toutes les spécification pourrait être envisagé ou d'autre mesures pourraient être rajoutées afin de résoudre l'ambiguïté de fautes.

La clé de la méthodologie proposée est un filtre de défauts qui est entraîné dans la phase de pré-diagnostic pour séparer les circuits défaillants à cause d'une faute catastrophique et les circuits défaillants à cause d'une faute paramétrique. Donc, le filtre de défauts nous fournit une approche unifié pour diagnostiquer les fautes catastrophiques et les fautes paramétriques. Nous avons utilisé le filtre de défauts proposé récemment dans [58] dans le contexte du test alternatif. Le filtre de défauts est basé sur une estimation non-paramétrique $\tilde{f}(m)$ de la fonction de densité de probabilité jointe $f(m)$, où m est le vecteur de mesure de diagnostic. Le filtre est caractérisé par un seule paramètre α, qui est réglé dans la phase de pré-diagnostic pour contrôler l'étendu du filtre.

Si $\tilde{f}(m, \alpha) = 0$, le circuit sous test est incohérent avec la nature statistique des données utilisées pour estimer la densité, donc il est considéré comme ayant une faute catastrophique. Ensuite ce circuit sera diagnostiqué avec la méthode de dictionnaire

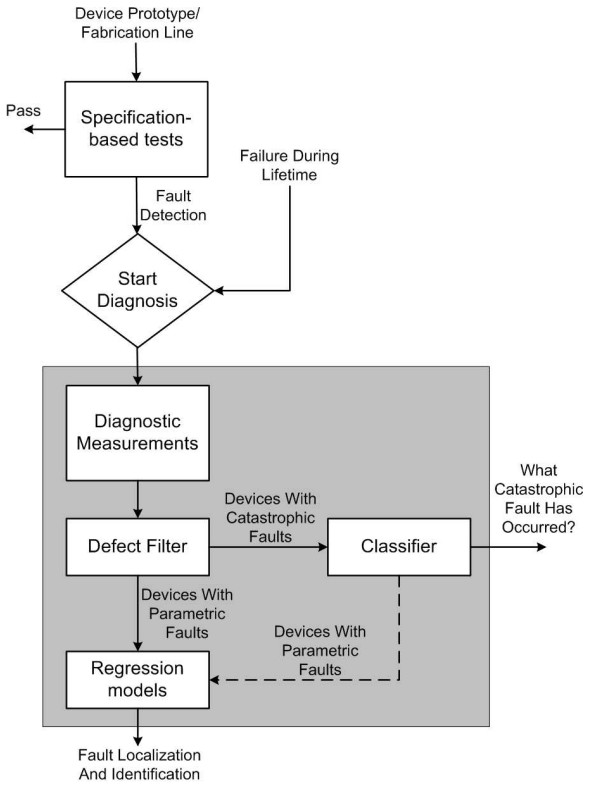


Figure 8.7: Méthodologie de diagnostic proposée

de fautes. Si $\tilde{f}(m, \alpha) > 0$, le circuit sous test est considéré d'avoir les variations de process, c'est-à-dire qu'une faute paramétrique a eu lieu. Pour diagnostiquer les fautes paramétriques, nous exprimons les relations entre le vecteur de diagnostic m et les valeurs de paramètres de circuit p_j, $j = 1, ..., n$ par n fonction de régression $f_j : m \mapsto p_j$. Cette approche nous permet de préciser implicitement la dépendance entre m et tous les paramètres p_j en utilisant les méthodes statistiques.

Le filtre de défauts est réglé pour filtrer les circuits avec fautes catastrophiques. Pourtant, certains circuits avec fautes paramétriques pourraient aussi être filtré. Pour résoudre cette fuite, le classificateur est entraîné pendant la phase de pré-diagnostic pour inclure la détection de circuits avec variations de process aussi. Donc, dans le cas où un circuit avec une faute paramétrique est présenté au classificateur, le classificateur le renvoie aux fonctions de régression.

8.4.2 Cas d'étude

Notre cas d'étude est un amplificateur faible bruit (*LNA : Low Noise Amplifier*) conçu avec la technologie 0.25 μm BiCMOS7RF de ST Microelectronics. Le schéma du circuit est montré dans la figure 8.8. Nous avons choisi les quatre paramètres S comme les mesures de diagnostic initiales. Chaque paramètre S est échantillonné de 1 GHz à 5 GHz avec un pas de 100MHz. Au total, nous avons $4 \times 41 = 164$ mesures de diagnostic.

Nous avons généré des ensembles de circuits par des simulations Monte Carlo pour

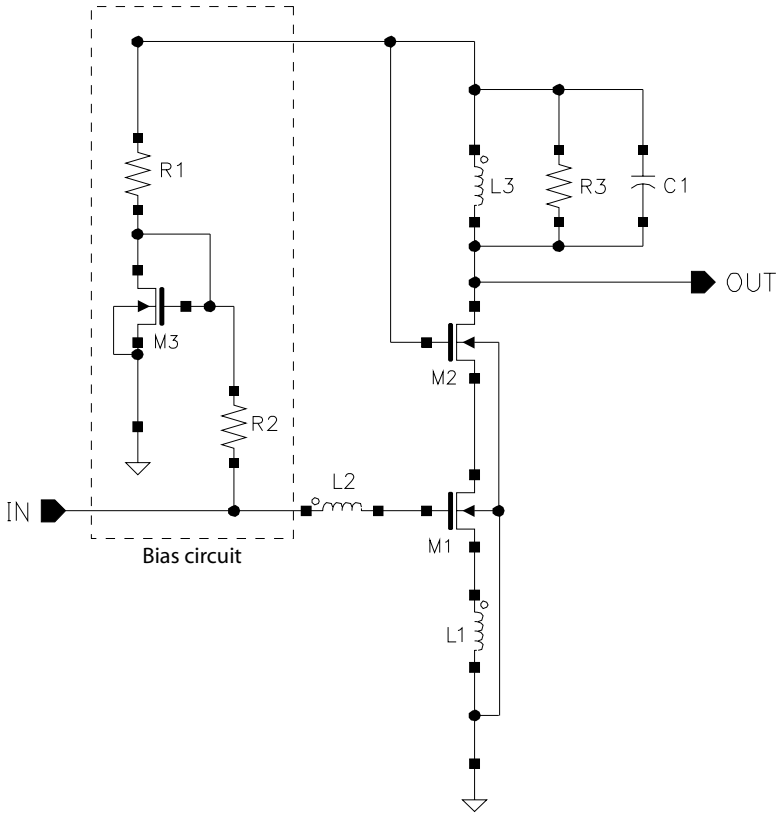


Figure 8.8: Schéma du LNA sous test

entraîner et valider les outils de diagnostic (le filtre de défauts, le classificateur et les fonctions de régression). Les circuits que nous avons générés comprennent les circuits avec fautes catastrophiques sous formes de court-circuits et circuit-ouverts ainsi que les circuits avec fautes paramétriques sous forme de déviations des paramètres du circuit (40% maximum). Après avoir construit les outils de diagnostic, nous avons vérifié l'efficacité des outils de diagnostic en injectant 1150 circuits avec fautes catastrophiques et 2000 circuits avec fautes paramétriques.

Tous ces circuits défaillants sont passés par le filtre de défauts et un seul circuit défaillant avec une faute paramétrique L2+30% est considéré ayant une faute catastrophique par le filtre. Cependant, le classificateur le classifie dans un groupe "variations de process" et le renvoie aux fonctions de régression comme montré par la flèche pointillée dans la figure 8.7. Les autres circuits avec fautes catastrophiques sont classés correctement, donc nous pouvons conclure que nous avons un taux de réussite de 100% pour le diagnostic des fautes catastrophiques. Pour le diagnostic de fautes paramétriques, l'erreur maximum de prédiction des paramètres du circuit est de moins de 3,5%. La figure 8.9 montre la projection de ces circuits sur les trois premières composantes après avoir effectué une analyse en composantes principales. Les groupes de fautes catastrophiques sont représentés par différentes couleurs et les circuits avec les variations de process sont représentés par les points noirs.

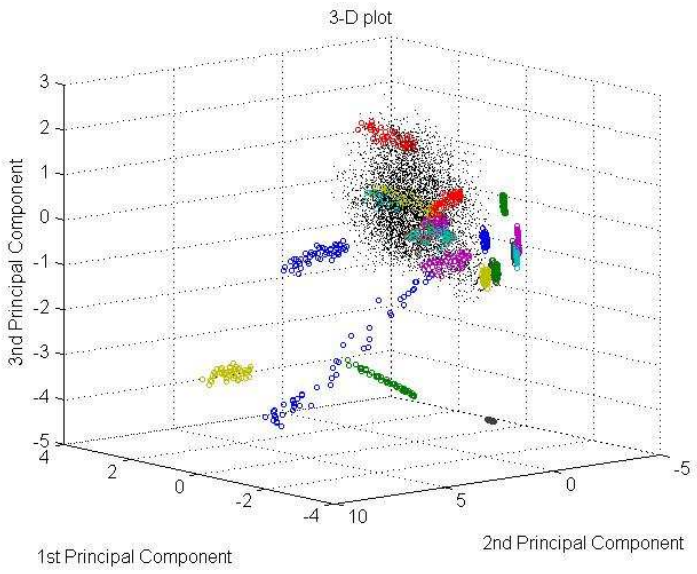


Figure 8.9: Projection de circuits entraînés dans premiers trois composantes.

8.5 Diagnostic de fautes basé sur l'estimation non -paramétrique de densité

Dans cette section, nous allons présenter une méthodologie de diagnostic de fautes pour les circuits analogiques basée sur l'estimation non paramétrique de la densité de probabilité. Nous avons utilisé un modèle de défauts qui prend en compte du comportement résistif d'un défaut. La fonction de densité de probabilité des mesures de diagnostic pour chaque défaut est estimée en utilisant une technique non paramétrique. Notre cas d'étude est le LNA montré dans la section précédente. Nous avons injecté les défauts au niveau layout et nous avons effectué des simulations post-layout pour évaluer les résultats de diagnostic.

8.5.1 Méthodologie proposée

Nous avons considéré un modèle de défauts basé sur une estimation non paramétrique de la densité de probabilité de la résistance de défaut. Nous avons choisi la méthode de l'estimation par noyau (*kernel density estimation*) à la place d'une hypothèse de distribution normale de la valeur de résistance de défaut. La méthodologie proposée est montrée dans la Figure 8.10.

Pour un circuit sous test, la fonction de vraisemblance (*likelihood function*) pour chaque défaut est d'abord estimée. Cela va nous permettre d'analyser les groupes d'ambiguïté des fautes, ce qui n'est pas possible en utilisant la méthode standard de dictionnaire de fautes. D'abord, une liste de n défauts est générée F_i, $i = 1, \cdots, n$ à partir de l'analyse de caractérisation de défauts. Ensuite nous estimons la fonction de densité de probabilité de la résistance r associé à chaque défaut. Cette densité est noté $p(R|F_i)$

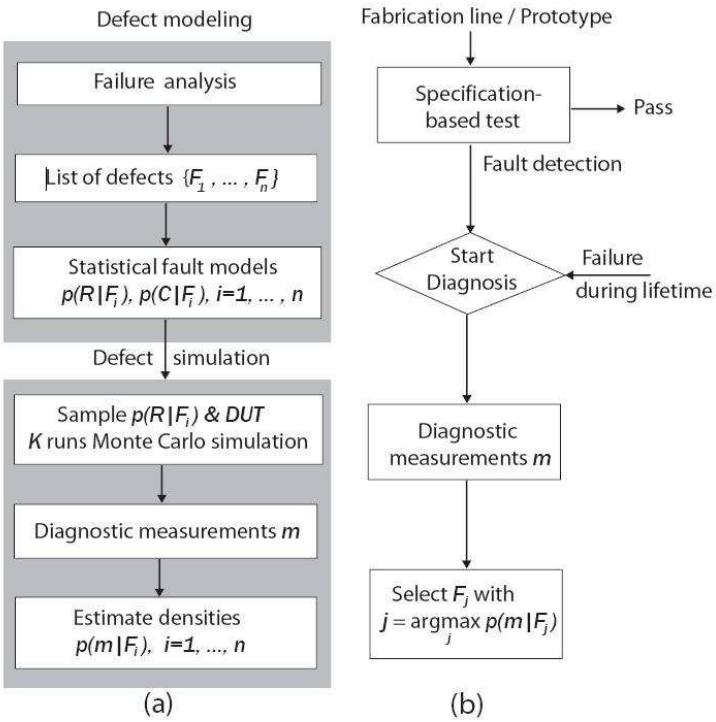


Figure 8.10: Méthodologie du diagnostic: (a) extraction de la densité de probabilité pour le diagnostic et (b) flot du diagnostic.

est elle est obtenue à partir d'échantillons expérimentaux de r en utilisant l'estimation non paramétrique. Une fois la densité $p(R|F_i)$ estimée, on peut l'échantillonner pour générer N différentes valeurs de résistance pour le défaut F_i. Ensuite, ces N valeurs sont injectées au niveau layout du circuit lors d'une simulation Monte Carlo post-layout afin d'obtenir les m meures du diagnostic correspondantes. Enfin, avec les mesures de diagnostic obtenues, nous pouvons estimer la fonction de vraisemblance $p(m|F_i)$ pour chaque défaut.

Lors de la phase du diagnostic, les mêmes mesures m sont prises et le défaut prédit sera le défaut F_j avec

$$j = \underset{j}{\text{argmax}} p(m|F_j)P(F_j), \tag{8.9}$$

8.5.2 Cas d'études

Notre cas d'étude est le LNA présenté dans la section précédente. Le layout du circuit est montré dans la Figure 8.11. Nous avons choisi les quatre paramètres S comme les mesures de diagnostic m. Chaque paramètre S est mesuré de 1 GHz à 5 GHz avec un pas de 100 MHz. Au total, nous avons $4 \times 41 = 164$ mesures de diagnostic.

Nous avons construit une liste de défauts selon l'analyse inductive de fautes (IFA : *Inductive Fault Analysis*) et nous avons obtenu 24 défauts. Ce sont les défauts de type court-circuit ou circuit-ouvert. Ensuite pour chaque défaut, nous avons estimé la fonction de densité de probabilité $p(R|F_i)$ selon les échantillons expérimentaux dans [13, 14]. La

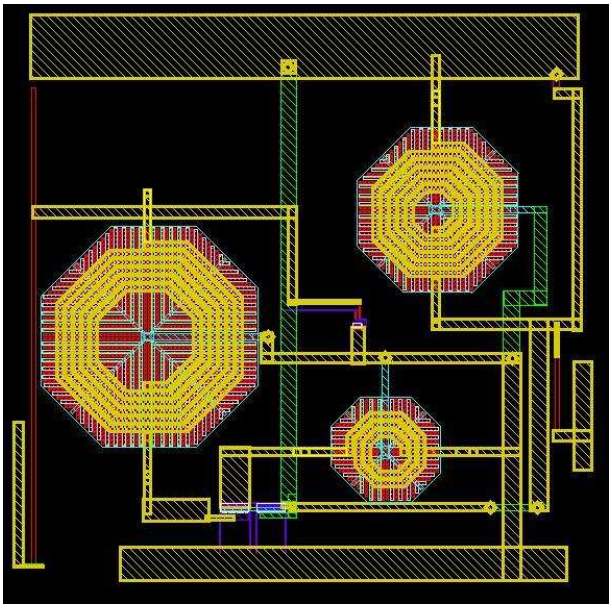


Figure 8.11: Layout du LNA sous test.

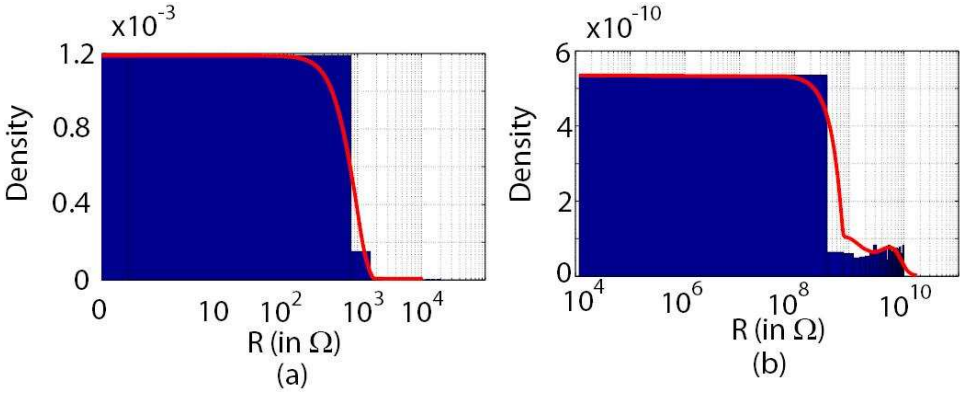


Figure 8.12: L'estimation de la fonction de densité de probabilité $p(R|F_i)$ pour deux types de défaut (a) court-circuit (b) circuit-ouvert

Figure 8.12 montre l'estimation de $p(R|F_i)$ pour les deux types de défaut. Ensuite nous avons effectué 500 simulations Monte Carlo post-layout pour chaque défaut en tenant compte de : (a) les variations de process (b) les parasites d'extraction du layout, et (c) la fonction estimée de la densité de probabilité de résistance de défaut $p(R|F_i)$ [104]. Donc, pour chaque défaut, nous avons obtenu 500 observations des mesures de diagnostic m. Ensuite nous avons estimé la fonction de vraisemblance $p(m|F_i), i = 1, \cdots, 23$ pour chaque défaut.

Afin d'évaluer notre méthodologie du diagnostic, nous avons généré un autre groupe de circuits défectueux. Les mêmes mesures de diagnostic m sont prises et la plupart des circuits sont prédits correctement. C'est-à-dire que pour un circuit ayant le défaut j avec les mesures de diagnostic m correspondantes, l'estimation de la fonction de densité montre que

$$p(m|F_j) > p(m|F_i), \forall i \neq j \tag{8.10}$$

Dans le cas où le défaut sous diagnostic ne donne pas le maximum de $p(m|F_j)$, il est toujours dans les trois valeurs les plus grandes de densité parmi les 23 densités estimées. Cela montre que la méthodologie proposée est capable de tenir en compte les ambiguïtés existantes entre différents types de fautes.

8.6 Résultats expérimentaux

Dans cette section, nous allons présenter la validation expérimentale de la méthodologie du diagnostic sur un cas d'études industriel, qui est un transceiver CAN (*Controller Area Network*) utilisé dans l'automobile. Nous avons focalisé sur les défauts *spot* et nous cherchons à identifier un ensemble de défauts qui sont probable d'avoir lieu dans un circuit défectueux. La méthodologie que nous utilisons est basée sur une combinaison des classificateurs. Nous avons un cas d'études qui nécessite un contrôle de qualité de haut niveau car la sûreté de fonctionnement est essentielle pour ce type de circuit.

8.6.1 Approche proposée

L'approche que nous proposons cherche à faciliter le diagnostic de défauts *spot* dans les circuits analogiques. Le diagnostic peut être considéré comme un problème de reconnaissance de formes. Dans un premier temps, une liste de défauts potentiels dans le circuit sous test peut être identifiée par une analyse inductive de fautes (IFA : *Inductive Fault Analysis*). À partir des mesures de diagnostic du circuit sous test, les défauts dans la liste sont ordonnés selon leurs probabilités d'occurrence en utilisant l'outil de diagnostic qui combine un ensemble des classificateurs. Les classificateurs sont entraînés en utilisant les données de simulation de fautes. Durant la simulation de fautes, nous considérons différentes valeurs de résistance de défaut avec chaque défaut représenté par une classe de fautes. Chaque classificateur attribue un score à chaque défaut et les scores de différents classificateurs sont combinés afin d'obtenir un seul score pour chaque défaut. Le diagnostic est très important pour ce type de circuits particulier car il est utilisé dans les systèmes automobiles. En plus, il est nécessaire d'étudier les données manquantes (*Missing data*) dans la simulation de fautes et dans test du circuit pour ce cas d'études réel à grande échelle.

La figure 8.13 montre une description de la méthodologie proposée. La première étape consiste à effectuer la simulation de fautes et construire le dictionnaire de fautes. En particulière, la liste de Q emplacements de défauts probables est générée par une analyse inductive de fautes (IFA). Cette liste est assumée de représenter la totalité de défauts susceptible d'avoir lieu dans la pratique. Un défaut F_j, $j = 1, \cdots, Q$, est modélisé soit par un court-circuit soit par un circuit-ouvert qui a une certaine valeur de résistance R. Cette résistance peut avoir une valeur selon la distribution $p(R|Fj)$ obtenue par les données de caractérisation de défauts comme montré dans la section précédente.

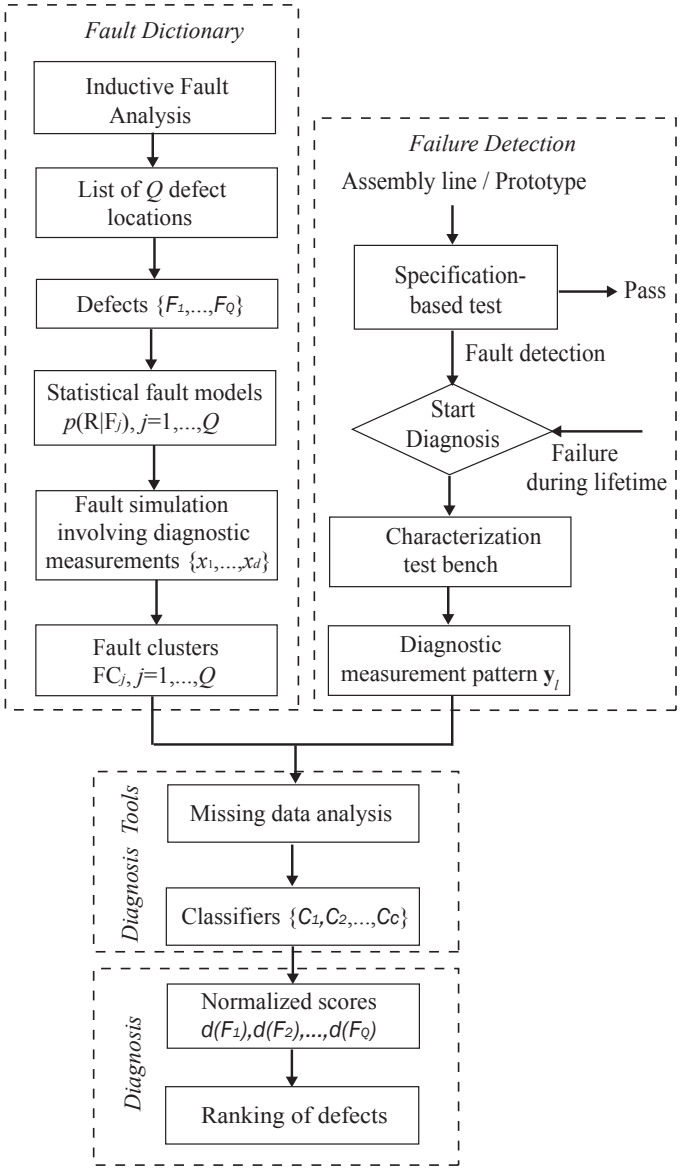


Figure 8.13: Flot du diagnostic proposé.

Ensuite nous choisissons d mesures de diagnostic afin d'effectuer la simulation de fautes. Le résultat de simulation de fautes peut être exprimé sous forme :

$$\mathbf{x}_i^j = \left[x_{i,1}^j, x_{i,2}^j, \cdots, x_{i,d}^j\right] \tag{8.11}$$

où $\mathbf{x}_i^j$ représente le vecteur de mesures de diagnostic pour le j-ième défaut qui une valeur de résistance R_i échantillonnée à partir de $p(R|Fj)$. Pour n valeurs de résistances, la j-ième classe de fautes peut être exprimée sous forme :

$$FC_j = \left\{\mathbf{x}_1^j, \cdots, \mathbf{x}_n^j\right\}. \tag{8.12}$$

nous pouvons utiliser les tests de spécification comme mesures de diagnostic dans un premier temps. Des mesures supplémentaires peuvent être rajoutées pour améliorer le résultat du diagnostic.

Ensuite dans la phase du diagnostic, les mêmes mesures de diagnostic de d dimensions sont obtenues pour le l-ième circuit sous test. Elles sont exprimées sous forme :

$$\mathbf{y}_l = \left[y_{l,1}, y_{l,2}, \cdots, y_{l,d}\right]. \tag{8.13}$$

Pour construire l'outil de diagnostic, nous avons besoin de traiter le problème de données manquantes (*missing data*) dans les vecteurs FC_j, $j = 1, \cdots, Q$, et $\mathbf{y}_l$. Spécifiquement, la simulation de fautes de certaines mesures de diagnostic n'a pas pu convergé ou des valeurs irréalistes sont obtenues. Si la valeur de la k-ième mesure du j-ième défaut avec la résistance R_i est manquante, $x_{i,k}^j$ est considéré comme manquant. De la même manière, si une mesure de diagnostic $y_{l,k}$ sur un circuit sous test atteint sa limite d'instrument, elle sera aussi considérée comme manquante. Nous allons par la suite montrer plus de détails sur le traitement des valeurs manquantes.

L'outil de diagnostic inclut un ensemble de c classificateurs $\{C_1, C_2, \cdots, C_c\}$ entraînés en utilisant le dictionnaire de fautes. Chaque classificateur attribue un score de probabilité à chaque défaut. Ensuite, les classificateurs sont combinés afin d'attribuer un seul score $d(F_j)$ à chaque défaut. L'efficacité de cette méthode de combinaison a été montrée dans [99, 100].

Analyse des valeurs manquantes

L'injection d'un défaut dans la *netlist* du circuit peut rendre le système des équations du simulateur insoluble. Par conséquence, il existe des mesures de diagnostic qui ne peuvent pas être obtenues dans la simulation de fautes pour certains défauts. Autrement dit, il existe des valeurs manquantes dans les classes de fautes FC_j dû à la non-convergence de simulation de fautes. Le problème des valeurs manquantes existe aussi dans les mesures de diagnostic $\mathbf{y}_l$ du circuit sous test. En effet, quand une mesure atteint la limite d'instrument à cause d'un défaut, sa valeur est "forcée" d'être égale à la valeur de limite d'instrument. Dans ce cas, nous ne pouvons que utiliser l'information passe/échoue et nous ne pouvons pas considérer sa valeur absolue.

Supposons que z_k soit la valeur de k-ième mesure de diagnostic. Selon la notation dans la section 8.6.1, $z_k = \{x_{i,k}^j, y_{l,k}\}_{i=1,\cdots,n}^{j=1,\cdots,Q}$. Nous appliquons ici le mécanisme de NMAR (*Not Missing At Random*) [101] qui déclare que z_k est manquant si $|z_k| > n_{th}$, où n_{th}

est une valeur de seuil. Le fait que les mesures de diagnostic sont normalisé entre [-1,1] nous permet d'utiliser une seule valeur de seuil n_{th}. Nous avons suivi la méthode proposée dans [101] qui considère plusieurs valeurs de n_{th} car la définition de n_{th} dépend de plusieurs éléments tel que la configuration/les parasites de la carte de test, les limites d'instrument, etc.

L'approche que nous proposons pour traiter les valeurs manquantes est :

1. Si $y_{l,k}$ est manquant, alors la k-iéme mesure de diagnostic sera exclue dans l'analyse.

2. Si $x_{i,k}^j$ est manquant mais le même élément est disponible pour au moins une autre valeur de résistance du j-ième défaut, alors $x_{i,k}^j$ sera remplacé par la valeur moyenne des éléments disponibles. Cette approche est appelée imputation moyenne (*mean imputation*) [101]. Par exemple, si $x_{h,k}^j$ est disponible pour $h = 1, \cdots, i-1, i+1, \cdots, n$, alors $x_{i,k}^j$ sera remplacé par $\frac{1}{n-1}\sum_{h \neq i} x_{h,k}^j$.

3. Supposons que

$$A^j = \begin{bmatrix} \mathbf{x}_1^j \\ \mathbf{x}_2^j \\ \vdots \\ \mathbf{x}_n^j \end{bmatrix} \tag{8.14}$$

soit la matrice qui correspond à la j-ième classe de fautes FC_j et

$$A = \begin{bmatrix} A^1 \\ A^2 \\ \vdots \\ A^Q \end{bmatrix}. \tag{8.15}$$

La matrice A est parcouru et quand un élément $x_{i,k}^j$ est manquant et il ne peut pas être remplacé avec la méthode d'imputation moyenne dans l'étape 2, soit le j-ième défaut soit la k-ième mesure de diagnostic sera exclu. Pour décider si c'est le défaut ou la mesure à être exclu, on compte le nombre de défauts pour lesquels la k-ième mesure est manquante, noté par N_{def}^k, et le nombre de mesures pour lesquelles le j-ième défaut est manquant, noté par N_{meas}^j. Si

$$\frac{N_{def}^k}{Q} > \beta \times \frac{N_{meas}^j}{d}, \tag{8.16}$$

où β est un coefficient à fixer par l'utilisateur, alors on exclut la k-ième mesure, sinon on exclut le j-ième défaut. Si β est fixé petit, plus de mesures de diagnostic seront excluent, sinon plus de défauts seront excluent.

Méthode de classification

Nous allons présenter dans cette section les différentes méthodes de classification. Chaque classificateur attribue un score entre [0,1] à chaque défaut, ensuite les scores de différents classificateurs sont combinés afin de donner un seul score normalisé.

1. Distance euclidienne

Cette méthode considère les distances euclidienne entre $\mathbf{y}_l$ et $\mathbf{x}_i^j$, $i = 1, \cdots, n$, $j = 1, \cdots, Q$, définies par

$$d(\mathbf{x}_i^j, \mathbf{y}_l) = \sqrt{(x_{i,1}^j - y_{l,1})^2 + \cdots + (x_{i,d}^j - y_{l,d})^2}. \tag{8.17}$$

Nous définissons la distance minimale par

$$d_{min} = \min_{i,j} d(\mathbf{x}_i^j, \mathbf{y}_l) \tag{8.18}$$

qui nous permet de normaliser les distances entre [0,1]

$$d^{'}(\mathbf{x}_i^j, \mathbf{y}_l) = d_{min}/d(\mathbf{x}_i^j, \mathbf{y}_l). \tag{8.19}$$

La distance minimal entre $\mathbf{x}_i^j$ et $\mathbf{y}_l$ est normalisée à 1. Nous attribuons ensuite un score normalisé pour chaque défaut F_j

$$d_1(F_j) = \frac{1}{n} \sum_{i=1}^{n} d^{'}(\mathbf{x}_i^j, \mathbf{y}_l). \tag{8.20}$$

2. Distance de Mahalanobis

Cette méthode considère la distance de Mahalanobis entre $\mathbf{y}_l$ et chaque classe de faute FC_j, $j = 1, \cdots, Q$.

$$d_M(FC_j, \mathbf{y}_l) = \sqrt{(\mathbf{y}_l - \mathbf{u}_j)^T \times S_j^{-1} \times (\mathbf{y}_l - \mathbf{u}_j)}, \tag{8.21}$$

où $\mathbf{u}_j = [u_{j,1}, \cdots, u_{j,d}]$ est le vecteur des valeurs moyennes avec

$$u_{j,k} = \sum_{i=1}^{n} x_{i,k}^j, \tag{8.22}$$

$$S_j = \begin{pmatrix} E[(x_{i,1}^j - u_{j,1})(x_{i,1}^j - u_{j,1})] & \cdots & E[(x_{i,1}^j - u_{j,1})(x_{i,d}^j - u_{j,d})] \\ E[(x_{i,2}^j - u_{j,2})(x_{i,1}^j - u_{j,1})] & \cdots & E[(x_{i,2}^j - u_{j,2})(x_{i,d}^j - u_{j,d})] \\ \vdots & \cdots & \vdots \\ E[(x_{i,d}^j - u_{j,d})(x_{i,1}^j - u_{j,1})] & \cdots & E[(x_{i,d}^j - u_{j,d})(x_{i,d}^j - u_{j,d})] \end{pmatrix} \tag{8.23}$$

S_j est la matrice de covariance montré dans (8.23), et $E[\cdot]$ indique la valeur espérée calculée sur toutes les valeurs de résistance $i = 1, \cdots, n$. Nous définissons la distance minimale comme :

$$d_{Mmin} = \min_j d_M(FC_j, \mathbf{y}_l), \tag{8.24}$$

Nous attribuons ensuite un score normalisé entre [0,1] pour chaque défaut :

$$d_2(F_j) = d_{Mmin}/d_M(FC_j, \mathbf{y}_l), \tag{8.25}$$

où, comme pour la distance euclidienne, le défaut avec le score plus grand sera le plus probable.

3. Estimation non-paramétrique de la densité par noyau (KDE)

Rappelons le théorème de Bayes qui déclare que la probabilité *a posteriori* qu'un circuit défectueux contienne le défaut F_j est exprimée comme

$$P(F_j|\mathbf{y}) = \frac{f_j(\mathbf{y}|F_j)P(F_j)}{p(\mathbf{y})}, \tag{8.26}$$

où est $P(F_j)$ est la probabilité *a priori* du défaut F_j, $f_j(\mathbf{y}|F_j)$ est la probabilité conditionnelle jointe de la fonction de densité $\mathbf{y}$ avec la présence du défaut F_j, et $p(\mathbf{y})$ est la fonction de densité de probabilité de $\mathbf{y}$. Un circuit défectueux est plus probable d'avoir le défaut F_m si

$$P(F_m|\mathbf{y}) > P(F_j|\mathbf{y}), \quad \forall j \neq m. \tag{8.27}$$

En combinant (8.26) et (8.27), nous avons

$$f_m(\mathbf{y}|F_m)P(F_m) > f_j(\mathbf{y}|F_j)P(F_j), \quad \forall j \neq m. \tag{8.28}$$

La probabilité *a priori* de défauts peut être obtenue par un IFA. Ici, nous assumons qu'elles sont équiprobables. Donc, un circuit défectueux est plus probable d'avoir le défaut F_m si

$$f_m(\mathbf{y}|F_m) > f_j(\mathbf{y}|F_j), \quad \forall j \neq m. \tag{8.29}$$

Cette méthode estime les densités $f_j(\mathbf{y}|F_j)$, $j = 1, \cdots, Q$ avec les observations disponibles $\mathbf{x}_i^j$, $i = 1, \cdots, n$ dans la j-ième classe de fautes FC_j. Afin d'estimer $f_j(\mathbf{y}|F_j)$, nous n'assumons aucune hypothèse sur sa forme paramétrique (e.g. normale) et nous utilisons une estimation non-paramétrique. L'estimation de densité par noyau est définie comme [92]

$$\hat{f}_j(\mathbf{y}|F_j) = \frac{1}{n \times h^d} \sum_{i=1}^{n} K_e(\frac{1}{h}(\mathbf{y} - \mathbf{x}_i^j)) \tag{8.30}$$

où h est un paramètre indiquant la largeur de bande, $K_e(t)$ est le noyau Epanechnikov

$$K_e(t) = \begin{cases} \frac{1}{2}c_d^{-1}(d+2)(1-t^Tt) & \text{if} \quad t^Tt < 1 \\ 0 & \text{otherwise} \end{cases} \tag{8.31}$$

et $c_d = 2\pi^{d/2}/(d \cdot \Gamma(d/2))$ est le volume de sphère de d dimensions. Ici, nous utilisons l'estimation adaptative, définie par [92]:

$$\hat{f}_{j,\alpha}(\mathbf{y}|F_j) = \frac{1}{n}\sum_{i=1}^{n} \frac{1}{(h\cdot\lambda_i)^d} K_e(\frac{1}{h\cdot\lambda_i}(\mathbf{y}-\mathbf{x}_i^j)) \tag{8.32}$$

où le facteur local de la largeur de bande λ_i est définie par:

$$\lambda_i = \{\hat{f}_j(\mathbf{x}_i^j|F_j)/g\}^{-\alpha}, \tag{8.33}$$

$\hat{f}_j(\mathbf{x}_i^j|F_j)$ est l'estimation de densité pilote donnée dans (6.17), g est la moyenne géométrique

$$\log g = n^{-1}\sum_{i=1}^{n} \log \hat{f}_j(\mathbf{x}_i^j|F_j) \tag{8.34}$$

et α est un paramètre qui contrôle les largeurs de bande locales. Plus α est grand, plus il y aura d'espace des mesures de diagnostic où la densité $\hat{f}_{j,\alpha}(\mathbf{y}|F_j)$ est non-zéro. Etant donné un circuit défectueux avec le vecteur de mesures $\mathbf{y}_l$, nous attribuons un score normalisé entre [0,1] à chaque défaut :

$$d_3(F_j) = \frac{\hat{f}_{j,\alpha}(\mathbf{y}_l|F_j) - \hat{f}_{min}}{\hat{f}_{max} - \hat{f}_{min}}, \tag{8.35}$$

où

$$\hat{f}_{min} = \min_j \hat{f}_{j,\alpha}(\mathbf{y}_l|F_j) \tag{8.36}$$

$$\hat{f}_{max} = \max_j \hat{f}_{j,\alpha}(\mathbf{y}_l|F_j). \tag{8.37}$$

comme dans d'autres classificateurs, le défaut ayant la plus grande densité $\hat{f}_{j,\alpha}(\mathbf{y}_l|F_j)$ est normalisé à 1. En plus, si $d_3(F_j)$ est zéro pour tous les défauts, alors $\mathbf{y}_l$ est considéré "étranger" à toutes les classes de fautes. Dans ce cas, on peut conclure que le défaut existe dans le circuit n'a pas été modélisé dans le dictionnaire de fautes. Donc, contrairement aux autres classificateurs qui attribuent toujours un score à chaque défaut, l'estimation non-paramétrique est la seule méthode capable d'identifier un défaut non modélisé.

4. Machine à vecteurs de support (SVM : *Support Vector Machine*)

Cette méthode alloue les frontières de séparation dans l'espace des mesures de diagnostic pour séparer les Q classes de fautes. En particulière, nous utilisons les SVMs [70] pour définir les frontières de séparation au milieu des distances Euclidiennes entre Q classes de fautes.

Les classificateurs SVM étaient développés pour la classification binaire. Pour la classification avec Q classes ($Q > 2$), nous pouvons changer le problème à $\binom{Q}{2}$ problèmes de classification binaire et appliquer ensuite la stratégie un-contre-un (*one-against-one*). Dans cette stratégie, chaque classificateur binaire attribue le circuit sous test à une classe de fautes, ensuite le vote pour la classe attribuée est incrémenté un. Finalement la classe avec le plus grand nombre de vote sera la classe à laquelle le circuit sous test appartient.

Cette méthode attribue les scores normalisés entre [0,1] à chaque défaut selon

$$d_4(F_j) = N_j/N_{max}, \tag{8.38}$$

où N_j est le nombre de classificateurs qui ont attribué le vote au défaut F_j et

$$N_{max} = \max_j N_j. \tag{8.39}$$

5. Méthode de vérification passé/échoué

Cette méthode examine simplement $\mathbf{y}_l$ et $\mathbf{x}_i^j$ par vérification d'information passé/échoué des mesures de diagnostic. Formellement, nous considérons l'indicateur de spécification $I_{i,k}^j$, tel que (a) $I_{i,k}^j = 1$ si $\mathbf{y}_l$ et $\mathbf{x}_i^j$ vérifient la spécification en même temps ou $\mathbf{y}_l$ et $\mathbf{x}_i^j$ échouent la spécification en même temps pour la k-ième mesure de diagnostic et (b) $I_{i,k}^j = 0$ si seulement un entre $\mathbf{y}_l$ et $\mathbf{x}_i^j$ vérifie la spécification pour la k-ième mesure de diagnostic. Le score normalisé entre [0,1] pour le défaut F_j est défini par:

$$d_5(F_j) = \frac{1}{n}\sum_{i=1}^{n}\frac{1}{d}\sum_{k=1}^{d} I_{i,k}^j. \tag{8.40}$$

Combinaison des classificateurs

Nous proposons d'utiliser la méthode de moyenne pour combiner les scores des différents classificateurs [99, 100]. Pour y_l, les scores de tous les classificateurs de tous les F_j sont exprimés par [100]:

$$DP(y_l) = \begin{pmatrix} d_1(F_1) & \cdots & d_1(F_j) & \cdots & d_1(F_Q) \\ \vdots & & \vdots & & \vdots \\ d_i(F_1) & & d_i(F_j) & & d_i(F_Q) \\ \vdots & & \vdots & & \vdots \\ d_c(F_1) & \cdots & d_c(F_j) & \cdots & d_c(F_Q) \end{pmatrix} \tag{8.41}$$

où c est le nombre de classificateurs considérés, Q est le nombre des classes de fautes, et $d_i(F_j)$ est le score normalisé de la j-ième classe de fautes du i-ième classificateur. Le score de la classe F_j pour c classificateurs est calculé

$$d_{com}(F_j) = \frac{1}{c}\sum_{i=1}^{c} d_i(F_j). \tag{8.42}$$

Combinaison des modèles manquants

Comme indiqué dans la section 8.6.1, il est plus approprié de considérer plusieurs modèles manquants (plusieurs valeurs de n_{th}) dans l'analyse de NMAR. Par conséquence, le score final pour le défaut F_j est donné par

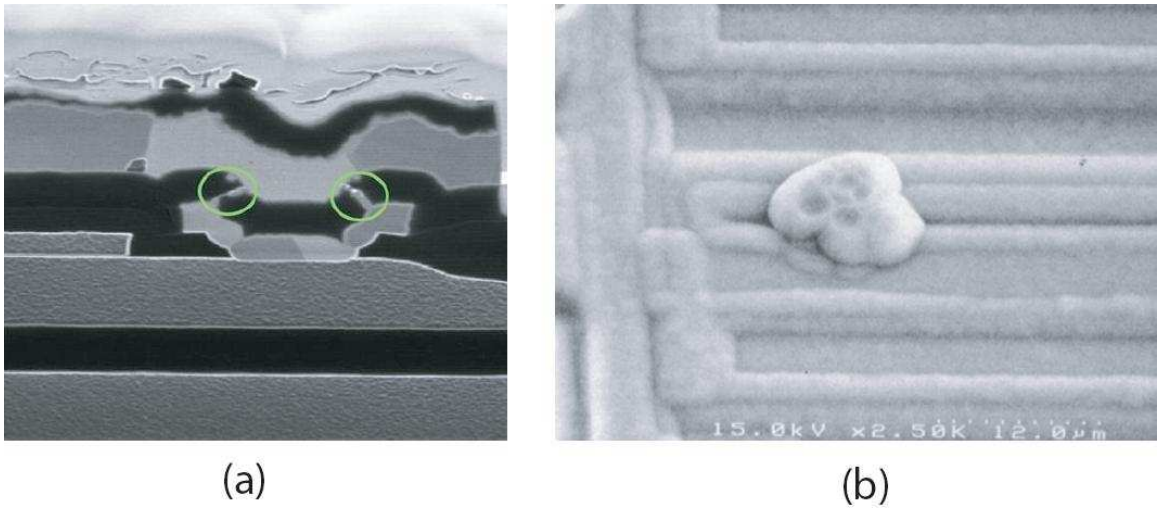


Figure 8.14: (a) Image réalisée par sonde ionique focalisée (FIB) du défaut observé dans DUT 18 et (b) Image réalisée par microscopie électronique à balayage (SEM) du défaut observé dans DUT 26.

$$d_{final}(F_j) = \frac{1}{p}\sum_{i=1}^{p} d^i_{com}(F_j), \tag{8.43}$$

où $d^i_{com}(F_j)$ indique le score du défaut F_j pour la i-ième valeur de n_{th}, $i = 1, \cdots, p$.

8.6.2 Cas d'études

Notre cas d'études est un transceiver CAN (*Controller Area Network*) conçu par NXP Semiconductors. Le circuit est fabriqué en grand volume et il constitue une partie essentielle pour le système électronique dans les automobiles. Par conséquence, il est très important de diagnostiquer les sources de défaillance afin d'assurer un meilleur contrôle de qualité et améliorer la conception pour éviter la reproduction d'un défaut similaire.

Nous avons 29 circuits défectueux venant de différents lots. L'analyse classique de défauts a été réalisée et elle montre que ces circuits contiennent les défauts du type court-circuit. Par exemple, la figure 8.14 montre (a) une image réalisée par sonde ionique focalisée (FIB) du défaut observé dans DUT 18 et (b) une image réalisée par microscopie électronique à balayage (SEM) du défaut observé dans DUT 26. Afin de valider la méthodologie, nous assumons que le défaut dans chaque circuit est inconnu. Nous avons obtenu une liste de Q =923 défauts du type court-circuit par IFA. Chaque circuit est modélisé par n =3 valeurs de resistances (e.g. $\{5\Omega, 50\Omega, 200\Omega\}$). Donc, nous avons effectué en total $3 \times 923 = 2769$ simulations de fautes. Durant l'entraînement d'outils de diagnostic, nous avons décidé d'abandonner la méthode de distance de Mahalanobis pour des raisons (a) la matrice de covariance S_j de certaines classes de fautes est non-inversible (b) l'existence de corrélation entre les mesures de diagnostic, ainsi que la méthode de SVM pour des raisons (a) le nombre d'observations (e.g. 3) pour chaque défauts dans l'entraînement est insuffisant (b) le nombre de dimensions (e.g. 97) est trop élevé (c) le

nombre de classes de fautes (e.g. 923) est trop élevé.

Résultat du diagnostic

Nous avons combiné 3 classificateurs : la distance euclidienne, l'estimation non-paramétrique de densité et la méthode de vérification passé/échoué. Le tableau 8.1 montre les 5 défauts les plus probables selon leurs scores pour les 29 circuits défectueux. Le tableau 8.2 montre une comparaison entre les différentes méthodes de classification ainsi que leur combinaison. Comme nous pouvons observer dans le tableau, la méthode de combinaison donne un meilleur résultat du diagnostic.

8.7 Conclusions et travaux futurs

Dans cette thèse nous avons présenté une méthodologie de modélisation et de diagnostic de fautes pour les circuits analogiques/mixtes. Une nouvelle approche basée sur l'apprentissage automatique a été proposée afin de considérer les fautes catastrophiques et paramétriques en même temps dans le diagnostic. Ensuite, nous avons focalisé sur le diagnostic de défauts *spot* qui sont considérés comme le mécanisme de défaut principal de circuits intégrés. Enfin, la méthodologie de diagnostic proposée a été validée par les données de circuits défectueux fournies par NXP Semiconductors aux Pays-bas.

En terme de travaux futurs, nous proposons de

1. Construire des modèles de fautes plus précis et réalistes.
2. Optimiser les stimuli de test et mesures de diagnostic afin d'améliorer le diagnostic et résoudre l'ambiguïté des fautes.
3. Améliorer la méthode de traitement des valeurs manquantes. La définition des valeurs manquantes avec une valeur de seuil n_{th} n'est pas une tâche facile car n_{th} peut dépendre plusieurs facteurs environnementaux. On peut envisager une définition de n_{th} plus réaliste en prenant compte de la configuration de carte de test, le matériel de test, les limites d'instruments de test, etc.

Table 8.1: Résultat du diagnostic.

DUT	True defect	Defect ranking	Normalized scores
1	107	**107** 90 920 114 347	0.924 0.923 0.923 0.923 0.923
2	320	**320** 341 126 374 111	0.948 0.867 0.833 0.827 0.822
3	125	47 616 **125** 681 360	0.914 0.839 0.838 0.837 0.837
4	101	**101** 117 459 50 388	0.831 0.829 0.826 0.817 0.817
5	216	**216** 666 192 516 120	0.831 0.795 0.792 0.788 0.785
6	300	524 608 744 294 789	0.900 0.890 0.862 0.855 0.850
7	20	**20** 126 24 27 111	0.889 0.866 0.862 0.850 0.849
8	27	**27** 111 126 446 341	0.891 0.856 0.837 0.834 0.834
9	104	111 **104** 465 721 126	0.848 0.844 0.839 0.823 0.822
10	21	310 682 524 789 608	0.867 0.858 0.855 0.855 0.851
11	101	**101** 117 459 50 388	0.831 0.829 0.826 0.818 0.817
12	19	**19** 541 106 562 595	0.810 0.794 0.780 0.780 0.780
13	19	**19** 541 562 595 106	0.799 0.791 0.788 0.771 0.771
14	140	401 **140** 457 40 919	0.936 0.912 0.911 0.910 0.910
15	20	**20** 24 126 27 111	0.887 0.865 0.862 0.853 0.849
16	101	**101** 117 459 50 388	0.831 0.829 0.826 0.817 0.817
17	107	**107** 90 920 114 347	0.924 0.923 0.923 0.923 0.923
18	31	117 **31** 50 388 622	0.901 0.888 0.882 0.881 0.880
19	101	252 305 366 363 31	0.883 0.857 0.846 0.844 0.843
20	19	**19** 541 106 562 595	0.821 0.794 0.793 0.780 0.780
21	156	524 608 744 789 682	0.903 0.893 0.872 0.872 0.866
22	20	**20** 126 24 27 111	0.882 0.870 0.867 0.864 0.853
23	107	**107** 90 920 114 347	0.924 0.923 0.923 0.923 0.923
24	22	**22** 19 541 338 106	0.826 0.808 0.808 0.795 0.795
25	107	**107** 90 920 114 347	0.924 0.923 0.923 0.923 0.923
26	380	666 192 516 676 457	0.910 0.906 0.905 0.904 0.903
27	376	383 456 112 34 196	0.924 0.920 0.830 0.826 0.824
28	28	666 192 516 355 676	0.910 0.907 0.898 0.896 0.896
29	300	524 608 744 475 215	0.896 0.896 0.866 0.864 0.862

Table 8.2: Comparaison des résultats du diagnostic avec différents classificateurs ainsi que leur combinaison.

Diagnosis method	First choice	First three choices	First five choices
Euclidean distance	10	11	19
Non-parametric KDE	7	7	11
Pass/fail verification	10	15	16
Combination method	17	21	21

Bibliography

[1] Semiconductor Industry Association (SIA), "International technology roadmap for semiconductors (ITRS)," http://www.itrs.net/Common/2010ITRS/ExecSum2010.pdf, 2010 edition.

[2] "Mask misalignment," http://www.siliconfareast.com.

[3] D.E. Grosjean, "Reducing defects in integrated surface-micromachined accelerometers," http://www.micromagazine.com/.

[4] V.K. Jayatilaka and P.B. Espinasse, "Lowering magnetic fields in metal dry-etch recipes to reduce mos leakage levels," http://www.micromagazine.com.

[5] F. Fantini and C. Morandi, "Failure modes and mechanisms for VLSI ICs - a review," in *IEE Proceedings, Part G*, 1985, vol. 132, pp. 74–81.

[6] "Via faults," http://www.si2.org.

[7] "Package-related failure mechanisms and attributes," http://www.siliconfareast.com.

[8] "Electromigration," http://en.wikipedia.org/wiki/Electromigration.

[9] "Oxide breakdown," http://www.siliconfareast.com.

[10] C.H. Stapper, "Modelling of integrated circuit defect sensitivities," *IBM Journal Research Development*, vol. 27, no. 6, pp. 549–557, 1983.

[11] S. Krishnan, K. D. Doornbos, R. Brand, and H. G. Kerkhoff, "Block-level bayesian diagnosis of analogue electronic circuits," in *Design, Automation & Test in Europe Conference*, 2010, pp. 1767–1772.

[12] B. Razavi, *RF Microelectronics*, Prentice Hall PTR, 1998.

[13] R. Rodriguez-Montanes, E. Bruis, and J. Figueras, "Bridging defects resistance measurements in a CMOS process," in *Proc. IEEE International Test Conference*, 1992, pp. 892–899.

[14] R. Rodriguez-Montanes, J.P. de Gyvez, and P. Volf, "Resistance characterization for weak open defects," *IEEE Design & Test of Computers*, vol. 19, no. 5, pp. 18–26, 2002.

[15] D.P. Vallett and J.M. Soden, "Finding fault with deep-submicron ICs," *IEEE Spectrum*, vol. 34, pp. 39–50, 1997.

[16] M. Sachdev and J.P. de Gyvez, *Defect-oriented testing for nano-metric CMOS VLSI circuits*, Springer Verlag, 2007.

[17] M. Abramovici, M. A. Breuer, and A. D. Friedman, *Digital Systems Testing and Testable Design*, IEEE Press, 1990.

[18] W. Maly, A.J. Strojwas, and S.W. Director, "VLSI yield prediction and estimation: A unified framework," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 5, pp. 114–130, 1986.

[19] M.J.M. Pelgrom, A.C.J. Duinmaijer, and A.P.C. Welbers, "Matching properties of MOS transistors," *IEEE Journal of Solid-State Circuits*, vol. 24, no. 5, pp. 1433–1439, 1989.

[20] K.R. Lakshmikumar, R.A. Hadaway, and M.A. Copeland, "Characterization and modeling of mismatch in MOS transistors for precision analog design," *IEEE Journal of Solid-State Circuits*, vol. 13, no. 1, pp. 1057–1062, 1986.

[21] F. Barson, "Emitter-collector shorts in bipolar devices," *IEEE Journal of Solid-State Circuits*, vol. 11, no. 4, pp. 505–510, 1976.

[22] Z. Zhang and D.A. Rabson, "Diagnosis and location of pinhole defects in tunnel junctions using only electrical measurements," *Journal of Applied Physics*, vol. 95, no. 1, pp. 199–203, 2004.

[23] A.F. Puttlitz, J.G. Ryan, and T.D. Sullivan, "Semiconductor interlevel shorts caused by hillock formation in Al-Cu metallization," *IEEE Transactions on Components, Hybrids, and Manufacturing Technology*, vol. 12, no. 4, pp. 619–626, 1989.

[24] B. Bacconnier, G. Lormand, M. Papapietro, M. Achard, and A.-M. Papon, "A study of heating rate and texture influences on annealing hillocks by a statistical characterization of al thin-film topography," *Journal of Applied Physics*, vol. 64, no. 11, pp. 6483–6489, 1988.

[25] W. Maly, "Modeling of lithography related yield losses for CAD of VLSI circuits," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 4, pp. 166–177, 1985.

[26] J. Pineda de Gyvez and C. Di, "IC defect sensitivity for footprint-type spot defects," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 11, no. 1, pp. 638–658, 1992.

[27] T. Yanagawa, "Yield degradation of integrated circuits due to spot defects," *IEEE Transactions on Electron Devices*, vol. ED-19, pp. 190–197, 1972.

[28] J.R. Black, "Electromigration - A brief survey and some recent results," *IEEE Transactions on Electron Devices*, vol. 16, no. 4, pp. 338–347, 1969.

[29] D.K. Schroder and J.A. Babcock, "Negative bias temperature instability: Road to cross in deep submicron silicon semiconductor manufacturing," *Journal of Applied Physics*, vol. 94, pp. 1–18, 2003.

[30] K.L. Chen, S.A. Saller, I.A. Groves, and D.B. Scott, "Reliability effects on MOS transistors due to hot-carrier injection," *IEEE Journal of Solid-State Circuits*, vol. 20, no. 1, pp. 306–313, 1985.

[31] A.M. Yassine, H.E. Nariman, M. McBride, M. Uzer, and K.R. Olasupo, "Time dependent breakdown of ultrathin gate oxide," *IEEE Transactions on Electron Devices*, vol. 47, no. 2, pp. 1416–1420, 2000.

[32] M. Soma, "Challenges in analog and mixed-signal fault models," *IEEE Circuits & Devices Magazine*, vol. 12, no. 1, pp. 16–19, 1996.

[33] F.J. Ferguson and J.P. Shen, "A CMOS fault extractor for inductive fault analysis," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 7, pp. 1181–1194, 1988.

[34] M. J. Ohletz, "Defect-oriented vs schematic-level based fault simulation for mixed-signal ICs," in *IEEE International Test Conference*, 1996, vol. 96, pp. 511–520.

[35] R. Glang, "Defect size distribution in VLSI chips," *IEEE Transactions on Semiconductor Manufacturing*, vol. 4, no. 4, pp. 265–269, 1991.

[36] A.V. Ferris-Prabhu, "Modeling the critical area in yield forecasts," *IEEE Journal of Solid-State Circuits*, vol. 20, no. 4, pp. 874–878, 1985.

[37] J.B. Khare, W. Maly, and M.E. Thomas, "Extraction of defect size distributions in an IC layer using test structure data," *IEEE Transactions on Semiconductor Manufacturing*, vol. 7, no. 3, pp. 354–368, 1994.

[38] J.E. Nelson, T. Zanon, J.G. Brown, O. Poku, R.D. Blanton, W. Maly, B. Benware, and C. Schuermyer, "Extracting defect density and size distributions from product ICs," *IEEE Design & Test of Computers*, vol. 23, no. 5, 2006.

[39] C.L. Henderson, J.M. Soden, and C.F. Hawkins, "The behavior and testing implications of CMOS IC logic gate open circuits," in *Proc. IEEE International Test Conference*, 1991.

[40] E. Acar and S. Ozev, "Diagnosis of the failing component in RF receivers through adaptive full-path measurements," in *Proc. IEEE VLSI Test Symp.*, 2005, pp. 374–379.

[41] S. Sunter and N. Nagi, "Test metrics for analog parametric faults," in *IEEE VLSI Test Symposium*, 1999, pp. 226–34.

[42] J. Tongbong, S. Mir, and J. L. Carbonero, "Evaluation of test measures for LNA production testing using a multinormal statistical model," in *Design, Automation and Test in Europe*, 2007, pp. 731–736.

[43] A. Bounceur, S. Mir, E. Simeu, and L. Rolindez, "Estimation of test metrics for the optimisation of analogue circuit testing," *Journal of Electronic Testing: Theory and Applications*, vol. 23, no. 6, pp. 471–484, 2007.

[44] H.-G. Stratigopoulos, J. Tongbong, and S. Mir, "A general method to evaluate RF BIST techniques based on non-parametric density estimation," in *Design, Automation and Test in Europe Conference*, 2008, pp. 68–73.

[45] E. Maricau and G. Gielen, "Efficient variability-aware NBTI and hot carrier circuit reliability analysis," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 29, no. 12, 2010.

[46] E. Liu, W. Kao, E. Felt, and A. Sangiovanni-Vincentelli, "Analog testability analysis and fault diagnosis using behavioral modeling," in *Proc. IEEE Custom Integr. Circuits Conf.*, 1994, pp. 413–416.

[47] S. Chakrabarti, S. Cherubal, and A. Chatterjee, "Fault diagnosis for mixed-signal electronic systems," in *Proc. IEEE Aerosp. Conf.*, 1999, pp. 169–179.

[48] Panel discussion, "Extended Diagnosis Requirements in Automotive Applications", *IEEE Eur. Test Symp.*, Seville, Spain, 2009.

[49] L. Milor, "A tutorial introduction to research on analog and mixed-signal circuit testing," *IEEE Transactions on Circuits and Systems-II: Analog and Digital Signal Processing*, vol. 45, no. 10, pp. 1389–1407, 1998.

[50] S. D. Huss and R. S. Gyurcsik, "Optimal ordering of analog integrated circuit tests to minimize test time," in *ACM/IEEE Design Automation Conference*, 1991, pp. 494–499.

[51] L. Milor and A. L. Sangiovanni-Vincentelli, "Minimizing production test time to detect faults in analog circuits," *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, vol. 13, no. 6, pp. 796–813, 1994.

[52] J. B. Brockman and S. W. Director, "Predictive subset testing: Optimizing IC parametric performance testing for quality, cost, and yield," *IEEE Transactions on Semiconductor Manufacturing*, vol. 2, no. 3, pp. 104–113, 1989.

[53] H.-G. Stratigopoulos, P. Drineas, M. Slamani, and Y. Makris, "RF specification test compaction using learning machines," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 18, no. 6, pp. 998–1002, 2010.

[54] S. Biswas and R. D. Blanton, "Test compaction for mixed-signal circuits using pass-fail test data," in *IEEE VLSI Test Symposium*, 2008, pp. 299–308.

[55] N. Akkouche, S. Mir, and E. Simeu, "Ordering of analog specification tests based on parametric defect level estimation," in *IEEE VLSI Test Symposium*, 2010, pp. 301–306.

[56] R. Voorakaranam, S. S. Akbay, S. Bhattacharya, S. Cherubal, and A. Chatterjee, "Signature testing of analog and RF circuits: Algorithms and methodology," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 54, no. 5, pp. 1018–1031, 2007.

[57] L. Abdallah, H.-G. Stratigopoulos, C. Kelma, and S. Mir, "Sensors for built-in alternate RF test," in *IEEE European Test Symposium*, 2010, pp. 49–54.

[58] H.-G. Stratigopoulos, S. Mir, E. Acar, and S. Ozev, "Defect filter for alternate RF test," in *Proc. IEEE Eur. Test Symp.*, 2009, pp. 101–106.

[59] E. Acar and S. Ozev, "Defect-oriented testing of RF circuits," *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, vol. 27, no. 5, pp. 920–931, 2008.

[60] H.-G. Stratigopoulos and Y. Makris, "Error moderation in low-cost machine-learning-based Analog/RF testing," *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, vol. 27, no. 2, pp. 339–351, 2008.

[61] R. Spina and S. Upadhyaya, "Linear circuit fault diagnosis using neuromorphic analyzers," *IEEE Transactions on Circuits and Systems-II: Analog and Digital Signal Processing*, vol. 44, no. 3, pp. 188–196, 1997.

[62] S. S. Somayajula, E. Sanchez-Sinencio, and J. Pineda de Gyvez, "Analog fault diagnosis based on ramping power supply current signature clusters," *IEEE Trans. Circuits Syst. II, Analog Digit. Signal Process.*, vol. 43, no. 10, pp. 703–712, 1996.

[63] M. Aminian and F. Aminian, "A modular fault-diagnosis system for analog electronic circuits using neural networks with wavelet transform as a preprocessor," *IEEE Trans. Instrum. Meas.*, vol. 56, no. 5, pp. 1546–1554, 2007.

[64] C. Alippi, M. Catelani, A. Fort, and M. Mugnaini, "Automated selection of test frequencies for fault diagnosis in analog electronic circuits," *IEEE Trans. Instrum. Meas.*, vol. 54, no. 3, pp. 1033–1044, 2005.

[65] J.W. Bandler and A.E. Salama, "Fault diagnosis of analog circuits," *IEEE Proceedings*, vol. 73, pp. 1279–1325, 1985.

[66] W. Fenton, T. M. McGinnity, and L. P. Maguire, "Fault diagnosis of electronic systems using intelligent techniques: A review," *IEEE Trans. Syst., Man, Cybern. C, Appl. Rev.*, vol. 31, no. 3, pp. 269–281, 2001.

[67] E. S. Erdogan, S. Ozev, and P. Cauvet, "Diagnosis of assemply failures for system-in-package RF tuners," in *Proc. IEEE Int. Symp. Circuits Syst.*, 2008, pp. 2286–2289.

[68] C.W. Hsu and C.J. Lin, "A comparison of methods for multi-class support vector machines," *IEEE Transactions on Neural Networks*, vol. 13, pp. 415–425, 2002.

[69] B. Ravikumar, D. Thukaram, and H.P. Khincha, "Application of support vector machines for fault diagnosis in power transmission system," *IET Generation, Transmission & Distribution*, vol. 2, no. 1, pp. 119–130, 2008.

[70] N. Cristianini and J. Shawe-Taylor, *Support Vector Machines and Other Kernel-Based Learning Methods*, Cambridge, 2000.

[71] Z. Wang, G. Gielen, and W. Sansen, "Probabilistic fault detection and the selection of measurements for analog integrated circuits," *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, vol. 17, no. 9, pp. 862–872, 1998.

[72] F. Liu, P.K. Nikolov, and S. Ozev, "Parametric fault diagnosis for analog circuits using a bayesian framework," in *IEEE VLSI Test Symposium*, 2006.

[73] B. R. Epstein, M. Czigler, and S. R. Miller, "Fault detection and classification in linear integrated circuits: An application of discrimination analysis and hypothesis testing," *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, vol. 12, no. 1, pp. 102–113, 1993.

[74] N. Sen and R. Saeks, "Fault diagnosis for linear systems via multifrequency measurements," *IEEE Trans. Circuits Syst.*, vol. 26, no. 7, pp. 457–465, 1979.

[75] F. Grasso, A. Luchetta, S. Manetti, and M.C. Piccirilli, "A method for the automatic selection of test frequencies in analog fault diagnosis," *IEEE Transactions on Instrumentation and Measurement*, vol. 56, no. 6, pp. 2322–2329, 2007.

[76] L. Rapisarda and R. A. Decarlo, "Analog multifrequency fault diagnosis," *IEEE Trans. Circuits Syst.*, vol. CAS-30, no. 4, pp. 223–234, 1983.

[77] H. Dai and M. Souders, "Time-domain testing strategies and fault diagnosis for analog systems," *IEEE Trans. Instrum. Meas.*, vol. 39, no. 1, pp. 157–162, 1990.

[78] G. J. Hemink, B. W. Meijer, and H. G. Kerkhoff, "Testability analysis of analog systems," *IEEE Trans. Comput.-Aided Des.*, vol. 9, no. 6, pp. 573–583, 1990.

[79] M. Slamani and B. Kaminska, "Analog circuit fault diagnosis based on sensitivity computation and functional testing," *IEEE Des. Test Comput.*, vol. 9, no. 1, pp. 30–39, 1992.

[80] R. Neumayer, A. Stelzer, F. Haslinger, and R. Weigel, "On the synthesis of equivalent-circuit models for multiports characterized by frequency-dependent parameters," *IEEE Transactions on Microwave Theory and Techniques*, vol. 50, no. 12, 2002.

[81] A. Verschueren, Y. Rolain, R. Vuerinckx, and G. Vandersteen, "Identifying S-parameter models in the Laplace domain for high frequency multiport linear networks," in *IEEE MTT-S International Microwave Symposium Digest*, 1998, vol. 1, pp. 25–28.

[82] F. Liu, S. Ozev, and M. Brooke, "Identifying the source of BW failures in high-frequency linear analog circuits based on S-parameters measurements," *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, vol. 25, no. 11, pp. 2594–2605, 2006.

[83] A. Fanni, A. Giua, and E. Sandoli, "Neural networks for multiple fault diagnosis in analog circuits," in *IEEE International Workshop on Defect and Fault Tolerance in VLSI Systems*, 1993, pp. 303–310.

[84] C. Alippi, M. Catelani, A. Fort, and M. Mugnaini, "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, 2002.

[85] F. Aminian, M. Aminian, and Jr. H. W. Collins, "Analog fault diagnosis of actual circuits using neural networks," *IEEE Transactions on Instrumentation and Measurement*, vol. 51, no. 3, pp. 544–550, 2002.

[86] S. Cherubal and A. Chatterjee, "Parametric fault diagnosis for analog systems using functional mapping," in *Design, Automation and Test in Europe Conference and Exhibition*, 1999, pp. 195–200.

[87] K. Chung, P.R. Shepherd, F. Eberhardt, and W. Tenten, "Hierarchical fault diagnosis of analog integrated circuits," *IEEE Transactions on Circuits and Systems-I: Fundamental Theory and Applications*, vol. 48, no. 8, pp. 921–929, 2001.

[88] S. Contu, A. Fanni, M. Marchesi, A. Montisci, and A. Serri, "Wavelet analysis for diagnostic problems," in *IEEE Mediterranean Electrotechnical Conference*, 1996, vol. 3.

[89] E. S. Erdogan and S. Ozev, "Single-measurement diagnostic test method for parametric faults of I/Q modulating RF transceivers," in *Proc. IEEE VLSI Test Symposium*, 2008, pp. 209–214.

[90] A. Robotycki and R. Zielonko, "Fault diagnosis of analog piecewise linear circuits based on homotopy," *IEEE Transactions on Instrumentation and Measurement*, vol. 51, no. 4, pp. 876–881, 2002.

[91] S. Yu, B. W. Jervis, K. R. Eckersall, I. M. Bell, A. G. Hall, and G. E. Taylor, "Neural network approach to fault diagnosis in CMOS opamps with gate oxide short faults," *Electronics Letters*, vol. 30, no. 9, pp. 695–696, 1994.

[92] B. W. Silverman, *Density Estimation for Statistics and Data Analysis*, Chapman & Hall/CRC, 1986.

[93] T. H. Lee, *The Design of CMOS Radio-Frequency Integrated Circuits*, Cambridge University Press, 2nd edition, 2004.

[94] A. Karatzoglou, A. Smola, K. Hornik, and A. Zeileis, "An S4 package for kernel methods in R," *J. Stat. Softw.*, vol. 11, no. 9, pp. 1–20, 2004.

[95] P.R. Gray, P.J. Hurst, S.H. Lewis, and R.G. Meyer, *Analysis and design of analog integrated circuits*, John Wiley & Sons, Inc., 4th edition, 2001.

[96] H.-G. Stratigopoulos, S. Mir, and A. Bounceur, "Evaluation of analog/RF test measurements at the design stage," *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, vol. 28, no. 4, pp. 582–590, 2009.

[97] J. E. Gentle, *Random Number Generation and Monte Carlo Methods*, Springer, 2nd edition, 2004.

[98] H. Hashempour, J. Dohmen, B. Tasic, B. Kruseman, C. Hora, M. Van Beurden, and Y. Xing, "Test time reduction in analogue/mixed-signal devices by defect oriented testing: An industrial example," in *Proc. Design, Automation & Test in Europe Conference*, 2011.

[99] A. Verikas, A. Lipnickas, K. Malmqvist, M. Bacauskiene, and A. Gelzinis, "Soft combination of neural classifiers: A comparative study," *Pattern Recognition Letters*, vol. 20, pp. 429–444, 1999.

[100] L.I. Kuncheva, ""Fuzzy" versus "nonfuzzy" in combining classifiers designed by boosting," *IEEE Transactions on Fuzzy Systems*, vol. 11, pp. 729–741, 2003.

[101] R.J.A. Little and D.B. Rubin, *Statistical Analysis with Missing data*, 2nd Edition, John Wiley & Sons, Inc, 2002.

[102] S.C. Bateman and W.H. Kao, "Simulation of an integrated design and test environment for mixed-signal integrated circuits," in *IEEE International Test Conference*, 1992, pp. 405–414.

[103] B. Webster, "An integrated analog test simulation environment," in *IEEE International Test Symposium*, 1989, pp. 567–571.

[104] K. Huang, H.-G. Stratigopoulos, and S. Mir, "Bayesian fault diagnosis of RF circuits using nonparametric density estimation," in *IEEE Asian Test Symposium*, 2010, pp. 295–298.

List of publications of the author

International journal papers

[1] K. Huang, H.-G. Stratigopoulos, S. Mir, C. Hora, Y. Xing and B. Kruseman. "Diagnosis of local spot defects in analog circuits", *IEEE Transactions on Instrumentation and Measurement* (Submitted paper).

International selected conference papers

[2] K. Huang, H.-G. Stratigopoulos and S. Mir. "Fault diagnosis of analog circuits based on machine learning", *In Proceedings of Design, Automation, and Test in Europe (DATE'10)*, 2010, pp. 1761-1766.
[3] K. Huang, H.-G. Stratigopoulos and S. Mir. "Bayesian fault diagnosis of RF circuits using nonparametric density estimation", *In Proceedings of Asian Test Symposium (ATS'10)*, 2010, pp. 295-298.

National selected conference papers

[4] K. Huang, H.-G. Stratigopoulos and S. Mir. "Diagnostic de fautes de circuits analogiques basé sur l'estimation non paramétrique de densité", *In 5e Colloque National du GDR SOC-SIP du CNRS, Lyon, France*, June 2011.
[5] K. Huang, H.-G. Stratigopoulos and S. Mir. "Diagnostic de fautes de circuits analogiques basé sur l'apprentissage automatique", *In 4e Colloque National du GDR SOC-SIP du CNRS, Cergy, France*, June 2010.

Fault modeling and diagnosis for nanometric mixed-signal/RF circuits

Abstract: Fault diagnosis of ICs has grown into a special field of interest in semiconductor industry. At the design stage, diagnosing the sources of failures in IC prototypes is very critical to reduce design iterations in order to meet the time-to-market goal. In a high-volume production environment, diagnosing the sources of failures can assist the designers in gathering information regarding the underlying failure mechanisms. In cases where the IC is part of a larger system that is safety critical (e.g. automotive, aerospace), it is important to identify the root-cause of failure and apply corrective actions that will prevent failure reoccurrence and, thereby, expand the safety features.
In this thesis, we have developed a methodology for fault modelling and fault diagnosis of analog/mixed circuits. A new approach has been proposed to diagnose both catastrophic and parametric faults based on machine learning. We then focused on spot defects which are more probable to occur in reality in order to develop an efficient diagnosis approach. The proposed diagnosis methodology has been demonstrated on data of failed devices provided by NXP Semiconductors in The Netherlands.

Keywords: Fault diagnosis, fault modeling, analog circuit testing, failure analysis, machine learning

---

Modélisation de fautes et diagnostic pour les circuits mixtes/RF nanométriques

Résumé: Le diagnostic de fautes est essentiel pour atteindre l'objectif de temps avant mise sur le marché (*time to market*) des premiers prototypes de circuits intégrés. Une autre application du diagnostic est dans l'environnement de production. Les informations de diagnostic sont très utiles pour les concepteurs de circuits afin d'améliorer la conception et ainsi augmenter le rendement de production. Dans le cas où le circuit est une partie d'un système d'importance critique pour la sûreté (e.g. automobile, aérospatial), il est important que les fabricants s'engagent à identifier la source d'une défaillance dans le cas d'un retour client pour ensuite améliorer l'environnement de production afin d'éviter la récurrence d'un tel défaut et donc améliorer la sûreté.
Dans le cadre de cette thèse, nous avons développé une méthodologie de modélisation et de diagnostic de fautes pour les circuits analogiques/mixtes. Une nouvelle approche basée sur l'apprentissage automatique a été proposée afin de considérer les fautes catastrophiques et paramétriques en même temps dans le diagnostic. Ensuite, nous avons focalisé sur le diagnostic de défauts de type *spot* qui sont considérés comme le mécanisme de défaut principal de circuits intégrés. Enfin, la méthodologie de diagnostic proposée a été validée par les données de circuits défectueux fournies par NXP Semiconductors aux Pays-Bas.

Mots clés: Diagnostic de fautes, modélisation de fautes, test analogique, analyse de défauts, apprentissage automatique

ISBN 978-2-84813-177-1