A Multi-Phase Diagnostic Flow Combining Scan-Based Refinement and Targeted Functional Tests for Automotive SoCs

IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 8 July 2026, accepted 22 July 2026, date of publication 30 July 2026, date of current version 5 August 2026.

Digital Object Identifier 10.1109/ACCESS.2026.3718146

APPLIED RESEARCH

A Multi-Phase Diagnostic Flow Combining Scan-Based Refinement and Targeted Functional Tests for Automotive SoCs

GIUSY IARIA[1], (Member, IEEE), GABRIELE FILIPPONI[1], (Member, IEEE), TOMMASO FOSCALE[1], (Graduate Student Member, IEEE), CLAUDIA BERTANI[2], GIUSEPPE GAROZZO[2], VINCENZO TANCORRE[2], AND PAOLO BERNARDI[1], (Senior Member, IEEE)

[1]Politecnico di Torino, 10129 Torino, Italy
[2]STMicroelectronics, 20864 Agrate Brianza, Italy

Corresponding author: Giusy Iaria (giusy.iaria@polito.it)

ABSTRACT Modern automotive systems increasingly rely on complex digital System-on-Chip (SoC) devices, where logic faults can severely compromise both performance and safety. Accurate post-silicon diagnosis is challenging because scan-based evidence may lead to large and ambiguous candidate sets, while purely functional validation often lacks the structural observability required to pinpoint the root cause. This study explores a multi-strategy diagnostic framework that leverages a *hybrid structural–functional diagnostic flow* based on *cross-domain correlation*. Scan-based diagnosis and iterative diagnostic patterns are used to identify and refine structural candidates; in turn, the resulting ranked candidates guide the development of targeted functional scenarios, whose outcomes are exploited to filter the initial candidate set. The framework is validated on a batch of real-world failed Automotive SoCs produced by STMicroelectronics, consistently reducing the candidate set across all testable devices in the batch and achieving accurate fault localization, as it is physically confirmed for an illustrative case study, by Physical Failure Analysis.

INDEX TERMS Logic diagnosis, silicon life-cycle management, logic built-in self-test, automotive, physical-failure-analysis.

I. INTRODUCTION

Modern Automotive systems have evolved from purely mechanical constructs to highly sophisticated electronic platforms [1]. A myriad of digital circuits, micro-controllers, sensors, and communication networks now form the backbone of vehicle operation, controlling everything from engine performance to Advanced Driver-Assistance Systems (ADAS) [2]. As these systems are becoming more and more complex and interconnected, ensuring their reliability and safety has emerged as a critical challenge [3], [4]. Defects within the digital logic circuits can lead to system-wide malfunctions, resulting in potential safety hazards [5], [6], [7].

This transformation in automotive design has been accompanied by an increased emphasis on rigorous diagnostic methodologies [1]. Conventional diagnostic approaches, such as standalone scan-based [8], have proven to be insufficient for the complex and often ambiguous responses generated by modern integrated circuits.

Standalone scan-based diagnosis can rapidly indicate the presence of a logic fault, yet it may generate large candidate sets that are difficult to be physically validated and expensive to debug [9], [10]. Conversely, functional testing can reproduce the failure mode under realistic operating conditions, but typically lacks the structural observability needed to backtrace the root cause at gate/flip-flop level [11].

The associate editor coordinating the review of this manuscript and approving it for publication was Ali Attaran.

Therefore, as automotive-grade devices become increasingly complex and deeply interconnected [12], [13], there is a growing need for a rigorous and scalable diagnostic flow that can systematically combine structural evidence and functional misbehavior to converge toward a small and physically meaningful set of candidates.

To tackle these challenges, this work dives into a multi-strategy diagnostic framework designed to integrate several complementary techniques to possibly reduce the final set of candidates, that will be then exploited for the Physical Failure Analysis (PFA). The framework is structured around three key phases (consistently referred to as Phase A, Phase B, and Phase C throughout the paper). Phase A focuses on scan-based detection and signature collection, Phase B performs scan-based refinement via iterative diagnostic patterns and candidate correlation, and Phase C executes targeted functional validation to further filter the structural candidates.

The first phase (Phase A) employs scan-based testing to generate rapid initial fault signatures. This testing methodology leverages an Automatic Testing Equipment (ATE) like approach to apply predefined scan patterns. Although this phase provides a rapid assessment, the resolution of the generated fault signatures may be limited by intrinsic noise and inherent ambiguities in the data.

The second phase (Phase B) is dedicated to the creation of diagnostic patterns and scan-based refinement. Early responses obtained during the scan-based tests in Phase A are analyzed in detail, and iterative diagnostic patterns are generated to target suspected fault regions with greater precision, thus improving the likelihood of correctly isolating defective areas.

The third phase (Phase C) involves targeted functional validation, which serves to further isolate and confirm the faults identified in the earlier phases. Functional testing subjects the automotive device to operating conditions that closely mimic real-world scenarios. This step provide validation that the suspected fault regions of previous phases indeed exhibit abnormal behavior under practical conditions. By correlating the results of the functional tests with the scan-based ranked candidates, the framework narrows down the specific areas that need an in-depth investigation.

The integration of multiple complementary diagnostic techniques not only addresses the weaknesses of individual methods but also creates a synergistic effect that results in a more accurate and smaller set of candidates. This study differs from traditional flows, where structural diagnosis and functional tests are treated as independent steps [14], [15], by applying a *hybrid structural–functional diagnostic flow* based on *cross-domain correlation* on real world failed Automotive devices. In particular, scan-based evidence (signatures, candidate lists, and structural backtracing) is used to develop focused functional tests, while functional outcomes are in turn exploited as a discriminant to filter structural candidates out. As such the proposed research may serve as a reference point for future diagnostic studies.

The key ideas of the applied study are:

1) A multi-step diagnostic strategy that integrates scan-based testing, and functional testing methods to reduce the total candidate set in complex Automotive SoCs.
2) Correlation of diverse diagnostic data sources, thereby refining the identification of defective regions and reducing diagnostic candidates.
3) Validation on real failed Automotive SoCs.

The effectiveness of the proposed methodology, however, remains dependent on several prerequisites, including: the scan-chain integrity [9], the fault coverage of the available LBIST data [16], and the functional excitability and observability of the scan-based candidates [14].

The paper continues as follows. Section II gives the reader an essential technical background and related state-of-the-art work. Section III describes the proposed methodology. Section IV reports the experimental results for a batch of failed Automotive devices produced by STMicroelectronics. Section V discusses the limitations of the approach and the conditions for its applicability. Finally, Section VI draws some conclusions.

II. BACKGROUND

This section provides a brief overview of the foundational aspects on which this work is based and discusses and compares related state-of-the-art work.

A. LOGIC DIAGNOSIS

In today's semiconductor industry, reducing yield loss during integrated circuit manufacturing remains a primary concern [8], [17], [18], [19]. As a result, significant research efforts have focused on understanding the relationship between defects and faults in digital circuits. Logic diagnosis is a critical process aimed at identifying and isolating the underlying causes that lead to erroneous behavior or failures in digital logic circuits [9], [20]. The objective of logic diagnosis is to precisely pinpoint the root causes of these errors.

The diagnostic process typically begins with the detection of abnormal circuit behavior or the observation of failing tests during functional or manufacturing evaluations. This is followed by a systematic analysis of circuit responses to various input stimuli, often using test patterns specifically tailored to target suspected faults. By carefully comparing observed outputs with expected results, logic diagnosis techniques can effectively identify faulty components, defective interconnections, or inherent design anomalies. Such insights are essential for improving circuit reliability and performance, as they support more efficient debugging and enable targeted repairs or optimizations.

B. PHYSICAL FAILURE ANALYSIS CHALLENGES

Physical Failure Analysis represents the ultimate ground-truth verification step in any diagnostic flow, providing independent confirmation of the root cause through direct physical inspection of the defective device [21]. However,

TABLE 1. Comparison of the proposed methodology with state-of-the-art logic diagnosis methodologies.

Comparison Axis	Holst et al. [27]	Pomeranz [28]	Gopalsamy et al. [17]	Riefert et al. [15]	Pomeranz [16]	Proposed Work
Iterative Patterns	× (FAST)	× (Multi-cycle)	√ (N-detection)	×	×	√ (Dynamic generation)
Integration of LBIST	×	×	√	×	×	√
Functional Validation Stage	×	×	×	√ (SBST programs)	√ (Sequence selection)	√
Structural–Functional Correlation	×	×	×	×	×	√
Validation on Real Devices	× (Benchmarks)	× (Benchmarks)	× (Benchmarks)	× (Simulated)	× (Benchmarks)	√ (Automotive SoCs)
PFA-Confirmed Outcome	×	×	×	×	×	√

PFA is inherently costly, time-consuming, and destructive, which imposes severe constraints on its practical applicability in industrial contexts [22].

A fundamental challenge in PFA is the progressive and irreversible nature of the de-layering process. As the device is physically sectioned layer by layer to expose the suspect region, the electrical state of the circuit cannot be re-examined once delayering is initiated [23]. This means that any misidentification of the target region prior to PFA results in a wasted device with no diagnostic gain. In automotive-grade SoCs, which routinely integrate tens of millions of gates across multiple metal layers and complex back-end-of-line (BEOL) stacks, the probability of targeting the wrong region increases dramatically when the structural candidate set is large or poorly ranked.

Beyond the technical challenges, PFA at the die level carries significant economic implications. The analysis requires specialized laboratory equipment, skilled failure analysis engineers, and substantial processing time, making exhaustive physical inspection of all diagnostic candidates practically infeasible in a production or field-return context. As highlighted in [23], die-level physical analysis always entails high risk, and the associated cost further limits the number of devices that can realistically undergo full PFA within an industrial failure analysis pipeline.

These constraints are compounded in the context of field-return diagnosis for automotive SoCs [24]. Field returns are typically available in limited quantities, and each device represents a unique diagnostic opportunity that cannot be replicated. Furthermore, as noted in the broader manufacturing RCA literature [25], the increasing complexity of modern production systems, characterized by large volumes of heterogeneous data, numerous interacting failure mechanisms, and the difficulty of distinguishing symptoms from actual root causes, makes the pre-PFA candidate reduction step a critical enabler of successful physical confirmation.

For these reasons, a diagnostic flow that can substantially reduce the candidate set and accurately identify the most probable suspect before PFA is highly valuable in practice. By focusing physical inspection on a narrowly defined region of interest, the flow minimizes the cost and risk associated with PFA and improves the probability of successful defect localization and confirmation. This capability is demonstrated through the case study presented in this work.

C. STATE-OF-THE-ART ABOUT LOGIC DIAGNOSIS METHODOLOGIES

In the state-of-the-art there are multiple studies that focus on logic diagnosis. The proposal described in [26] introduces a novel logic diagnosis technique specifically targeting Hidden Delay Defects (HDDs). This work addresses the considerable challenge of diagnosing HDDs in the presence of an extremely limited amount of incorrect response bits detected by Faster-than-At-Speed Testing (FAST). To overcome this difficulty, novel backtracing and response matching methods that take into account process variations, specifically developed for HDD diagnosis, are proposed. The importance of HDD diagnosis is highlighted in relation to the prevention of early failures and aging issues, which are particularly critical in autonomous and medical applications. The methodology employs a waveform-based timing simulation and presents a robust experimental design on large benchmark circuits through fault injection. The use of Faster-than-At-Speed Testing (FAST) and its integration into the diagnostic process are central elements, as is the introduction of the concept of MRD (Minimum Required Delay) back-propagation to define an initial set of potential HDD candidates.

The [27] paper suggests an alternative approach to improve logic diagnosis through the use of multi-cycle tests that use the same input data as fault detection tests. The focus is on increasing the volume of fault data directly from faulty units, circumventing the need to define specific target faults for the generation of additional diagnostic tests. Methods for offline and online selection of useful multi-cycle tests based on the number of erroneous output values produced are proposed. The paper also discusses how a logic diagnosis procedure can effectively utilize the additional fault data resulting from multi-cycle tests. Experimental results obtained on benchmark circuits demonstrate the potential for significant reductions in the number of candidate faults through the use of multi-cycle tests, especially in the context of multiple defects. It is also highlighted how multi-cycle tests can complement both conventional fault detection test sets and target fault-based diagnostic test sets.

The work described in [16] addresses the problem of using Logic Built-In Self-Test (LBIST) for logic diagnosis. The proposed storage-based LBIST scheme generates a diagnostic test set by augmenting the test data entries used for fault detection with additional subsets of scan vectors and permutations. Targets for diagnostic test generation

are derived by identifying faults that recur frequently in large candidate sets produced from simulated faulty units, and N-detection test generation is used to ensure sufficient fail data is collected for accurate diagnosis. Although the approach demonstrates that LBIST can be extended to produce diagnostic test sets with improved candidate-set resolution, the validation is limited to benchmark circuits and no functional correlation or physical confirmation of the results is performed.

Riefert et al. [14] propose a framework for the automatic generation of diagnostic Software-Based Self-Test (SBST) programs for processor cores, targeting stuck-at faults. The framework is based on Bounded Model Checking and incorporates a Validity Checker Module to enforce constraints specific to the SBST test environment, such as valid instruction sequences and memory coherence. For each pair of faults, the methodology proposes generating a diagnostic sequence capable of distinguishing between the two faults or demonstrating their equivalence within the imposed constraints, thereby enabling the calculation of the best achievable diagnostic coverage within a given test environment. The study demonstrates that it is possible to automatically generate functional test programs with high diagnostic efficiency. However, the methodology is designed specifically for processor cores with accessible instruction sets and focuses on stuck-at faults. Furthermore, the study does not address the broader challenge of validating the methodology on real-world case studies.

The work in [15] considers the problem of selecting functional test sequences from a simulation-based design verification pool for use in defect diagnosis. Rather than relying on diagnostic fault simulation, the selection procedure is driven directly by the results of defect diagnosis applied to available faulty units, identifying which sequences from the pool are most effective at reducing the number of candidate faults. The procedure is formulated as a set-covering problem and is shown to generalize effectively to faulty units not considered during selection. While this work demonstrates the value of functional test sequences for diagnosis, it relies on a pre-existing pool generated by design verification and does not address the development of targeted functional tests guided by structural scan-based candidates, which is the distinguishing mechanism of the proposed framework.

In this context, the proposed approach focuses on integrating scan-based testing, iterative diagnostic-based pattern generation, and functional testing. While [26] and [27] provide specific and significant contributions in particular areas of logic diagnosis, and while [14], [15], and [16] extend the diagnostic scope toward LBIST and functional test environments respectively, as summarized in Table 1, the proposed approach differs from these works in that structural scan-based candidates are used to drive the development of targeted functional tests. The primary objective of the proposed framework is to reduce the final set of fault candidates, with a key emphasis on the correlation of different sources of diagnostic data, a process that is essential to refine the identification of faulty regions and reduce the region of interest for PFA. Beyond these methodological aspects, the framework is validated on a batch of real failed automotive devices, providing the research community with diagnostic results from an industrial case study, including confirmation obtained through Physical Failure Analysis.

III. THE PROPOSED APPROACH

The diagnostic framework explored in this work offers multiple advantages that address the complexities inherent in modern automotive SoC diagnosis. First, it adopts a multi-stage diagnostic strategy that synergistically integrates scan-based testing for rapid fault identification, iterative diagnostic pattern generation to target suspicious regions more accurately, and functional testing to isolate and validate candidates. Second, the framework implements a practical methodology to correlate multiple sources of diagnostic data, allowing more accurate identification of fault regions and significantly reducing diagnostic ambiguities.

The framework operates under the assumption that each device is affected by a single fault. This premise underlies the candidate-set intersections and the cross-domain filtering across all three phases; its implications are discussed in Section V

Figure 1 provides an overview of the complete diagnostic flow, which is structured into three sequential and complementary phases. Each phase builds upon the results of the previous one, progressively narrowing down fault candidates from broad scan-based assessments to detailed failure characterization. The synergistic integration of these methodological phases constitutes the distinctive element of the proposed diagnostic framework, promising a more complete and effective approach to address the challenges posed by logic diagnosis in modern automotive SoCs.

The proposed diagnostic framework is divided into three sequential and complementary phases:

- **A.** ***Detection of faulty signatures***: This initial phase is based on the collection of data obtained from scan-based tests, providing a first indication of the possible fault locations within the digital circuit. The process includes a preliminary scan chain integrity test phase, a chip mapping phase using manufacturer-provided tests, an initial identification of the fault candidates using the ATPG, and a preliminary classification of the fault model (stuck-at or transition faults) based on the response to the clock frequency variation. The output of this phase is an initial candidate set $\mathcal{C}_0$.
- **B.** ***Refinement of fault locations***: The responses caught in the previous phase are analyzed in detail to derive additional scan-based diagnostic patterns, specifically targeting the candidate faults. Subsequently, an iterative diagnosis is performed to reduce the number of candidates, exploiting set theory techniques and candidate recurrence analysis. The output of this phase is a prioritized candidate set $\mathcal{C}^B_{\text{final}}$.

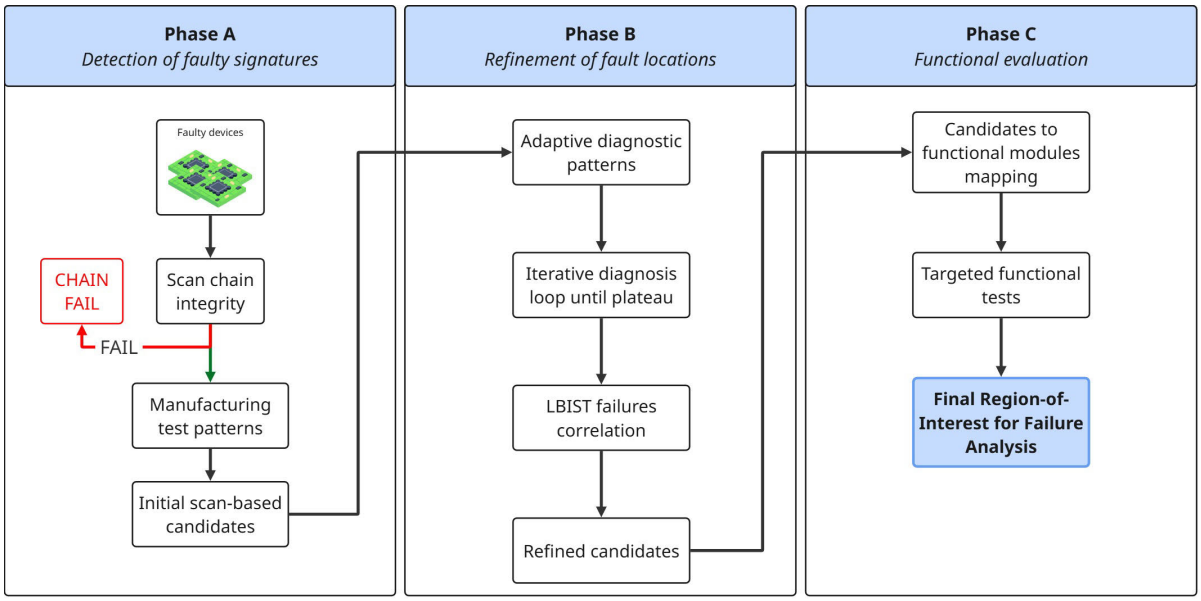


FIGURE 1. General overview of the proposed methodology.

C. ***Functional validation of candidate faults***: In the final phase, the Device-Under-Test (DUT) is subjected to a series of functional scenarios that emulate real-world operating conditions, with the aim of further validating and isolating previously detected anomalies. The data acquired during the functional tests are compared with the diagnostic results obtained in the previous phase. This phase acts as a crucial verification point, confirming and further refining the final candidate set for the device.

To make the interaction between structural evidence and functional behavior explicit, the proposed framework implements a closed-loop correlation mechanism between scan-based diagnosis and targeted functional programs. The key idea is to use scan-based candidates to guide the generation of functional programs, and then use the outcomes to further filter the structural candidates. This mechanism is formalized in Algorithm 1, which provides a step-by-step description of the complete diagnostic flow.

A. DETECTION OF FAULTY SIGNATURES

The first phase of the framework is designed to rapidly establish baseline fault signatures and validate the testability of the DUT. This phase leverages scan-based testing methodologies to collect preliminary diagnostic data that will guide subsequent analysis stages.

The process begins with a preliminary validation step that ensures scan chain integrity. If the device fails this preliminary step, it is discarded because it is not possible to continue with the diagnosis, as the integrity of the scan chain is a necessary premise for subsequent logic diagnostic steps.

Following, the complete scan-based pattern set used during the manufacturing production flow is applied to initially detect the faulty outputs from the devices.

Once the initial fault candidates are extracted, a classification step is performed to distinguish between different fault models. This procedure involves lowering the capture clock frequency to perform the capture phase at a frequency significantly lower than the nominal one. If the DUT continues to fail, it indicates a Stuck-At Fault (SAF), otherwise a Transition Delay Fault (TDF).

A SAF manifests independently of operating conditions, as it represents a permanent logical state corruption. A TDF, conversely, is attributable to timing-dependent physical mechanisms, such as resistive interconnect degradation or localized metallization narrowing, that produce incorrect responses only under tight timing margins. This step is significant, since it allows a first classification of the devices based on the type of failure.

At the conclusion of Phase A, the framework has established the initial candidate set $\mathcal{C}_0$ along with a fault model label for each testable DUT. This output serves as the critical input for the refinement strategies employed in Phase B. Furthermore, it directly conditions the functional test strategy adopted in Phase C, limiting the creation or use of specific functional tests to target specific faults.

B. REFINEMENT OF FAULT LOCATIONS

After the initial candidates are identified, Phase B is dedicated to refining the diagnostic results through multiple complementary strategies. This phase recognizes that the initial diagnosis, while rapid, may produce an excessive number of candidates or contain ambiguities that make physical validation impractical. The goal of this phase is to adaptively generate new diagnostic patterns and apply various analytical techniques to converge on a highly probable set of fault candidates.

Algorithm 1 Multi-Step Logic Diagnosis Flow

Require: T_{scan}: Failing scan tests T_{scan}.
Require: S: scan signatures/logs S.
Require: Design database (netlist + hierarchy + layout).
Require: R_{LBIST}: Optional LBIST report.
Ensure: Filtered candidate set C_{final} and region-of-interest for failure analysis ROI.

▷ **Phase A**
if not scan_chain_integrity_pass() **then**
  return
end if
$C \leftarrow \text{Diagnose}(T_{\text{scan}}, S)$ ▷ Initial candidates (C_0)
$k \leftarrow 0$
▷ **Phase B**
do
  $k \leftarrow k + 1$
  $cnt \leftarrow |C|$
  $C \leftarrow \text{IterativeRefine}(C)$ ▷ yields C_k^{SCAN}
while $cnt \neq |C|$
if R_{LBIST} is available **then**
  $C_* \leftarrow C \cap R_{\text{LBIST}}$ ▷ high-confidence overlap
  if $|C_*| \neq 0$ **then**
    $C \leftarrow C_*$
  end if
end if
$C \leftarrow \textit{recurrence\_analysis}(C)$ ▷ order C_k^{SCAN} or C_* by ρ; no filtering
▷ **Phase C, at this phase C = $\mathbf{C^B_{final}}$**
$H \leftarrow \text{MapToHierarchy}(C)$ ▷ map candidates to modules/clock domains/IOs
$F \leftarrow \text{GenerateFunctionalScenarios}(H)$ ▷ scenarios targeting suspected subsystems
$O \leftarrow \text{RunFunctionalTests}(F)$ ▷ collect functional outcomes/signatures
$C_{\text{final}} \leftarrow \text{FilterCandidates}(C, O)$ ▷ retain only $O(c) =$ fail candidates (single-fault assumption)
ROI $\leftarrow \text{ProjectOnLayout}(C_{\text{final}})$

The initial candidate set $\mathcal{C}_0$ may be too large to directly enable reliable functional or physical analysis. It is also prohibitive in terms of time to perform functional analyses due to the large number of candidates obtained from the initial diagnosis. Iterative diagnosis involves a cycle of diagnostic patterns application on the device. At each iteration k, a new set of diagnostic patterns is generated using the failure signatures and candidate set $\mathcal{C}_{k-1}$ from the previous step, producing a refined candidate set $\mathcal{C}_k \subseteq \mathcal{C}_{k-1}$. The process continues until it reaches a plateau, defined as $|\mathcal{C}_k| = |\mathcal{C}_{k-1}|$. The intent is to converge toward a limited set of high-confidence candidates that can be practically validated with cross-domain evidence.

To complement and accelerate the iterative diagnosis process, some auxiliary techniques are introduced to reduce the candidate set and rank the remaining candidates by priority, enabling the subsequent functional validation phase to focus on the most probable candidates by following the ranking.

1) SET-THEORETIC LBIST CORRELATION

Let $\mathcal{C}_K^{\text{SCAN}}$ denote the candidate set produced by the iterative scan-based diagnosis loop after k iterations, and let $\mathcal{C}^{\text{LBIST}}$ denote the candidate set independently derived from the LBIST signatures available for the DUT. The LBIST-correlated candidate set is defined as:

$$\mathcal{C}^* = \mathcal{C}_K^{\text{SCAN}} \cap \mathcal{C}^{\text{LBIST}} \tag{1}$$

Candidates in $\mathcal{C}^*$ are assigned the highest priority rank, as their presence in both structurally independent diagnostic sources makes them the most likely to justify physical investigation and queues them earliest for functional and physical follow-up. Although LBIST approaches are known to have lower coverage than the ATPG and typically cover only a portion of the circuit, the candidate sets generated by LBIST allow to highlight candidates that should be considered priorities for identifying the real candidates for the fault observed in the DUT. Consequently, $\mathcal{C}^*$ may be empty for some devices, in which case this filtering step is bypassed and the full set $\mathcal{C}_K^{\text{SCAN}}$ is forwarded to the subsequent techniques.

2) RECURRENCE ANALYSIS

Let $\{\mathcal{C}_i\}_{i=1}^n$ be the collection of candidate sets produced for the DUT by the different diagnostic analyses considered during the flow, namely the individual scan diagnostic pattern sets and, where available, the LBIST and functional analyses, with $\mathcal{C}_i \subseteq \mathcal{F}$ and $\mathcal{F}$ the global fault space of the circuit. The recurrence score of a candidate $c \in \bigcup_{i=1}^n \mathcal{C}_i$ is defined as the number of these candidate sets in which c appears:

$$\rho(c) = |\{i : c \in \mathcal{C}_i\}| \tag{2}$$

A high value of $\rho(c)$ indicates that c is consistently identified across independent diagnostic analyses, a property expected of genuine physical defects, particularly those associated with the stuck-at fault model. The recurrence score is a pure occurrence count, employed exclusively to prioritize candidates by sorting them in descending order of $\rho(c)$ so that the most recurrent candidates are investigated first by the following phases. The score does not, by itself, add or remove any candidate; the candidate set is reduced only by the iterative scan diagnosis, by the LBIST intersection of (1), and by the functional tests of (3). The same score determines the ordering of the set $\mathcal{C}^B_{\text{final}}$ forwarded to Phase C.

At the conclusion of Phase B, the framework produces a refined and prioritized candidate set $\mathcal{C}^B_{\text{final}}$, namely the set $\mathcal{C}_K^{\text{SCAN}}$ (or $\mathcal{C}^*$ where available) ordered by the recurrence score $\rho(c)$. The number of candidates has been substantially reduced through the combination of iterative diagnosis and LBIST correlation. The candidates are ordered by descending recurrence, making the most probable defects the first targets for the final validation phase.

C. FUNCTIONAL VALIDATION OF CANDIDATE FAULTS

Phase C operationalizes the cross-domain correlation that constitutes the core methodological idea of this work. Given the ranked candidate set $\mathcal{C}^B_{\text{final}}$ produced by Phase B, this phase pursues two complementary objectives: (i) to possibly verify that the identified structural candidates manifest observable functional misbehavior under realistic operating conditions, thereby filtering candidates that are functionally untestable; and (ii) to use functional failure signatures as an additional discriminant to filter the structural candidates, producing a final candidate set $\mathcal{C}_{\text{final}}$ suitable for PFA. The rationale for this filtering is that a candidate confirmed to be functionally excitable and observable under the applied tests, established via fault simulation, but not producing the corresponding erroneous signature on real silicon, constitutes negative evidence against that candidate: if it had been the real defect, the applied functional tests should have revealed it. This reasoning applies uniformly to both stuck-at and transition-delay fault candidates. For stuck-at faults it is immediate, as their manifestation is independent of operating conditions. For transition-delay faults it remains valid because functional coverage is assessed for the specific delay that would trigger the misbehavior, and the screening is performed at the operating frequency at which Phase A has determined that the device manifests the fault.

Based on the same reasoning, a candidate that is covered under these timing conditions by functional tests and that subsequently produces no erroneous responses on the silicon is removed from the final set of candidates.

The phase proceeds through three formally defined steps.

1) CANDIDATE-TO-MODULE MAPPING

Each candidate $c \in \mathcal{C}^B_{\text{final}}$ is mapped to its enclosing functional sub-module through a hierarchy traversal of the design netlist. Let $\mathcal{M} : \mathcal{C}^B_{\text{final}} \rightarrow \mathcal{H}$ denote this mapping, where $\mathcal{H}$ is the set of functional sub-modules of the design. The resulting set of target sub-modules $H = \mathcal{M}(\mathcal{C}^B_{\text{final}})$ identifies the regions in the circuit that need to be excited by the functional tests campaign.

2) FUNCTIONAL TESTS DEVELOPMENT AND EXECUTION

For each sub-module $h \in H$, a set of targeted functional tests $\mathcal{T}_h$ is synthesized to test h, informed by the fault model label assigned in Phase A. The union $\mathcal{T}_{\text{func}} = \bigcup_{h \in H} \mathcal{T}_h$ represents the functional test for the DUT. Before application on silicon, fault simulation of $\mathcal{T}_{\text{func}}$ is performed to determine, for each candidate c, whether it is excited and its effect propagated to an observable point. Let $O(c)$ be the functional test result (*pass* or *fail*) observed for the candidate c when $\mathcal{T}_{\mathcal{M}(c)}$ is applied, which can be either ''fail'' or ''pass'' based on test outcome.

3) CROSS-DOMAIN FILTERING

The functional outcomes are used to filter the structural candidates, yielding the final candidate set:

$$\mathcal{C}_{\text{final}} = \left\{ c \in \mathcal{C}^B_{\text{final}} : O(c) = \text{fail} \right\} \tag{3}$$

That is, candidates for which no functional misbehavior is observed under the considered functional tests are discarded. Under the single-fault assumption, if at least one candidate produces a functional failure ($O(c) = \text{fail}$), the real defect must be among the failing candidates; consequently, candidates that are not covered by any functional test are also excluded from $\mathcal{C}_{\text{final}}$, as the defect has already been localized to a functionally confirmed candidate.

$$\text{ROI} = \text{ProjectOnLayout}(\mathcal{C}_{\text{final}}) \tag{4}$$

This closed-loop interaction between structural and functional evidence, where scan-based candidates guide the development of functional tests, and functional outcomes in turn refine the structural candidate set, is the distinguishing element of the proposed framework with respect to conventional diagnostic flows, in which structural diagnosis and functional validation are treated as independent sequential steps with no feedback between them.

IV. EXPERIMENTAL RESULTS

This section reports the experimental validation of the proposed framework on a batch of 17 failed devices produced by STMicroelectronics.

The section is organized as follows: (i) an overview across the full batch of DUTs, reporting the diagnostic data and candidate-set reduction achieved by Phase A, Phase B, and Phase C across all faulty devices; (ii) a detailed *zoom-in* on an illustrative device (DUT#13), where the design of the targeted functional test campaign is described step by step, together with the *cross-domain filtering* logic used to discard candidates that do not manifest functional misbehavior and to correlate the surviving evidence back with the scan-based results; (iii) the Physical Failure Analysis performed on the same illustrative DUT#13, providing an independent confirmation of the identified root cause.

A. EXPERIMENTAL SETUP

The case studies focus on a batch of 17 failed devices that are intended for automotive provided by STMicroelectronics. The SoC has the following specifications:

- About 20 million gates;
- About 700 thousand flip-flops;
- Multi-core architecture;
- ASIL-D compliant;
- 7 LBIST partitions.

The complete fault list size of the case of study is composed of about 80 million SAF and TDF.

The setup comprehends both hardware and software. The hardware side is composed of three different boards, each one with a dedicated purpose, as shown in Figure 2. Going from left to right:

- The tester: a prototype board provided by STMicroelectronics. It is equipped with a Zynq Ultrascale+ MPSoC FPGA, which implements all the necessary mechanisms to interact with the software to start tests and collect signatures;

- The bridge interface: a prototype board designed to act as a bridge between the tester and the DUT. Moreover it equips level-shifter capabilities to handle different voltages requirements between the tester and DUT;
- The DUT: the SoC requires a support board that allows access to its pads. For this reason, two boards are used: the daughterboard, which exposes some of the chip's pads via a JTAG port and provides core voltage through a dedicated port, and the motherboard, which in addition to providing power to the rest of the DUT, exposes the scan chain pins via male connectors.

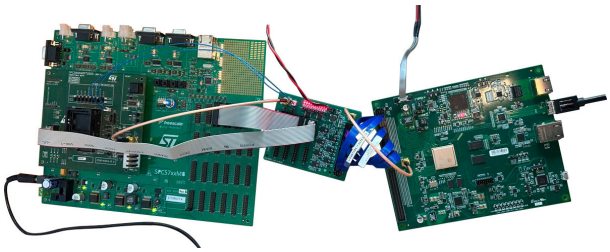


FIGURE 2. Experimental setup hardware boards.

The software part is managed by the tool *Rhythm* made by *Roguevation* [28], which allows to control any semiconductor test platform. In this scope, this software tool is used to parse the STIL files, execute the diagnosis patterns on the DUTs and collect the related responses.

All the experiments regarding the diagnostic pattern generation and fault simulation have been executed on a high-performance multi-processor server equipped with a 64-bit 16-core processor AMD EPYC 7301, 256GB of RAM, a storage system of 10 TBytes, and a Centos Linux 7 operating system by using *TestKompress* from the Tessent-ATPG suite, made by *Siemens*.

B. MULTI-PHASE DIAGNOSIS ON FAULTY DEVICES

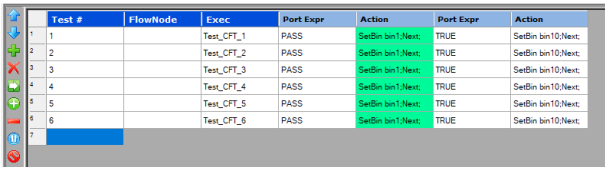
This section reports the results of the complete three-phase diagnostic flow applied to the full batch of 17 DUTs. For each testable device, Phases A, B, and C were executed sequentially. Table 2 summarizes the candidate counts at each step of the flow for all the DUTs.

1) PHASE A: PRELIMINARY TEST DIAGNOSIS AND INITIAL CANDIDATES IDENTIFICATION

This phase is essential to verify the correct functioning of the scan chains.

In the reported case of studies, 6 different chain integrity patterns were used, applied one at a time to the 17 DUTs. The time required for each of the chain integrity test amounts to 1.3 s for data upload to the Rhythm tool, and 3.5 s for data processing, upload and retrieval, yielding $\approx$ 5 s for every pattern. Figure 3 reports a screenshot from Rhythm showing how one of the DUTs successfully passed the chain integrity tests, confirming the functionality of the scan chains.

Once the preliminary verification is completed, if tests passed, the initial candidate identification is performed.



	Test #	FlowNode	Exec	Port Expr	Action	Port Expr	Action
1	1		Test_CFT_1	PASS	SetBin bin1;Next;	TRUE	SetBin bin10;Next;
2	2		Test_CFT_2	PASS	SetBin bin1;Next;	TRUE	SetBin bin10;Next;
3	3		Test_CFT_3	PASS	SetBin bin1;Next;	TRUE	SetBin bin10;Next;
4	4		Test_CFT_4	PASS	SetBin bin1;Next;	TRUE	SetBin bin10;Next;
5	5		Test_CFT_5	PASS	SetBin bin1;Next;	TRUE	SetBin bin10;Next;
6	6		Test_CFT_6	PASS	SetBin bin1;Next;	TRUE	SetBin bin10;Next;
7							

FIGURE 3. Screenshot of Rhythm tool. Details view of Test Scheduler Tab, that shows the successfully passed scan chain integrity tests for one of the 17 STMicroelectronics' DUTs.

In this phase a combination of patterns provided by STMicroelectronics, the Rhythm tool, and the simulation environment, yield the initial candidate set. More in detail, each DUT is tested through the production scan-based patterns, and the outcome (pass or fail), and as well corresponding signatures, are recorded. In case of a test failure, the software saves the position of the bits involved and the pattern that generated the discrepancy compared to the expected response described in the pattern file.

For each pattern set when a failures occur, a screening verification by adjusting the frequency parameters to minimize the capture frequency, is performed. This step is intended to differentiate between SAF and TDF.

This phase is particularly time-consuming, as each DUT must be run through several pattern sets. The time required to complete the diagnostic tests of a single chip is ≈ 200 s.

The generated reports are then processed by a suite of dedicated programs to create the failure files. These files, along with the flat-model, STIL files and layout database, are used to perform the diagnostic operation, that is backtracking the failure response to a set of possible candidates.

2) PHASE B: CANDIDATE ITERATIVE DIAGNOSIS

The iterative diagnosis represents a crucial process to further refine the candidate set obtained in the previous phases. Each iteration generates new patterns, designed to better characterize the candidates. This loop lasts until the number of candidates remains constant between successive iterations. That is after 4 iterations in the reported case of studies.

3) PHASE C: FUNCTIONAL VALIDATION

Upon completion of Phase B, Phase C was applied to all 14 testable DUTs. For each device, the ranked candidate set $\mathcal{C}^B_{\text{final}}$ was mapped to its enclosing functional sub-modules, and targeted functional tests scenarios were synthesized and executed accordingly. The functional tests results were used to exclude candidates that, although structurally plausible, did not produce any observable functional effect, even when excited, as described in Section III.

Table 2 shows the complete trend in the number of candidates for each DUT across all phases. Across the batch, Phase C achieved an average additional candidate reduction of 2.94× over the Phase B output. Since each DUT required dedicated functional test development, one illustrative case study, DUT#13, is discussed in Section IV-E, detailing how its functional tests were generated to target the candidates.

TABLE 2. Summary of candidate counts across all phases of the diagnostic flow for each device.

DUT	Phase A		Phase B						Phase C
			Iterative Diagnosis				Intersection with LBIST		
	SCAN CHAIN INTEGRITY PASS	#CDT Initial	1^{st} Iter.	2^{nd} Iter.	3^{rd} Iter.	4^{th} Iter.	#CDT LBIST	#CDT SCAN $\bigwedge$ LBIST	#CDT After Funct. Screening
0	√	12407	7843	2248	1090	565	11764	468	197
1	√	9784	5700	510	195	113	-	113	43
2	×	-	-	-	-	-	-	-	-
3	√	17103	9996	3924	1404	1179	7948	834	342
4	×	-	-	-	-	-	-	-	-
5	√	7532	4601	1444	587	566	-	566	243
6	×	-	-	-	-	-	-	-	-
7	√	4948	4901	1003	227	227	13553	111	43
8	√	6425	3765	1002	235	129	-	129	58
9	√	2394	1095	641	333	95	-	95	35
10	√	4696	3154	1337	1055	332	-	332	139
11	√	10028	6015	1403	428	348	-	348	153
12	√	2988	2330	778	384	384	-	384	138
13	√	138	121	89	53	14	2200	9	1
14	√	10280	5727	2034	1137	640	44320	555	228
15	√	1132	907	246	115	115	-	115	50
16	√	9457	5963	1592	1565	1126	15990	892	339

TABLE 3. Quantitative metrics across all phases of the diagnostic flow for each testable device. R_{scan}: candidate reduction factor after Phase B (scan-only); R_{LBIST}: candidate reduction factor after LBIST intersection (where available, '-' otherwise); $R_{\text{Phase C}}$: additional candidate reduction contributed by Phase C. Runtimes are reported in hours for Phase B and seconds for Phase C. The column "Patterns generated (%)" reports the number of iterative diagnostic patterns generated during Phase B as a percentage of the production scan pattern set applied in Phase A.

DUT	Phase B				Phase C		
	Patterns generation time (h)	Patterns generated (%)	$R_{\text{scan}}(\times)$	$R_{\text{LBIST}}(\times)$	Development time (h)	#CDT after screening	$R_{\text{Phase C}}(\times)$
0	10.25	7.29	21.96	26.51	5	197	2.37
1	1.89	1.37	86.58	-	4	43	2.63
3	13.55	9.08	14.51	20.51	6	342	2.44
5	8.41	5.98	13.31	-	5	243	2.33
7	3.21	2.13	21.80	44.58	4	43	2.58
8	7.82	5.19	49.81	-	4	58	2.22
9	1.60	1.14	25.20	-	4	35	2.71
10	3.70	2.63	14.14	-	5	139	2.39
11	4.28	2.84	28.82	-	5	153	2.27
12	17.82	12.42	7.78	-	5	138	2.78
13	0.17	0.11	9.86	15.33	2	1	9.00
14	5.71	3.82	16.06	18.52	6	228	2.43
15	2.80	1.99	9.84	-	4	50	2.30
16	11.66	8.29	8.40	10.60	6	339	2.63

C. QUANTITATIVE METRICS

To allow a comprehensive cost-benefit assessment of the proposed framework, the candidate reduction factor at each phase is defined as:

$$R_{\text{phase}} = \frac{|\mathcal{C}_{\text{in}}|}{|\mathcal{C}_{\text{out}}|} \tag{5}$$

where $\mathcal{C}_{\text{in}}$ and $\mathcal{C}_{\text{out}}$ denote the candidate sets at the input and output of the considered phase, respectively. Table 3 reports the quantitative metrics across all phases of the diagnostic flow for the 14 testable DUTs. Table 4 summarizes the average values across the batch, enabling a direct comparison of the contribution of each phase and of the full flow against scan-only and LBIST-only baselines.

TABLE 4. Average quantitative metrics across all 14 testable DUTs. Phase A values are identical across all devices. R^*_{LBIST} is averaged over the 6 DUTs for which LBIST data was available.

Phase	Metric	Value
A	Chain integrity test time (per pattern)	$\approx 5\ s$
	Full pattern application time (per DUT)	$\approx 200\ s$
	Pattern sets applied (per DUT)	57
B	Runtime per DUT (h)	6–18
	Number of iterations to plateau	4
	R_{scan} — scan-only baseline (×)	23.4
	R^*_{LBIST} — LBIST intersection (×)	22.7
C	Development Time [h]	4.64
	Additional reduction from Phase C (×)	2.94
Full Flow	R_{full} — complete flow (×)	72.88

D. LAYOUT VIEW OF ROOT CAUSE OF FAILURES

The graphical representation of the candidates over the layout is key to highlight the critical areas identified as possible failure candidates. This visual approach, facilitates the identification of correlations between the observed malfunctions and the responsible elements. Figure 4 illustrates this three-way correlation on DUT#3, chosen here purely as a general layout example ahead of the detailed DUT#13 zoom-in of Section IV-E: the scan-based candidates (left) are spatially scattered across the die, the LBIST-derived candidates (center) independently cover a partially overlapping

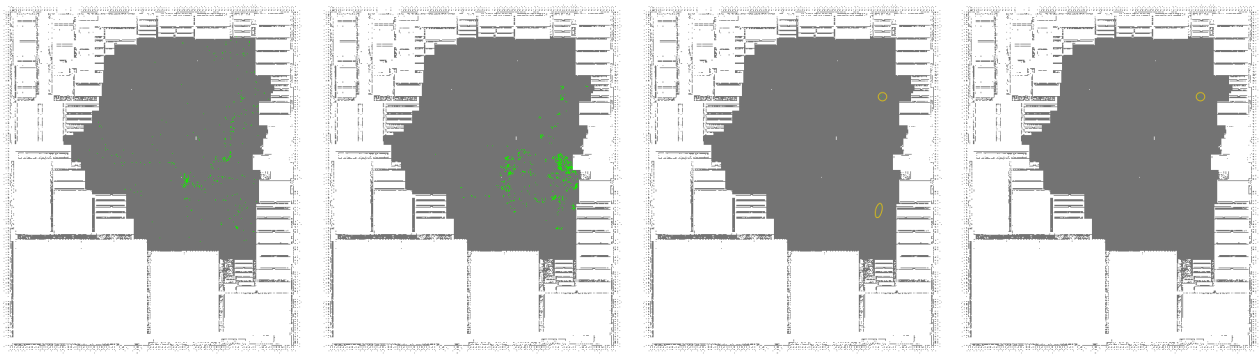


FIGURE 4. Layout views of the DUT#3. Going left to right, first the plot of the candidates given as a result of the diagnosis for one DUT. The second image represent the LBIST candidates. The third image shown the intersection between the scan and LBIST methodology; Finally on the right the Phase C outcome after functional screening.

region, and their intersection (right) collapses to a markedly tighter spatial cluster.

This spatial concentration is the concrete mechanism by which cross-domain correlation reduces PFA cost: since the de-layering area, and therefore the physical-analysis effort discussed in Section II, is governed by the spatial extent of the surviving candidates rather than by their raw count, an intersection that preferentially removes spatially scattered candidates yields a reduction in the region-of-interest that is steeper than the reduction in candidate count alone.

E. DUT#13: ZOOM IN

Among the devices in the batch, DUT#13 is selected as the illustrative case study for a step-by-step demonstration and discussion in details of the cross-domain correlation strategy.

1) FUNCTIONAL PROGRAMS GENERATION

Its candidate set after Phase B converged to 9 candidates, all confined to a single sub-module of the Controller-Area-Network (CAN). Table 5 summarizes the candidate-set reduction along the cross-domain flow for this device. Of the 9 candidates identified by Phase A and Phase B, 2 are functionally testable through the CCU calibration routine and are therefore considered for Phase C; this indicates which candidates can be functionally excited, but does not in itself exclude any candidate from the complete diagnostic set yet.

Since the functional program covers one of the faults, and the program generated a failure signature on the actual silicon, the framework identified one candidate as the target for the PFA. It is important to mention that this works under the assumption of a single fault per DUT, that is at the basis of the proposed flow.

Going more in detail, the sub-module corresponds to the Clock-Control-Unit (CCU) of the CAN. This module enables synchronization of nodes over the bus with limited information on the baud rate used on the channel. The CCU operates as an add-on to the CAN module, generating a calibrated CAN Time Quanta Clock from an on-chip RC oscillator by evaluating CAN messages received by the attached CAN. The experimental setup involved a companion

TABLE 5. Candidate-set reduction for DUT#13 along the cross-domain flow. The functionally testable step marks the subset of candidates that can be excited; the final step retains the one candidate that reproduces the observed silicon failure with functional tests and is taken to PFA.

Step	#CDT	Evidence / Criterion
Phase B - Scan-based diagnosis	9	Iterative scan diagnosis convergence (Table 2)
Phase C - Functionally testable subset	2 of 9	Scan-based candidates excitable and observable via the ad-hoc functional patterns (CCU calibration routine)
Phase C - Functionally confirmed (PFA target)	1	Cross-domain correlation of the functional failure signature with the scan-based candidates.

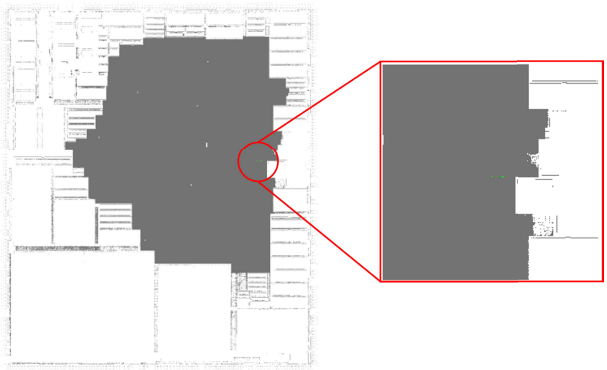


FIGURE 5. Candidates of DUT#13, highlighted in green, spread on the CAN subsystem.

module [29] transmitting a sequence of frames with leading bits, while the DUT board received these frames and calibrated accordingly.

The functional flow was implemented as a lightweight calibration routine, summarized in the following pseudo-code:

Silicon-level validation confirmed that the CCU maintained calibration integrity across 50 consecutive runs, with signatures matching the reference simulation pattern. Testing

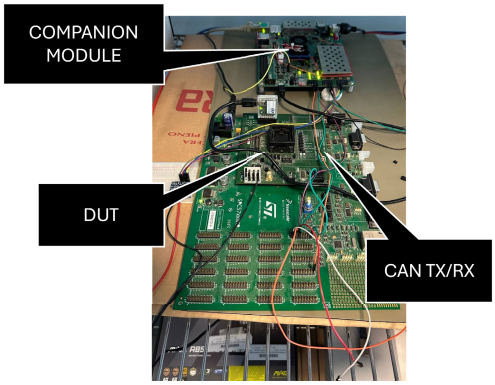


FIGURE 6. Experimental setup: companion module transmitting calibration frames to DUT.

Algorithm 2 Functional Routine of the CCU for DUT#13

Require: CCU initialized and companion module available.
Ensure: Calibration sequence and IRQ execution.

```
initialize_CCU()
for i ← 0to8 do
    send_frame(companion_module)
    wait_for_IRQ()
    if calibration_complete() then
        update_signature()
    end if
end for
reset_FSM_if_needed()
verify_signature()
```

under varying clock conditions revealed that the CCU successfully operated at up to 200 *MHz* when the host clock was limited to 100 *MHz*. However, failures occurred when the host clock exceeded 200 *MHz*, verifying the transition candidates yielded by the proposed approach.

2) PHYSICAL FAILURE ANALYSIS AND VALIDATION OF THE DIAGNOSTIC OUTCOME

To provide an independent ground truth and definitively confirm the root cause, Physical Failure Analysis (PFA) was performed on DUT#13. By analyzing the functional testability of the candidates and correlating them with the failures observed during the functional test application, the proposed flow successfully filtered a single candidate as the most probable candidate, so that the physical inspection could be explicitly targeted to this localized region.

As shown in Figure 7, the targeted PFA revealed a localized, scratch-like defect within the Back-End-of-Line (BEOL) metallization. This defect overlays the specific standard cells responsible for the CCU's interrupt request (IRQ) handling. This physical evidence directly corroborates the functional verification results: the calibration routine (Algorithm 2) heavily relies on the `wait_for_IRQ()` operation, which failed to execute correctly under operative conditions.

Independent evaluation confirmed this anomaly is a genuine, process-induced defect rather than a packaging or sample-preparation artifact. A BEOL scratch of this nature typically weakens the inter-metal dielectric (IMD) or induces localized metal displacement, creating resistive leakage paths between adjacent nets. This physical failure mechanism perfectly explains the frequency-dependent behavior observed during testing, validating why the transition fault only manifested at capture clocks exceeding 200 *MHz*.

While the required de-layering removed the upper oxide layer, preventing further cross-sectional analysis of the exact microscopic leakage path, the morphological evidence remains conclusive. This demonstrates that the proposed cross-domain methodology managed to reduce the candidate set size, and helped in the final fault localization.

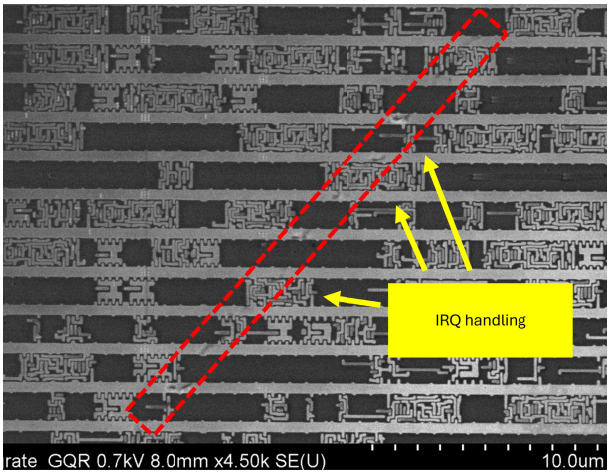


FIGURE 7. Physical Failure Analysis in the area of the target candidate. The red outline highlights a scratch-like defect over the cells. The yellow boxes denote the candidates responsible for IRQ handling, physically confirming the failure.

V. LIMITATIONS AND APPLICABILITY

The applied methodology combines scan-based diagnosis, LBIST data, and targeted functional tests to progressively reduce the candidate set and possibly improve the final region of the defect. Nevertheless, some practical limitations must be considered.

First, the methodology starts from the assumption that a single fault interests the circuit. If not, the intersections made across all the phases about the candidates cannot be performed and would not be valid since in case of multiple faults it is not possible to reduce candidates with the proposed diagnostic flow.

Second, the methodology assumes the availability of reliable scan chain. Scan-chain integrity failures may compromise the collection of diagnostic data and prevent the generation of meaningful scan-based candidates. In such situations, neither scan diagnosis nor the subsequent cross-correlation stages can be applied.

Third, although LBIST information can provide valuable complementary evidence and help prioritize candidate locations, its usefulness is constrained by the fault coverage

of the implemented LBIST architecture. Certain faults may remain undetected through LBIST signatures, limiting its contribution to the overall diagnosis process.

Moreover, the functional validation stage introduces additional constraints. Functional confirmation is only possible when the candidate fault can be both excited and observed through functional execution. Consequently, some scan-diagnosed candidates may not be accessible to functional validation due to limited controllability or observability at the system level.

In addition, the development of dedicated functional routines often requires significant engineering effort [30], [31]. Test sequences specific for the candidates must be created, integrated, and verified to ensure proper fault activation and observation. As a result, the cost of functional validation scales with both the number of candidates and the complexity of the affected design regions.

Moreover, each functional program iteration necessitates of fault simulation to assess candidate coverage from previous phases. These fault-simulation campaigns can introduce substantial computational overhead, particularly for large industrial designs. The challenge is further amplified when targeting transition-delay faults. Unlike static fault models such as SAF, TDF require timing-sensitive activation and observation conditions that are often more difficult to reproduce under functional operation. Consequently, achieving sufficient coverage for TDF validation may require additional simulation effort and carefully crafted test scenarios.

Finally, the scalability of the functional validation stage can become a limiting factor when scan-based diagnosis produces candidates distributed across multiple and potentially unrelated design regions. In these cases, dedicated functional routines may be required for several modules, each demanding separate analysis, implementation, and verification. The associated increase in fault-simulation complexity and engineering effort may reduce the practicality of exhaustive functional validation. Nevertheless, even under these conditions, the methodology remains valuable as a candidate-ranking and evidence-correlation framework, enabling a more informed diagnosis process than any individual technique used in isolation.

VI. CONCLUSION

The work presented in this paper introduces a multi-strategy diagnostic framework that combines scan-based diagnosis and diagnostic pattern generation with targeted functional validation, enabling a *hybrid structural–functional diagnostic flow* based on *cross-domain correlation*.

This approach addresses the challenges inherent in logic diagnosis by possibly reducing the final diagnostic candidate set. Experimental validation on real failed automotive SoCs shows that the proposed flow reduced the final candidate set on all testable devices considered, by correlating structural evidence (scan signatures and ranked candidates) with functional outcomes. The resulting localization was independently confirmed by PFA on an illustrative device, while the corresponding confirmation on the further devices analyzed cannot be disclosed here.

Overall, the integration of scan-based refinement and targeted functional validation establishes a practical flow that, under the limitations and applicability scenarios described, can be applied to modern automotive electronic systems.

REFERENCES

[1] U. Abelein, H. Lochner, D. Hahn, and S. Straube, "Complexity, quality and robustness—The challenges of tomorrow's automotive electronics," in *Proc. Design, Autom. Test Eur. Conf. Exhib. (DATE)*, Mar. 2012, pp. 870–871.

[2] R. Srivastava, N. Mudgil, G. Gupta, and H. Mondal, "SoC time to market improvement through device driver reuse: An industrial experience," in *Proc. Int. Symp. Electron. Syst. Design (ISED)*, Dec. 2012, pp. 56–61.

[3] F. Angione et al., "Test, reliability and functional safety trends for automotive system-on-chip," in *Proc. IEEE Eur. Test Symp. (ETS)*, May 2022, pp. 1–10.

[4] T. Tran, S. R. Gundala, K. Soni, A. Baker, A. Fogle, and S. Chandrashekhar, "No trouble found (NTF) customer return analysis," in *Proc. IEEE Int. Rel. Phys. Symp. (IRPS)*, Apr. 2020, pp. 1–6.

[5] P. Bernardi, R. Cantoro, S. De Luca, E. Sánchez, and A. Sansonetti, "Development flow for on-line core self-test of automotive microcontrollers," *IEEE Trans. Comput.*, vol. 65, no. 3, pp. 744–754, Mar. 2016.

[6] E. Zanoni and P. Pavan, "Improving the reliability and safety of automotive electronics," *IEEE Micro*, vol. 13, no. 1, pp. 30–48, Feb. 1993.

[7] S. Alcaide, L. Kosmidis, H. Tabani, C. Hernandez, J. Abella, and F. J. Cazorla, "Safety-related challenges and opportunities for GPUs in the automotive domain," *IEEE Micro*, vol. 38, no. 6, pp. 46–55, Nov. 2018.

[8] P. Bernardi, G. Filipponi, G. Iaria, C. Bertani, and V. Tancorre, "Logic diagnosis based on logic built-in self-test signatures collected in-field from failing system-on-chips," *Electronics*, vol. 13, no. 21, p. 4234, Oct. 2024.

[9] Y. Huang, R. Guo, W.-T. Cheng, and J. C.-M. Li, "Survey of scan chain diagnosis," *IEEE Design Test Comput.*, vol. 25, no. 3, pp. 240–248, May 2008.

[10] B. Benware, C. Schuermyer, M. Sharma, and T. Herrmann, "Determining a failure root cause distribution from a population of layout-aware scan diagnosis results," *IEEE Design Test Comput.*, vol. 29, no. 1, pp. 8–18, Feb. 2012.

[11] A. Cook, M. Elm, H.-J. Wunderlich, and U. Abelein, "Structural in-field diagnosis for random logic circuits," in *Proc. 16th IEEE Eur. Test Symp.*, May 2011, pp. 111–116.

[12] P. Bernardi, L. Cardone, G. Iaria, D. Appello, G. Garozzo, and V. Tancorre, "About the correlation between logical identified faulty gates and their layout characteristics," in *Proc. IEEE 29th Int. Symp. On-Line Test. Robust Syst. Design (IOLTS)*, Jul. 2023, pp. 1–7.

[13] G. Iaria, P. Bernardi, C. Bertani, L. Cardone, G. Garozzo, and V. Tancorre, "A comprehensive scan test cost model to optimize the production of very large SoCs," *IEEE Trans. Comput.*, vol. 74, no. 4, pp. 1278–1292, Apr. 2025.

[14] A. Riefert, R. Cantoro, M. Sauer, M. S. Reorda, and B. Becker, "On the automatic generation of SBST test programs for in-field test," in *Proc. Design, Autom. Test Eur. Conf. Exhib. (DATE)*, Mar. 2015, pp. 1186–1191.

[15] I. Pomeranz, "Selecting functional test sequences for defect diagnosis," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 26, no. 10, pp. 2160–2164, Oct. 2018.

[16] S. Gopalsamy and I. Pomeranz, "A storage based LBIST scheme for logic diagnosis," in *Proc. IEEE 42nd VLSI Test Symp. (VTS)*, Apr. 2024, pp. 1–7.

[17] J. Rajski, V. Chickermane, J.-F. Coté, S. Eggersglüß, N. Mukherjee, and J. Tyszer, "The future of design for test and silicon lifecycle management," *IEEE Design Test*, vol. 41, no. 4, pp. 35–49, Aug. 2024.

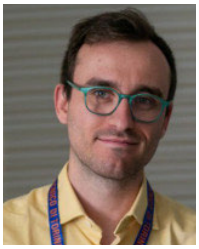
[18] M. Casarsa, G. Harutyunyan, and Y. Zorian, "Test and diagnosis solution for functional safety," in *Proc. IEEE Int. Test Conf. (ITC)*, Nov. 2020, pp. 1–5.

[19] P. Bernardi, G. Filipponi, G. Iaria, C. Bertani, and V. Tancorre, "Enhancing logic diagnosis of field returns through logic BIST in automotive SoCs," in *Proc. IEEE 26th Latin Amer. Test Symp. (LATS)*, Mar. 2025, pp. 1–6.

[20] M. Abramovici, M. A. Breuer, and A. D. Friedman. (1990). *Digital Systems Testing and Testable Design*. [Online]. Available: https://api.semanticscholar.org/CorpusID:31291424

[21] A. Gattiker, "Diagnosis meets physical failure analysis: How long can we succeed?" in *Proc. Int. Conf. Test*, 2004, p. 1441.
[22] J. D'Souza, "Improving PFA accuracy and defect localization with volume scan diagnosis," in *Proc. IEEE Int. Symp. Phys. Failure Anal. Integr. Circuits (IPFA)*, Jul. 2018, pp. 1–5.
[23] S. Wang, Y. Liu, and H. Bai, "Automotive electronics EOS failure analysis and diagnosis," in *Proc. IEEE Int. Symp. Phys. Failure Anal. Integr. Circuits (IPFA)*, Jul. 2024, pp. 1–5.
[24] Y. Huang, W. Yang, and W.-T. Cheng, "Advancements in diagnosis driven yield analysis (DDYA): A survey of state-of-the-art scan diagnosis and yield analysis technologies," in *Proc. 20th IEEE Eur. Test Symp. (ETS)*, May 2015, pp. 1–10.
[25] D. Pietsch, M. Matthes, U. Wieland, S. Ihlenfeldt, and T. Munkelt, "Root cause analysis in industrial manufacturing: A scoping review of current research, challenges and the promises of AI-driven approaches," *J. Manuf. Mater. Process.*, vol. 8, no. 6, p. 277, Dec. 2024.
[26] S. Holst, M. Kampmann, A. Sprenger, J. D. Reimer, S. Hellebrand, H.-J. Wunderlich, and X. Wen, "Logic fault diagnosis of hidden delay defects," in *Proc. IEEE Int. Test Conf. (ITC)*, Nov. 2020, pp. 1–10.
[27] I. Pomeranz, "Multicycle tests with fault detection test data for improved logic diagnosis," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 41, no. 5, pp. 1587–1591, May 2022.
[28] (2026). *Rhythm by Roguevation*. Accessed: 2026. [Online]. Available: https://roguevation.com/index.php/solutions/rhythm
[29] F. Angione, P. Bernardi, N. D. G. Giardino, G. Filipponi, C. Bertani, and V. Tancorre, "A system-level test methodology for communication peripherals in system-on-chips," *IEEE Trans. Comput.*, vol. 74, no. 2, pp. 731–739, Feb. 2025.
[30] G. Filipponi, "FSWGEN: A device-tree specification driven system-level test workload generator," in *Proc. IEEE Int. Test Conf. (ITC)*, Sep. 2025, pp. 558–559.
[31] F. Angione, P. Bernardi, G. Iaria, C. Bertani, and V. Tancorre, "Automatic generation of system-level test for un-core logic of large automotive SoC," *IEEE Trans. Comput.*, vol. 74, no. 9, pp. 3195–3209, Sep. 2025.

GIUSY IARIA (Member, IEEE) received the Ph.D. degree in computer and control engineering from Politecnico di Torino, Italy, in 2025. She is currently a Research Fellow with Politecnico di Torino. She is also a part of the Electronic CAD & Reliability Research Group with a strong focus on embedded electronics testing for large automotive devices.



GABRIELE FILIPPONI (Member, IEEE) received the Ph.D. degree in computer and control engineering from Politecnico di Torino, in 2026. He is currently a Research Fellow with Politecnico di Torino. He is also a part of the Electronic CAD & Reliability Group, with a focus on embedded electronics testing. His research interest includes ultra reliability for in-field automotive SoC, specifically exploiting LBIST.

TOMMASO FOSCALE (Graduate Student Member, IEEE) received the degree in computer engineering with a specialization in computer networks and cloud computing from Politecnico di Torino, and the master's thesis (cum laude), in 2021. He is currently pursuing the Ph.D. degree in computer and control engineering with Politecnico di Torino, affiliated with the Electronic CAD & Reliability Group. His research interest includes the development of testing techniques for automotive system-on-chip (SoC).

CLAUDIA BERTANI received the Graduate degree in electronic engineering from Politecnico di Milano. She is currently a Product and Test Engineer Manager at STMicroelectronics. She has more than 20 years of expertise in the semiconductor industry. She is also managing the team in charge of system-level test introduction on automotive ADAS SOC products.

GIUSEPPE GAROZZO was born in Catania, Italy, in December 1972. He received the M.S. and Ph.D. degrees in physics from the University of Catania, in 1996 and 2001, respectively. Since 1998, he has been with STMicroelectronics and works on data analysis and management. He is the author of some papers on plasma process simulation and statistical process control.

VINCENZO TANCORRE received the M.S. degree in electronics engineering from Politecnico di Bari. He is currently a Yield Enhancement Engineer at STMicroelectronics with more than 20 years of expertise in large SoC test for automotive applications. His research interests include test-related process monitoring using diagnostic solutions for memories and unstructured logic based on DfT methodologies.

PAOLO BERNARDI (Senior Member, IEEE) received the M.S. and Ph.D. degrees in computer science, in 2002 and 2006, respectively. He is currently an Associate Professor with Politecnico di Torino, where he is also with the Electronic CAD & Reliability Research Group. His current research interests include system-on-chip tests and reliability. He is the Program Chair of the IEEE International Test Conference 2025. He has been the General Chair of the IEEE European Test Symposium 2023.