IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 19 February 2024, accepted 5 March 2024, date of publication 11 March 2024, date of current version 15 March 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3375333

RESEARCH ARTICLE

Increasing the Solar Reliability Factor of a Dual-Axis Solar Tracker Using an Improved Online Built-In Self-Test Architecture

SORIN LIVIU JURJ AND RAUL ROTAR

Advanced Computing Systems and Architectures (ACSA) Laboratory, Computers and Information Technology Department, Politehnica University of Timisoara, 300223 Timisoara, Romania

Corresponding author: Sorin Liviu Jurj (jurjsorinliviu@yahoo.de)

ABSTRACT This paper introduces a novel mathematical approach to significantly enhance dual-axis solar trackers' Solar Reliability Factor (SRF) by developing and implementing an advanced Online Built-In Self-Test (OBIST) architecture. This innovative architecture is designed to efficiently address and correct single bit-flip errors within the system's microcontroller unit (MCU), a common control unit in contemporary solar trackers. By employing an improved diagnostic scheme based on extended Hamming codes, our OBIST architecture identifies and autonomously corrects all detected single bit-flip errors, reducing the fault coverage to 0%. This capability marks a significant advancement in the field, directly contributing to a substantial increase in the SRF by 47.48%. The study meticulously analyzes the potential fault domain influenced by environmental factors such as prolonged sunlight exposure and varying weather conditions, which are critical in the regular operation of solar trackers. Furthermore, we introduce a probabilistic model for defining and addressing stuck-at-faults, enhancing the system's overall reliability. The successful application of novel fault coverage-aware metrics demonstrates the OBIST architecture's effectiveness in improving solar tracker reliability, significantly contributing to the photovoltaic (PV) systems domain. This research presents a groundbreaking approach to enhancing solar tracker reliability and sets the stage for future advancements in the maintenance and efficiency of renewable energy systems.

INDEX TERMS Online built-in self-test, bit-flip mechanism, extended hamming codes, stuck-at faults, solar tracking.

I. INTRODUCTION

The pursuit of enhanced reliability in photovoltaic (PV) systems, particularly in solar tracking devices, is a critical area of research driven by the growing demand for sustainable energy solutions. Despite significant advancements, static and mobile PV panels' reliability analysis remains challenging, often necessitating costly and energy-intensive diagnostic equipment. This leads to a concomitant increase in computational and hardware expenses, with some testing modules amplifying costs due to additional fault diagnosis algorithms required on the testing platform. Existing methodologies for diagnosing faults in PV systems, such as those employing Artificial Neural Networks (ANNs) and other advanced techniques, are effective yet often complex and resource-intensive. Moreover, the dynamic and harsh environmental conditions exposed to solar tracking systems necessitate robust fault-tolerant strategies to ensure continuous operation. The complexity of digital circuits within these systems, including microcontrollers and integrated circuits, introduces a susceptibility to hardware errors that can compromise stability and lead to system failure.

Recent research works [1], [2], [3], [4], [5] show that periodical diagnosis of the static solar panel's additional

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

equipment comprising (Direct Current) DC-DC converters, solar charge controllers, and power electronics is a mandatory requirement for assessing the reliability of the entire PV system. Regarding DC-DC inverter diagnosis, an appropriate method is based on ANNs, which estimate the current and voltage applied to an external resistive load and achieve high detection rates for open circuit faults [1]. In contrast, short-circuit current detection is achievable with a new fast detection module that, combined with a dynamic reactive power compensator, STATCOM, provides multiple grid functions, including the active monitoring of the current rise rate and current magnitude in the PV grid systems [2]. A more advanced fault diagnosis procedure can be performed via PV panel power electronics (estimator, fault detection and identification logic, PV converter control system), implemented entirely on a single all-programmable System-on-Chip (SoC) device, which includes an FPGA and ARM core [3]. Other fault models, such as transmission lines and high impedance faults, are detectable using Mathematical Morphology (MM) based filters in DC systems [4] and utilizing Stockwell's Transform-based multi-resolution. Regardless of the considered fault models, PV systems must operate continuously, with their availability controlled by a Fault Tolerant Control (FTC) strategy.

On the other hand, mobile PV systems consisting of solar tracking devices pose a more challenging issue in the Design for Testing (DFT) domain and fault diagnosis methodology due to the higher perplexity of the electrical equipment. Since climate change is a very relevant problem in today's society [6] and considering goal number 7 (affordable and clean energy) and goal number 13 (climate action) of the United Nation's sustainable development goals [7], efforts to develop and use low-power embedded devices are made by many companies, examples varying from Application Programming Interface (API) oriented Arduino MCUs to more energy-efficient STM32 Nucleo boards which can be easily integrated into an embedded platform. Consequently, to reduce the carbon footprint and the electricity bills, efforts towards renewable energy are made [8], with many researchers building solar tracking systems [9], [10], [11], [12], [13] to capture the Sun's energy with maximum efficiency, as seen in Table 1.

Regarding the data provided in Table 1, most solar trackers' electrical equipment comprises sensors (light-dependent resistors), DC-geared motors (servomotors), motor drivers, and a main Arduino UNO board. Regarding the hardware setup used in this work, the one seen in [14] is chosen, and it distinguishes itself from the above-listed implementations by replacing the traditional light-dependent resistors with 4 Optocoupler components encapsulated in an LTV-847 Integrated Circuit (IC). The proposed prototype also uses an Arduino UNO, two L298N motor drivers, and a pair of Stepper motors to steer the solar panel according to the cast-shadow principle [14].

Because solar tracking systems are deployed outside of domestic homes for in-field testing and considering the complexity of the digital circuits (MCUs, ICs, and Dual H-Bridges) incorporated in the electrical equipment of solar trackers, several hardware errors can affect the device's stability, leading to system failure in more severe cases. These hardware errors can occur because of prolonged sunlight exposure and changing weather conditions, all of which affect the regular operation of the external circuitry.

Temperature, for example, affects the value of resistors and the properties of transistors, capacitors, inductors, batteries, and other components. Electronic devices are designed to work within specific temperatures, generally 0 to 70 or -40 to 60 degrees Celsius. Parametric faults [15] often refer to the parameters of the components that drift from their tolerance range but do not affect their connectivity. Although the parametric faults cannot result in a complete break of the circuit's function, they will produce unwanted output and cause unacceptable circuit performance for solar tracking systems. However, since the initial solar tracking prototype was improved [14], several IC boards (L298N motor drivers) have been equipped with larger heatsinks, allowing proper heat dissipation and thus reducing the risks of parametric faults. On the other hand, constant rainfalls and high humidity conditions can damage components in three ways: a) altering IC packages, b) leading to corrosion, or c) causing shorting

TABLE 1. Various solar tracking systems implementations based on the arduino uno microcontroller.

Paper Title	Solar Tracker Type	Components	Power Gain
"An Analysis on Arduino based Single Axis Solar Tracker" [9]	Single-Axis	Solar Panel, DC geared motor, Light-dependent resistor, Driver Integrated Circuit, Transistor, Operational amplifier, Potentiometer, Voltage regulator, Capacitor, Resistors, Battery, Arduino UNO	Not specified
"Arduino based low cost active dual-axis solar tracker" [10]	Dual-Axis	Solar Panel, Light-dependent resistor, Servo Motors, Driver Integrated Circuit, Arduino UNO	13.44%
"Design and Implementation of a Solar Tracker System with Dual Axis for Photovoltaic Panels in El Oued Region of Algeria" [11]	Dual-Axis	Solar Panel, Light-dependent resistor, 5V Relays, 12V Charge/Discharge Regulator, 12V-62Ah lithium-ion battery, 12V DC motors, Arduino UNO	~20%
"Dual-axis solar tracker based on predictive control algorithms" [12]	Dual-Axis	Solar Panel, Temperature sensor, G15 Cube Servo ID 1, G15 Cube Servo ID 2, G15 Driver, 5V and 12V Power Supplies, SD Module, PC, Arduino UNO	21.4%
"Performance Enhancement of Solar Panel Using Dual Axis Solar Tracker" [13]	Dual-Axis	Solar Panel, Light-dependent resistor, Servo Motors, Driver Integrated Circuit, SD Card	33.16%

or changing impedance. For security purposes, the external circuitry of solar trackers is covered by a protective case that lowers the chance of water leakage, thus eliminating the risks of short circuits on electric devices.

Since it is out of scope for this paper to analyze all the above-mentioned fault scenarios, this case study will focus on improving the detection of stuck-at faults by considering the reliability of the tested devices. Due to imperfections caused by the manufacturing process, specific revisions of Arduino UNO devices become prone to hardware faults, producing undesired outputs despite the intact firmware running on the portable board. This is explainable since the I/O pins are very vulnerable to electrostatic discharge (ESD), causing the outputs to fail during long-term tasks, such as orienting the payload of the solar tracking systems daily.

Given the importance of the two domains of DFT and Renewable Energy in today's society's needs, this work introduces an efficient and low-cost implementation of an improved OBIST architecture that can be successfully deployed to test the digital equipment of solar tracking systems. This architecture addresses the potential fault domain of solar tracking components, considering risks such as prolonged sunlight exposure and changing weather conditions. As a case study, the enhanced OBIST design is connected to the electrical components of the dual-axis solar tracker, applying the bit-flip mechanism to the Arduino UNO. For this, a novel probabilistic model is proposed to define stuck-at-faults that can affect the functionality of the Atmega328 MCU, the primary control unit in prevalent solar trackers. This model enhances the detection of such faults, which are critical to the reliability of the tested devices. The error detection method follows the stuck-at-fault model by comparing the Circuits Under Test (CUTs) healthy patterns with the invalid signatures stored in a dedicated database.

We successfully identify and repair all single bit-flip errors by applying an improved OBIST diagnosis scheme based on extended Hamming codes, reducing the fault coverage to 0%. This significant reduction in fault coverage is instrumental in increasing the SRF by 47.48%. The goal of the repairing mechanism is to decrease the fault coverage of singular bit-flip errors, which can alter the performance of the solar tracking system. The last stage of this work presents how the estimated fault coverage is reused to compute a Solar Test Factor (STF), which quantifies the total number of detected system errors. The STF parameter is further employed to calculate an SRF parameter that ultimately evaluates the reliability/availability of the entire solar tracking device.

By addressing these research gaps, our contributions significantly advance solar energy reliability. The proposed OBIST architecture and associated methodologies not only enhance the fault tolerance of solar tracking systems but also present a cost-effective and energy-efficient solution for maintaining and improving PV system reliability.

The paper is organized as follows. In Section II, the related work is presented. Section III details the probabilistic model for detecting Stuck-at faults. Section IV presents the improved OBIST architecture with a bit-flip correction strategy. Section V describes the experimental setup and results. Section VI presents the limitations of this study and future work. Finally, Section VII summarizes the conclusions of this paper.

II. STATE-OF-THE-ART BUILT-IN SELF-TEST

Built-In Self-Test (BIST) architectures are low-cost, energy-efficient testing techniques that accommodate various DFT circuits and components. Traditional BIST designs, on the other hand, have two critical drawbacks due to their static deployed structure: limited fault coverage and a lack of diverse operational modes. Current research indicates a rising interest in designing reconfigurable designs and versatile BIST architectures that may be tailored to various DFT requirements to circumvent these constraints.

High-speed link testing is one of the areas where a single form of pattern generation demonstrates its limitations. As a result, the authors of [16] suggest a multi-purpose BIST architecture that may be employed as a pattern generator, a pseudo-random sequence generator, a scrambler, a descrambler, or a snapshot and applied to various link tests. Because each of the previously mentioned components necessitates design work, the researchers created a dedicated circuit in 28-nm Complementary Met-al-Oxide-Semiconductor (CMOS) technology that operates at 10 GB/s while consuming only 9 mW, allowing them to validate all of the functions described in their paper.

On the other hand, Random Access Memory (RAM) testing is a testing domain that concentrates on multiple binary location checking and fixing. The authors of [17] propose a Reconfigurable Built-In Self-Repair (ReBISR) architecture for RAMs that includes diagnostic and repair capabilities. Compared to Dedicated Built-In Self-Repair (DeBISR) methodologies, the proposed reconfigurable architecture provides minimal hardware overhead and low area cost while improving fault coverage with the March C-algorithms, the foundations for RAM testing with minimum test length. Furthermore, in [18], the authors implement a ReBISR system for numerous repairable RAM cores of varying sizes and redundancy groups for SoC devices. Moreover, the authors offer a fast Built-In Redundancy Analysis (BIRA) algorithm for allocating redundancies in the ReBISR scheme. The suggested algorithm is implemented on a Reconfigurable BIRA (ReBIRA) for the ReBISR scheme, resulting in a high repair rate, a cheap area cost of the reconfigurable design, and a shortened analysis time to test time (0.25 percent ratio for the used March-14N algorithm).

As previously stated, complex SoC devices, particularly those designed with embedded IP cores, offer high DFT features like design reuse, reconfigurability, and customizability [19]. As a result, the authors of [20] suggest a Reconfigurable Processing Unit (RPU) for digital circuit testing that includes acceleration characteristics and BIST approaches. Their proposed architecture takes just 88 Clock Cycles (CCs), or 1.76 nanoseconds (ns), with a 50 MHz clock, resulting

in a total speedup of 1 × 104 when compared to traditional simulators, which take a total time of 17 milliseconds (ms) to complete test routine execution. Similarly, in [21], the authors describe a self-configurable platform for BIST applications based on four testing techniques. Two testing methodologies, sequential Runtime Reconfiguration (RTR) and sequential Compile-time Reconfiguration (CTR), are empirically compared, demonstrating that CUTs may be reduced by 60% without affecting device functioning. Furthermore, stuck-at-fault injection is possible in many regions of the CUTs, exposing both visible and undetectable defects. The work in [22] presents a novel cognitive BIST architecture for contact-free measurements, displaying low additional consumption (less than 2%) and shortened testing time.

Linear Feedback Shift Register (LFSR) and Configurable LFSR (CLFSR) units and security characteristics are critical for the DFT domain regarding BIST scan chains. The authors in [23] construct a CLFSR to generate the 8-bit and 16-bit test patterns. The simulation duration for the 8-bit and 16-bit CLFSRs is 51000 ns and 13107000 ns, respectively, with a 200 ns clock cycle for producing 255 and 65535 random outputs. The CLFSR can construct a programmable gold codes generator, a pseudo-random sequence generator, a Cyclic Redundancy Check (CRC) generator and checker circuit, DFT, and BIST design. The work in [24] describes the design of efficient programmable test-per-scan logic BIST modules, which include a Test Pattern Generator (TPG), Output Response Analyzer (ORA), comparator, and Read-Only Memory (ROM) for testing. Configurability is added to every structural element in BIST to improve IC testing fault coverage. The suggested structural architecture is simulated in the Modelsim Register Transfer Level (RTL) simulator, and the fault coverage spans from 39.26 percent to 88.59 percent. The authors of [25] present a novel Multiple input Single Input Change (MSIC) TPG for SoC device module testing. The MSIC test vectors are generated using a reprogrammable Johnson counter and fixed seed values. The author's experimental results demonstrate that the MSIC design can save up to 7.5 percent of test power. The authors of [26] present a secure scan chain that employs a phase-locking mechanism and a reconfigurable LFSR to create a dynamic length key to test the CUT, significantly reducing area overhead. The proposed solution was evaluated utilizing TSM CMOS 65nm technology, demonstrating that the area overhead is lower than existing market solutions. Circular Self-Test Path (CSTP) is an appealing technique for testing digital ICs in the nanometer era. It allows for at-speed testing with a limited test data volume and a short test application period. However, due to the difficulty of checking random-pattern-resistant faults, CSTP cannot consistently achieve high fault coverage. Therefore, the authors in [27] describe a Deterministic CSTP (DCSTP) structure comprising a DCSTP chain and jumping logic that achieves high fault coverage with a low area overhead. Their experimental results on ISCAS'89 benchmarks show that 100% fault coverage can be obtained with low area overhead and CPU time, especially for large circuits.

The work in [37] explores a novel approach to fault detection in solar PV systems. This research uses Support Vector Machine (SVM) algorithms and thermal image processing techniques to identify and diagnose faults within solar panels, a critical component of broader solar energy systems. While the OBIST architecture in this paper focuses on dual-axis solar trackers and employs a hardware-centric approach with software support for fault correction, the SVM and thermal imaging method targets the PV panels directly, using data analysis and image processing to detect anomalies.

The work in [38] proposes to improve the performance of solar PV systems by detecting faults and reconfiguring the PV array using thermal image processing. The authors propose a method that uses a thermal camera to capture images of the PV array, which are then analyzed using MATLAB/Simulink to extract image properties. These properties are used to classify faults and to determine a reconfiguration pattern for the PV array to optimize power output, especially under conditions of partial shading or minor faults. More precisely, the paper details the development of a fault classifier and a reconfiguration algorithm based on thermal image analysis. The classifier and algorithm are validated using a 5 kW PV system with a 4 × 5 Total-Cross-Tied (TCT) array configuration. The proposed method is compared to existing processes, and the results show that it can effectively mitigate the effects of partial shading and other minor faults, leading to enhanced power output from the PV system. The authors also discuss implementing the image processing technique for fault detection and reconfiguration, including the setup for capturing thermal images and extracting image features for fault classification using a neural network. The paper concludes that the proposed method is feasible and can be implemented in real-time applications, potentially benefiting large PV power plants by providing a means for early fault detection and system reconfiguration to maximize power generation efficiency.

The study in [39] introduces an approach that compares residual faults with a predefined threshold value. The main goal of this approach is to accurately detect and locate faults within the lines of a solar rooftop string, a series of solar panels connected to form a more extensive PV system. The methodology focuses on identifying line-line faults, common types of faults in PV systems that can lead to significant power loss or system damage if not promptly addressed. The effectiveness of the proposed fault detection and localization technique is validated through simulations using MATLAB/Simulink and practical experiments using a 3 × 3 solar panel setup. The experiments are conducted first with PV arrays of a fixed size and then with arrays of varying sizes to test the adaptability and accuracy of the method across different system configurations. The findings from the study indicate that the proposed technique is both practical and accurate in detecting and locating faults within solar PV systems. This has important implications for the protection and maintenance of these systems, as it can help in the early identification of issues, thereby reducing downtime,

preventing further damage, and ensuring optimal performance and longevity of solar PV installations.

Also, the authors in [40] address the issue of Partial Shading Conditions (PSC) on solar PV systems, which can significantly affect their performance by creating multiple power maxima in their electrical characteristics. Partial shading is a common problem where some panels in a PV array are shaded. In contrast, others are fully exposed to sunlight, leading to a mismatch in electricity generation across the array. Bypass diodes can mitigate this mismatch, which protects the system by allowing current to pass around shaded modules. Still, it also results in multiple maxima in the power-voltage curve, including local and global Maximum Power Points (MPP). To ensure that the PV system operates at its most efficient point, tracking and using it at the global MPP, even under PSC, is necessary. The paper discusses two Metaheuristic Maximum Power Point Tracking (MPPT) strategies: the Enhanced Grey Wolf Optimization (EGWO) algorithm and the Marine Predator Algorithm (MPA). These algorithms are designed to help the PV system find and maintain operation at the global MPP despite the challenges posed by dynamically changing shading patterns. Metaheuristic algorithms are inspired by natural processes and are used to solve complex optimization problems. The EGWO is an improved version of the Grey Wolf Optimization algorithm, which mimics the leadership hierarchy and hunting mechanism of grey wolves in nature. The foraging behavior of marine predators inspires the MPA. Both algorithms are used to search for the global MPP efficiently and effectively. Finally, the paper compares these two algorithms. It discusses their application in tracking the global MPP under PSC for an 8S (eight series-connected panels) PV setup. It presents the tracking results to demonstrate the performance of the suggested MPPT approaches in dealing with the challenges of PSC.

The work in [41] discusses the importance of identifying faults in PV arrays to enhance the safety and reliability of PV systems. PV arrays, which convert sunlight into electricity, have non-linear characteristics and are equipped with technologies like MPPTs and blocking diodes to optimize their performance. However, these features can also lead to mismatch levels that complicate the system's operation. The study focuses on detecting two types of faults: line-to-line and line-to-ground. These faults are isolated using Overcurrent Protection Devices (OCPD) and Ground Fault Protection Devices (GFPD) to prevent damage to the system. The research explores AI-based techniques to improve fault detection accuracy, including Fuzzy inference, wavelet analysis, SVM, and K-Nearest Neighbors (KNN). Despite the potential of AI-based methods for fault detection, the paper identifies two main drawbacks: the need for large datasets to effectively identify faults, which can be challenging under conditions of low irradiation, and the requirement for a significant number of voltage and current sensors, which can increase the complexity and cost of the system. To address these issues, the researchers propose a method that uses a small number of sensors. They test this method on a 160 W, 4 $\times$ 4 solar PV array under various fault conditions. The proposed method successfully identifies faults that conventional methods fail to detect. Moreover, the study reports a significant power gain of approximately 152% (97 W) in the PV array due to the effective identification of faults.

Also, the authors in [42] provide a comprehensive review of various fault detection techniques for PV systems, emphasizing the importance of accurate monitoring and periodic follow-up to optimize performance and minimize losses due to faults. It discusses the classification of PV system faults, including temporary and permanent failures, and the challenges in determining the type and location of faults for efficient maintenance. The work highlights the role of thermography methods in classifying and localizing different types of faults. It presents an overview of recent techniques that combine AI tools with thermography methods for improved fault detection. Additionally, it covers other AI techniques used in fault detection of PV systems and provides guidance and recommendations for future research in this area.

State-of-the-art approaches in the DFT domain show that configurable LFSR designs integrated into reconfigurable BIST architectures efficiently increase the detection ratio, thus improving the global fault coverage of intrusive hardware errors. Therefore, this work distinguishes itself from the previous works by proposing an efficient and enhanced OBIST architecture that corrects singular bit-flip errors in real time with the help of a dedicated bit-flip mechanism.

III. PROBABILISTIC MODEL FOR STUCK-AT FAULTS DETECTION

This chapter focuses on adapting a probabilistic representation for modeling stuck-at faults by utilizing the dual-axis solar tracker's component's reliability as a middleman for the theoretical background. Therefore, this section is divided into two major parts: the solar tracking system's software implementation description and the targeted stuck-at faults reliability-based probabilistic framework.

A. STUDY CASE OF THE DUAL-AXIS SOLAR TRACKER

The software code is designed on the Arduino UNO platform. It circumvents the need for sophisticated mathematical procedures that rely on geometrical formulas to calculate the Sun's position's theoretical altitude and azimuth angles. Fig. 1 depicts a simplified model of the pseudocode that served as the foundation for the developed automation program [14].

To gain a better insight into the working principle, each solar cell group from the four corners will be designated as TR (Top Right), TL (Top Left), BR (Bottom Right), and BL (Bottom Left) in linkage with their locations. The average voltages for each side, such as AVR (Average Voltage Right), AVL (Average Voltage Left), AVT (Average Voltage Top), and AVD (Average Voltage Down), are also considered. In contrast, the total average value will be denoted as TAV. Before the algorithm starts its typical routine, the solar tracker generally has an initial eastward position. After the Sun rises,

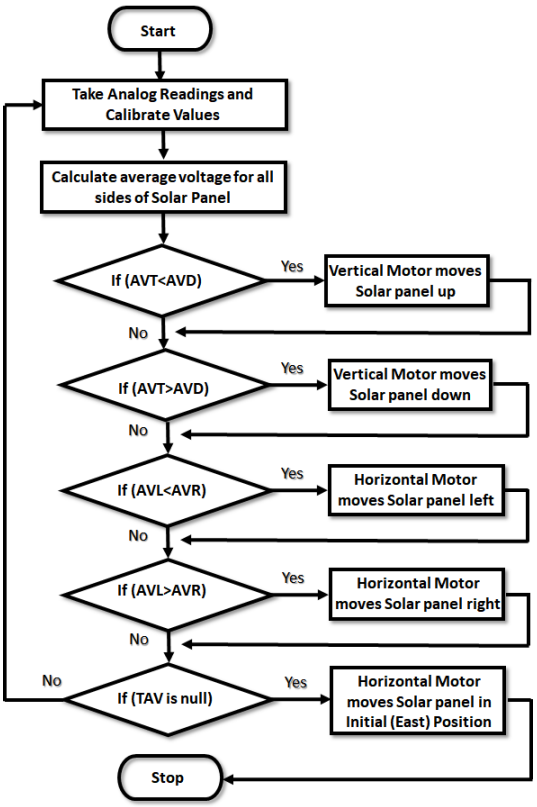

FIGURE 1. Logical flowchart of the dual-axis solar tracking system [14]. Based on the average voltage values, the system decides whether to rotate the solar panel and in which direction. The decision-making process involves checking if the total average voltage of the four corners is equal to or less than a predefined number (for example, 8) and if the limit switches have the required range. This ensures that the panel moves only when necessary and within mechanical constraints.

the Optocoupler's incoming voltage values are temporarily stored in newly declared variables that require additional in-program calibration. At this stage, average values for all sides of the solar panel will be determined using formulas from the Arduino IDE's *math.h* library.

Although it is not explicitly described in Fig. 1, before the algorithm checks each *if* branch condition to rotate the solar panel in the correct direction, it usually verifies if the total average voltage of the four corners is equal to or less than a predefined number (for example, 8) and if the limit switches have the required range. This procedure is performed for both stepper motors, ensuring that analog readings are constantly updated. A shadow on one of the pairs of cell groups commonly determines the direction in which the solar panel moves. When the algorithm identifies a significant difference in voltage averages, it rotates the panel toward the shaded area. When all voltages on the four corners reach zero at the end of the day, the solar tracker will drive the engine horizontally back to its start position, expecting the Sun to rise again the following day.

To summarize all possible PV panel positioning scenarios, a truth table based on the Optocoupler and Arduino UNO's combined logic is constructed by following the flowchart stages in Fig. 1. According to the data values in Table 2, the Optocoupler can be viewed as a pure combinational device. In contrast, the Arduino UNO will be treated as a sequential unit. Additional variables for describing the movements of the dual-axis solar tracker were added on the right section of Table 2, representing the right rotation (RR), the left rotation (LR), the upward rotation (UR), and the downward rotation (DR).

For simplicity, all rows were completed with a logic HIGH or LOW voltage (for the Optocoupler and Arduino UNO) or active/inactive (A/I) states of the PV panel. Despite the well-defined logic in Table 2, altered signal values that may be delivered or generated from the Arduino UNO disrupt the logic of the MCU's firmware and thus cause significant energy loss due to the incorrect placement of the PV panel. Therefore, a more in-depth signal analysis was conducted

TABLE 2. Dual-axis solar tracking device rotation decision based on the LTV-847 and Arduino UNO's logic.

Optocoupler combined with ADC				Arduino UNO Internal Logic				PV Panel Rotation Direction			
Logic Output Configurations				*Logic Output Configurations*				*Logic Output Configurations*			
TR	*TL*	*BR*	*BL*	*AVR*	*AVL*	*AVT*	*AVD*	*RR*	*LR*	*UR*	*DR*
LOW	LOW	LOW	LOW	LOW	LOW	LOW	LOW	I	I	I	I
LOW	LOW	LOW	HIGH	LOW	HIGH	LOW	HIGH	A	I	A	I
LOW	LOW	HIGH	LOW	HIGH	LOW	LOW	HIGH	I	A	A	I
LOW	LOW	HIGH	HIGH	HIGH	HIGH	LOW	HIGH	I	I	A	I
LOW	HIGH	LOW	LOW	LOW	HIGH	HIGH	LOW	A	I	I	A
LOW	HIGH	LOW	HIGH	LOW	HIGH	HIGH	HIGH	A	I	I	I
LOW	HIGH	HIGH	LOW	HIGH	HIGH	HIGH	HIGH	I	I	I	I
LOW	HIGH	HIGH	HIGH	HIGH	HIGH	HIGH	HIGH	I	I	I	I
HIGH	LOW	LOW	LOW	HIGH	LOW	HIGH	LOW	I	A	I	A
HIGH	LOW	LOW	HIGH	HIGH	HIGH	HIGH	HIGH	I	I	I	I
HIGH	LOW	HIGH	LOW	HIGH	LOW	HIGH	HIGH	I	A	I	I
HIGH	LOW	HIGH	HIGH	HIGH	HIGH	HIGH	HIGH	I	I	I	I
HIGH	HIGH	LOW	LOW	HIGH	HIGH	HIGH	LOW	I	I	I	A
HIGH	HIGH	LOW	HIGH	HIGH	HIGH	HIGH	HIGH	I	I	I	I
HIGH	HIGH	HIGH	LOW	HIGH	HIGH	HIGH	HIGH	I	I	I	I
HIGH	HIGH	HIGH	HIGH	HIGH	HIGH	HIGH	HIGH	I	I	I	I

in [28] by developing an OBIST for single bit-flip and single stuck-at-fault detection. Output signals of the Arduino UNO MCU were continuously monitored in real-time with the help of a PC Oscilloscope, according to the waveforms presented in Fig. 2.

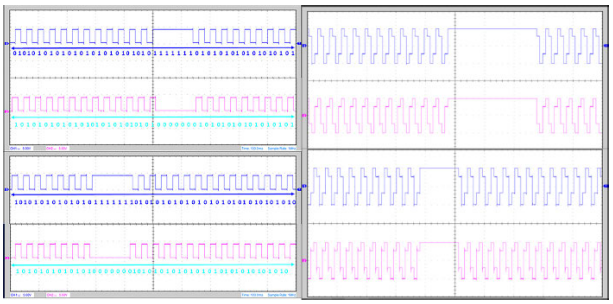

FIGURE 2. Arduino UNO valid signals (left) and Arduino UNO distorted signals (right) by external noise, collected from 4 digital pins [28].

The healthy waveforms of the Arduino MCU, represented by step signal values generated according to the repeating sequences of 32-bit length each, are depicted in the left frame of Fig. 2, whereas corrupted data signals by external noise, which are not recognized by the solar tracking test program can be seen in the right frame of Fig. 2. Based on these findings, the detection mechanism presented in [28] can be improved by proposing an adapted probabilistic reliability-oriented model for multiple stuck-at faults, as further detailed in Section B of this chapter.

B. A PROBABILISTIC MODEL FOR MULTIPLE STUCK-AT FAULTS IN COMBINATIONAL AND SEQUENTIAL DEVICES

The Many variables associated with technological scaling, including manufacturing accuracy constraints, parametric device fluctuations, supply voltage reduction, increased operation frequency, and power dissipation, have influenced the demand for dependability in nanoscale designs. These factors increase the likelihood of a defect in a circuit, primarily permanent faults caused by process stages. The Stuck-at fault is one of these faults [29], a type of permanent defect that can occur in transistors. A transistor with a Stuck-at fault defect will always drive current, regardless of the signal given to its input terminal. Many strategies have been proposed to improve the dependability of ICs. Most of these solutions rely on various types of redundancy, such as time, hardware, and information redundancy. These strategies, in general, add some overhead to the system. Hardware redundancy, for example, has an area penalty. TMR (Triple Module Redundancy) is a widely used hardware redundancy technique. TMR, on the other hand, significantly expands the circuit area [30].

One of the most complex issues is establishing the tradeoff between the benefits and drawbacks of insert or no tolerance approaches. Due to the high complexity of reliability evaluation, these strategies may diminish scaling gains in some circumstances. In these cases, probabilistic and stochastic approaches are frequently preferable. A matrix structure for transistors can be used to model components at the circuit level. The Probabilistic Transfer Matrix (PTM) formulation, which describes the output of a circuit for the input vectors, is one of the most accurate methods. The matrix M in PTM comprises i lines and j columns, with the (i, j)th entry representing the probability p of an occurrence of output value j given input value, i.e., i, $p(j \mid i)$. An Ideal Transfer Matrix (ITM) in a fault-free circuit displays the correct value at the output with probability 1, i.e., no-fault probability [30].

Although most PTM implementations consider a single fault probability r for all input vectors in the matrix, the fault probability varies depending on the type of fault. The unique r simplifies the reliability computation using PTM but obscures the input influence on many types of errors. This section demonstrates the effect of inputs on the fault probability for Stuck-on permanent faults. It also proposes a methodology for modeling the probability of Stuck-on faults in combinational circuits while considering the input effects on reliability. The probabilistic model used in most dependability approaches, such as the classic PTM technique, is fundamentally oriented towards studying the device's transistor configuration to evaluate the reliability under Stuck-on faults. The main advantage of this mathematical model is that it can be applied to combinational devices and sequential devices that can be separated from the storage elements [30].

For a more concrete example, let us consider the minimized combinational circuit of the LTV-847 IC in Fig. 3, according to its functionality presented in [28].

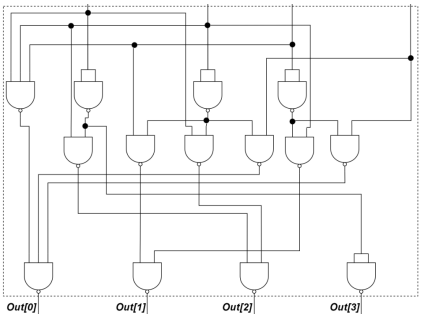

FIGURE 3. LTV-847 NAND gate implementation.

A truth table, a deterministic mapping of input values to output values, can represent the function of a combinational logic circuit in an error-free operating system. The ITM represents this function, where row indices in ITM represent all potential input combinations, while column indices represent all potential output values. For example, the truth table for a NAND gate converts the input vector !AB represents the input value 01 to the output value 1. However, during soft errors, persistent faults, or manufacturing problems, this input may occasionally result in a 0 at the gate's output. One can use a PTM to predict this behavior if one knows how frequently it will occur. In PTM and ITM, row indices reflect all potential input combinations, while column indices represent possible output values. Consider the following for NAND gate PTMs,

as illustrated in equation (1) [30]:

$$ITM_{NAND} = \begin{bmatrix} 0 & 1 \\ 0 & 1 \\ 0 & 1 \\ 1 & 0 \end{bmatrix} PTM_{NAND} = \begin{bmatrix} 1-r & r \\ 1-r & r \\ 1-r & r \\ r & 1-r \end{bmatrix} \quad (1)$$

where the gates provide the proper output value with probability r, where r is the gate's reliability factor. Generally, any given probability distribution can be used for the row of the matrices. Computing the entire circuit's reliability is possible using knowledge about a gate's dependability. The global PTM of a circuit is calculated by aggregating the PTMs of all the gates in the circuit. Two rules must be followed to compute a circuit's global PTM: a) if two gates G1 and G2 with PTM's PG1 and PG2 are combined in series, the resulting PTM is PG1 · PG2; b) when two gates G1 and G2 with PTMs PG1 and PG2 are coupled in parallel, the resulting PTM is PG1 PG2 or the tensor product of the matrices.

Based on the above-mentioned mathematical statements, the PTM of the first level of NAND gates, which are parallel connected, will be computed, resulting in a tensor product of all gates. Only one tensor product of two matrices will be calculated for demonstration purposes since the remaining calculations can be performed by following the previously mentioned rules. Additionally, to simplify the equations, the probability of a fault is denoted by p, as expressed in relation (2) [30]:

$$\begin{cases} PTM_{NAND} = \begin{bmatrix} 1-r & r \\ 1-r & r \\ 1-r & r \\ r & 1-r \end{bmatrix} \\ 1-r = p \end{cases} \Rightarrow PTM_{NAND} = \begin{bmatrix} p & r \\ p & r \\ p & r \\ r & p \end{bmatrix} \quad (2)$$

Thus, the tensor product of two PTM_{NAND} matrices is represented in equation (3) [30] as follows:

$$PTM_{NAND1} \otimes PTM_{NAND2} = \begin{bmatrix} p & r \\ p & r \\ p & r \\ r & p \end{bmatrix} \otimes \begin{bmatrix} p & r \\ p & r \\ p & r \\ r & p \end{bmatrix}$$

$$= \begin{bmatrix} p \cdot \begin{bmatrix} p & r \\ p & r \\ p & r \\ r & p \end{bmatrix} & r \cdot \begin{bmatrix} p & r \\ p & r \\ p & r \\ r & p \end{bmatrix} \\ p \cdot \begin{bmatrix} p & r \\ p & r \\ p & r \\ r & p \end{bmatrix} & r \cdot \begin{bmatrix} p & r \\ p & r \\ p & r \\ r & p \end{bmatrix} \\ p \cdot \begin{bmatrix} p & r \\ p & r \\ p & r \\ r & p \end{bmatrix} & r \cdot \begin{bmatrix} p & r \\ p & r \\ p & r \\ r & p \end{bmatrix} \\ r \cdot \begin{bmatrix} p & r \\ p & r \\ p & r \\ r & p \end{bmatrix} & p \cdot \begin{bmatrix} p & r \\ p & r \\ p & r \\ r & p \end{bmatrix} \end{bmatrix} = \begin{bmatrix} p^2 & p \cdot r & r \cdot p & r^2 \\ p^2 & p \cdot r & r \cdot p & r^2 \\ p^2 & p \cdot r & r \cdot p & r^2 \\ p \cdot r & p^2 & r^2 & r \cdot p \\ p^2 & p \cdot r & r \cdot p & r^2 \\ p^2 & p \cdot r & r \cdot p & r^2 \\ p^2 & p \cdot r & r \cdot p & r^2 \\ p \cdot r & p^2 & r^2 & r \cdot p \\ p^2 & p \cdot r & r \cdot p & r^2 \\ p^2 & p \cdot r & r \cdot p & r^2 \\ p^2 & p \cdot r & r \cdot p & r^2 \\ p \cdot r & p^2 & r^2 & r \cdot p \\ r \cdot p & p \cdot r & p^2 & p \cdot r \\ r \cdot p & p \cdot r & p^2 & p \cdot r \\ r \cdot p & p \cdot r & p^2 & p \cdot r \\ r^2 & p^2 & p \cdot r & p^2 \end{bmatrix} \quad (3)$$

The final PTM of the entire circuit can be computed by following rules (a) and (b). The adapted methodology investigates the strengths of PTM for gates under stuck-at faults, providing helpful information about gate reliability in each case of the input vectors. According to a detailed study conducted in [30], the reliability of a NAND gate was already established in two test scenarios: a) the value of the reliability is 0, 92475 considering stuck-at faults; b) the value of the reliability is 0, 90 without stuck-at faults [30]. When dealing with gate dependability, it is critical to understand the input vectors that reflect the output with maximum reliability. Establishing a PTM with more precise results for each combination enables the designer to determine which gates represent the better options when engaging with the design.

IV. IMPROVED OBIST ARCHITECTURE WITH BIT-FLIP CORRECTION STRATEGY

This section presents an improved variant of a previously proposed OBIST architecture [28] constructed using the extended Hamming code algorithms.

In computer theory, Hamming codes are a group of linear Error Correcting Codes (EECs) capable of detecting two-bit errors or correcting one-bit errors. On the other hand, the basic parity check code cannot correct errors and can only identify an odd number of bits in error. Thus, Hamming codes are considered perfect codes due to their property of achieving the highest possible codes with their block length and minimum distance of three. From the mathematical point of view, for each integer r ≥ 2, there is a code with block length denoted with n, and is expressed in the formula (4):

$$n = 2^r - 1 \quad (4)$$

The associated message length, which is denoted with k, will be written as in equation (5):

$$k = 2^r - r - 1 \quad (5)$$

Furthermore, by dividing equation (5) by equation (4), the rate of Hamming codes is obtained, as presented in the expression (6):

$$R = \frac{k}{n} = \frac{2^r - r - 1}{2^r - 1} = \frac{2^r - 1 - r}{2^r - 1} = \frac{2^r - 1}{2^r - 1} - \frac{r}{2^r - 1} = 1 - \frac{r}{2^r - 1} \quad (6)$$

which is the highest possible rate for codes with a minimum distance of three. A Hamming distance of four is accomplished by extended Hamming codes, which helps the decoder differentiate between when there is at most one one-bit error and several two-bit errors. Consequently, extended Hamming codes are known in the literature as Single-Error Correction and Double-Error Detection (SECDED). Moreover, recent approaches in the extended Hamming codes domain show that more than two-bit errors can be successfully detected with evolved error-detecting methods such as Single Error Detection, Double Error

Detection, Triple Adjacent Error Detection (SEC-DED-TAED) [31] and Soft Information Single Error Correction (SISEC) [32].

Inspired by the abovementioned works, this improved OBIST error detection and correction diagram adds several task blocks to the initial design. It implements an efficient bit-flip correction mechanism that reduces the fault coverage of single bit-flip errors to 0%. As can be seen in Fig. 4, the Pseudo-Random Pattern Generator (PRPG) and Dual Multiple-Input Signature Register (MISR) blocks are both connected to a BIST Controller that implements the functionality of the Switch Control Panel presented in [28], thus deciding when the test program will start injecting test vectors into the system.

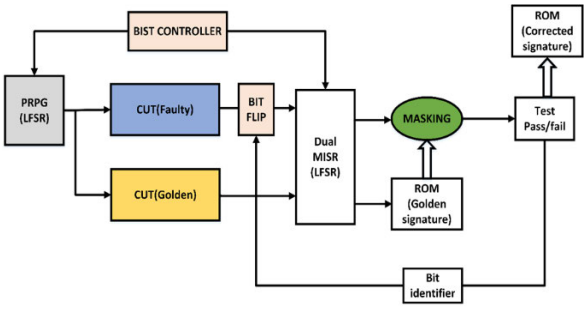

FIGURE 4. Block diagram of the improved OBIST architecture with Bit-Flip Mechanism. In the initial error detection phase, data from the PRPG (LFSR) is fed into both the faulty and reference circuits, with outcomes sent to the Dual MISR for analysis. This unit produces two signatures: one for the faulty circuit via the MASKING block and one for the reference circuit stored in ROM. The MASKING block identifies errors by XORing the inputs and indicates where corrections are needed, issuing a report based on the results. The second phase begins when the MASKING block detects a change in bit pattern, indicating that the OBIST architecture's correction function is needed. The Bit Identifier pinpoints the error for the "BIT FLIP" block to fix, and the corrected signature is then saved in ROM.

In the first stage, called the error detection stage, the data from the PRPG (LFSR) unit must be injected into the faulty and golden CUTs, and their responses will be transported to the Dual MISR unit. The Dual MISR block is provided with two outputs: one that generates the signature of the defective circuit in the MASKING block and the second that generates the gold signature in the dedicated ROM block. The MASKING block detects errors and performs the bitwise EXOR operation of both inputs that connect to this block. The output of this block will detect the positions on which errors have occurred in the string and which need to be corrected and consequently generate a success or failure report depending on the scenario.

The second stage of the block diagram presented in Fig.4 is triggered when a change in the bit pattern is detected and refers to the correction capability of the improved OBIST architecture. If the MASKING block reports a modified bit pattern, the Bit Identifier block will locate the error in real time and further direct it to the "BIT FLIP" block for correction. Finally, the corrected signature will be stored in a dedicated ROM block. However, data correction is possible after obtaining the Hamming results, which show us all the locations where the bits have been altered. This will help us implement the correction measures only for the respective locations, resulting in an efficient system in terms of execution time and energy consumption. The proposed solution is mathematically expressed as in the equation set (7):

$$\begin{cases} corr = MISR \oplus MISR\_f \\ M = Hamm(4, dist) \\ correction\_bit = shift(corr) \\ correction = bit\_flip(correction\_bit) \end{cases} \tag{7}$$

where parameter *corr* will retain the output bit resulting from the EXOR operation between the valid MISR and faulty MSIR, variable M stores the extended Hamming code of distance 4, *correction_bit* will retain the shifted bit from the EXOR operation, and correction will perform the bit-flip mechanism on the bit that requires repairing.

V. EXPERIMENTAL SETUP AND RESULTS

This section presents the Test Mode architecture of the proposed OBIST strategy and a reliability evaluation of the tested solar tracking equipment with and without error correction techniques.

A. TEST MODE ARCHITECTURE FOR THE PROPOSED OBIST STRATEGY

To validate the robustness of the proposed OBIST implementation, the initial CUT chain was cloned in the software simulation to obtain two devices, one that generates correct patterns and another one that provides incorrect responses. The proposed architecture aims to compare the pseudo-random MISR output signatures with the valid generated MISR signatures inside a dedicated block called Signal/Signature Analyzer, which is depicted in Fig. 5.

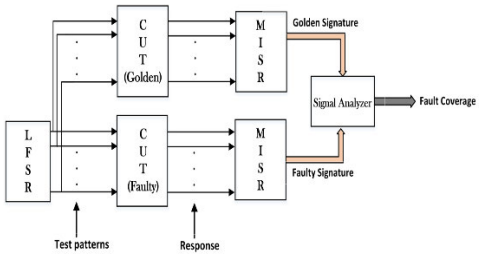

FIGURE 5. Test mode architecture for the proposed OBIST strategy. To test the proposed OBIST system's robustness, the initial CUT chain was replicated in software to create two devices: one generating correct patterns (CUT Golden) and the other incorrect ones (CUT Faulty). The architecture's purpose is to compare pseudo-random MISR output signatures with valid MISR signatures using a dedicated Signal/Signature Analyzer block.

During the software implementation, the fault coverage for three types of faults was evaluated: random singular and double bit-flips and single stuck-at faults. The test chain of the CUTs contains 16 bits, but targeting only 12 bits was of interest in the experiments. These 12 bits are associated with 3 CUTs: Optocoupler, Arduino UNO, and one L298N circuit. In the case of 12 bits, the possible number of faults

is 4.096. Thus, the entire chain of 16 bits can be divided into two parts: one is the least significant part, and the other is the most significant part.

The fault coverage can be evaluated by analyzing these two parts. Single bit-flip errors can be easily detected using the single parity checking method. The single parity checking method is applied in the software implementation by firstly performing an EXOR operation between all the 16 bits of the test chain and adding the extra parity bit in the least or most significant position of the bits chain, as seen earlier in Fig. 2. Secondly, to determine if a single bit-flip has occurred during data transmission, the Code worded signal is rechecked by calculating the parity bit in the same manner as mentioned earlier and comparing it to the initial extra parity bit. One parity bit will be added over the entire encoded word to detect double-bit errors reliably.

The fault coverage for the single bit-flips, double bit-flips stuck-at faults, and the global coverage are presented in Table 3, where FC_{SBF} represents the fault coverage of detected single bit-flip errors, FC_{DBF} stands for the fault coverage of detected double bit-flip errors, FC_{SaF} means the fault coverage for detected stuck-at faults, and FC_G represents the global fault coverage for both the single bit-flips and single stuck-at faults.

TABLE 3. Fault coverage of single bit-flip errors, double bit-flip errors as well as single bit stuck-at faults.

Crt. Nr.	Initial Seed (HEX)	FC_{SBF}	FC_{DBF}	FC_{SaF}	FC_G
		Last 8 bits (Mutant)			*Random 12 bits*
1	FFFF	93.95%	50.63%	100%	72.15%
2	8FFF	93.93%	50.24%		
3	8CFF	93.92%	50.23%		
4	8C9F	93.91%	50.44%		
5	8C94	93.94%	50.25%		

The software simulation applied the single parity checking method for single bit-flip errors. As a result, 93.93% of the targeted errors were detected from 65.535 injected test patterns, each with a different initial seed value during 5 test cases, as seen in Table 3. Concerning double bit-flip errors, double parity checking was applied. As a result, a 50.36% detection ratio was achieved from the same number of injected test patterns, showing that the occurrence of double-bit errors is 46.38% lower than that of singular-bit-flip errors.

It is known that stuck-at faults can be easily detected, as they mainly occur due to damaged logical gates, transistors, or permanent circuit damage. The five test cases regarding stuck-at faults seen in Table 3 are performed by injecting 8, 12, and 16 bits in the test chain, and based on the analysis of the Signal/Signature Analyzer, it was determined that the fault coverage is 100%. To obtain the global fault coverage of the test chain, the initial test cases were extended from 8 bits (the least significant and most significant part of the test chain) to 12 bits, representing the entire DUT. Thus, the global fault coverage was evaluated for the total number of 12 bits and was determined to be 72.15%, proving that the proposed OBIST solution can detect all targeted errors regardless of the initial seed value.

Because the aliasing usually happens when the flawed device's signature is the same as the perfect device's signature, the probability of aliasing occurrence is calculated with the relation $2^{-16} = 0.0001$ and results in the conclusion that during experiments, aliasing appears in sporadic cases.

B. INCREASING THE RELIABILITY FACTOR OF SOLAR TRACKING DEVICES USING A MODIFIED METRICS SYSTEM

Reliability evaluation is an essential metric for measuring the performance of modern solar tracking systems and can be assessed in several ways, depending on the targeted product component. Since the solar tracking equipment is composed of electrical components affected by hardware faults (single bit-flip errors, double bit-flip errors, and stuck-at faults), the focus is mainly on evaluating the reliability of the Optocoupler LTV 847, Arduino UNO board, and the two L298N ICs. This paper investigates the possibility of increasing the SRF parameter of solar tracking equipment using a modified equation system based on three novel fault coverage-aware metrics presented in previous works [33], [34], [35].

PV system availability and reliability can be estimated by inspecting collected failure data obtained from field testing [36]. Hence, this data is further used to establish the maintenance costs for future equipment repair. Similarly, the experimental data derived from the OBIST solution is used to shape the hardware fault coverage. Fault coverage is the percentage of tested vectors divided by the total number of test patterns. The calculation formula for the Fault Coverage (FC) is given in equation (8):

$$FC = \frac{T_V}{T_P} \cdot 100\% \tag{8}$$

where T_V represents the number of test vectors, and T_P designates the number of tested patterns. Further, a hardware error factor is defined denoted with E and expressed as in the mathematical relation (9):

$$E = FC \cdot N_P \tag{9}$$

where FC is the hardware fault coverage, and N_P represents the total number of test patterns. The error factor is an important parameter that can be used to simplify the Solar Test Factor (STF) involved in the fault coverage-aware metrics equation set calculus. The STF parameter for hardware test scenarios can be expressed mathematically as presented in relation (10):

$$STF_H = \frac{E}{T_P \cdot 2^D} \tag{10}$$

where T_P designates the number of tested patterns, and D represents the number of D flip-flops employed in the MISR\s hardware design. However, expression (8) is a general formula and must be correspondently adapted to single bit-flips,

double bit-flips, and stuck-at faults. Therefore, the initial metrics system [33] is modified to fit the three test scenario requirements, as presented in equation system (11):

$$\begin{cases} STF_{SBF} = \dfrac{E_{SBF}}{T_P \cdot 2^D} \\ STF_{DBF} = \dfrac{E_{DBF}}{T_P \cdot 2^D} \\ STF_{SaF} = \dfrac{E_{SaF}}{T_P \cdot 2^D} \\ STF_G = \dfrac{STF_{SBF} + STF_{DBF} + STF_{SaF}}{3} \\ STF_G = \dfrac{E_{SBF}}{T_P \cdot 2^D} + \dfrac{E_{DBF}}{T_P \cdot 2^D} + \dfrac{E_{SaF}}{T_P \cdot 2^D} \\ STF_G = \dfrac{E_{SBF} + E_{DBF} + E_{SaF}}{T_P \cdot 2^D} \end{cases} \tag{11}$$

First, the error factor E is computed for each of the test scenarios according to the equation system (12):

$$\begin{cases} E_{SBF} = FC_{SBF} \cdot N_P \\ E_{DBF} = FC_{DBF} \cdot N_P \\ E_{SaF} = FC_{SaF} \cdot N_P \end{cases} \tag{12}$$

For each test scenario, N_P is the maximum number of test patterns generated by the statically deployed LFSR unit, namely 65.535. The associated results for single bit-flips can be seen in the equation set (13):

$$\begin{cases} E_{SBF1} = FC_{SBF1} \cdot N_P \\ E_{SBF2} = FC_{SBF2} \cdot N_P \\ E_{SBF3} = FC_{SBF3} \cdot N_P \\ E_{SBF4} = FC_{SBF4} \cdot N_P \\ E_{SBF5} = FC_{SBF5} \cdot N_P \\ E_{SBF1} = 93.95\% \cdot 65,535 = 61570.1325 \\ E_{SBF2} = 93.93\% \cdot 65,535 = 61557.0255 \\ E_{SBF3} = 93.92\% \cdot 65,535 = 61550.472 \\ E_{SBF4} = 93.91\% \cdot 65,535 = 61543.9185 \\ E_{SBF5} = 93.94\% \cdot 65,535 = 61563.579 \end{cases} \tag{13}$$

With regards to double bit-flips, the computed results are visible in the equation system (14):

$$\begin{cases} E_{DBF1} = FC_{DBF1} \cdot N_P \\ E_{DBF2} = FC_{DBF2} \cdot N_P \\ E_{DBF3} = FC_{DBF3} \cdot N_P \\ E_{DBF4} = FC_{DBF4} \cdot N_P \\ E_{DBF5} = FC_{DBF5} \cdot N_P \\ E_{DBF1} = 50.63\% \cdot 65,535 = 33180.3705 \\ E_{DBF2} = 50.24\% \cdot 65,535 = 32924.784 \\ E_{DBF3} = 50.23\% \cdot 65,535 = 32918.2305 \\ E_{DBF4} = 50.44\% \cdot 65,535 = 33055.854 \\ E_{DBF5} = 50.25\% \cdot 65,535 = 32931.3375 \end{cases} \tag{14}$$

Concerning stuck-at faults, since for each of the five test points, a 100% detection ratio is achieved, the error factor will always be $E = 65.535$.

Secondly, the *STF* parameter was calculated for the first three mathematical expressions from equation system (11), as presented in equation set (15):

$$\begin{cases} STF_{SBF1} = \dfrac{E_{SBF1}}{T_P \cdot 2^D} = \dfrac{61570.1325}{65,535} = 0.9395 \\ STF_{SBF2} = \dfrac{E_{SBF2}}{T_P \cdot 2^D} = \dfrac{61557.0255}{65,535} = 0.9393 \\ STF_{SBF3} = \dfrac{E_{SBF3}}{T_P \cdot 2^D} = \dfrac{61550.472}{65,535} = 0.9392 \\ STF_{SBF4} = \dfrac{E_{SBF4}}{T_P \cdot 2^D} = \dfrac{61543.9185}{65,535} = 0.9391 \\ STF_{SBF5} = \dfrac{E_{SBF5}}{T_P \cdot 2^D} = \dfrac{61563.579}{65,535} = 0.9394 \\ STF_{DBF1} = \dfrac{E_{DBF1}}{T_P \cdot 2^D} = \dfrac{33180.3705}{65,535} = 0.5063 \\ STF_{DBF2} = \dfrac{E_{DBF2}}{T_P \cdot 2^D} = \dfrac{32924.784}{65,535} = 0.5024 \\ STF_{DBF3} = \dfrac{E_{DBF3}}{T_P \cdot 2^D} = \dfrac{32918.2305}{65,535} = 0.5023 \\ STF_{DBF4} = \dfrac{E_{DBF4}}{T_P \cdot 2^D} = \dfrac{33055.854}{65,535} = 0.5044 \\ STF_{DBF5} = \dfrac{E_{DBF5}}{T_P \cdot 2^D} = \dfrac{32931.3375}{65,535} = 0.5025 \\ STF_{SaF} = \dfrac{65,535}{65,535} = 1 \end{cases} \tag{15}$$

Concerning the equation system (13), it is observable that the relation between the *STF* parameter for all targeted errors and the *FC* is given by the mathematical rule (16):

$$STF = \frac{FC}{100} \leftrightarrow FC = STF \cdot 100 \tag{16}$$

At this point, the graphical representations of the *STF* parameter can be generated for the single and double bit-flip errors, respectively, as presented in Fig. 6.

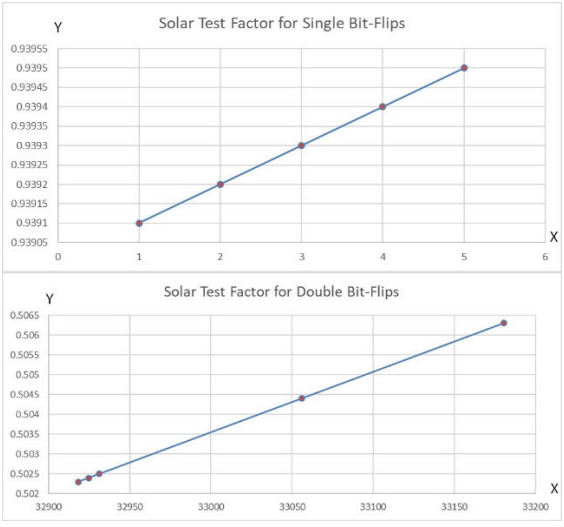

FIGURE 6. STF graphical representation for single and double bit-flip errors.

As observed in both scenarios, the graphical representations of the *STF* parameters exhibit a linear dependency with the number of detected errors since the computed test factor is directly proportional to the *FC*. Regarding the *SRF* parameter for single bit-flip errors, the computation can be executed by using the equation set (17):

$$\begin{cases} SRF_{SBF1} = e^{-STF_{SBF1}} \\ SRF_{SBF2} = e^{-STF_{SBF2}} \\ SRF_{SBF3} = e^{-STF_{SBF3}} \\ SRF_{SBF4} = e^{-STF_{SBF4}} \\ SRF_{SBF5} = e^{-STF_{SBF5}} \end{cases} \tag{17}$$

Thus, by solving the above equation system, the results are obtained, as presented in equation set (18):

$$\begin{cases} SRF_{SBF1} = e^{-STF_{SBF1}} = 0.39097 \\ SRF_{SBF2} = e^{-STF_{SBF2}} = 0.39094 \\ SRF_{SBF3} = e^{-STF_{SBF3}} = 0.39091 \\ SRF_{SBF4} = e^{-STF_{SBF4}} = 0.39086 \\ SRF_{SBF5} = e^{-STF_{SBF5}} = 0.39082 \end{cases} \tag{18}$$

Following, the mathematical relations of the *SRF* for double bit-flip errors are given by the equation system (19):

$$\begin{cases} SRF_{DBF1} = e^{-STF_{DBF1}} \\ SRF_{DBF2} = e^{-STF_{DBF2}} \\ SRF_{DBF3} = e^{-STF_{DBF3}} \\ SRF_{DBF4} = e^{-STF_{DBF4}} \\ SRF_{DBF5} = e^{-STF_{DBF5}} \end{cases} \tag{19}$$

The *SRF* results associated with the double bit-flip errors are provided in the equation system (20):

$$\begin{cases} SRF_{DBF1} = e^{-STF_{DBF1}} = 0.60501 \\ SRF_{DBF2} = e^{-STF_{DBF2}} = 0.60507 \\ SRF_{DBF3} = e^{-STF_{DBF3}} = 0.60502 \\ SRF_{DBF4} = e^{-STF_{DBF4}} = 0.60386 \\ SRF_{DBF5} = e^{-STF_{DBF5}} = 0.60272 \end{cases} \tag{20}$$

The graphical representations derived from equation sets (19) and (20) are presented in Fig. 7.

The *SRF* charts prove that the more hardware errors affect the solar tracking device, the more the reliability will have a downward tendency to reach the value 0, directly impacting the system's availability.

In the last part of this section, the global *SRF* is compared with and without error correction, showing that reduced fault coverage is essential in increasing the reliability factor of solar tracking systems. First, by using a modified metrics system that is adapted for three fault models (single bit-flip errors, double bit-flip errors, and stuck-at faults), as described in equation set (11), the *STF* parameter can be computed

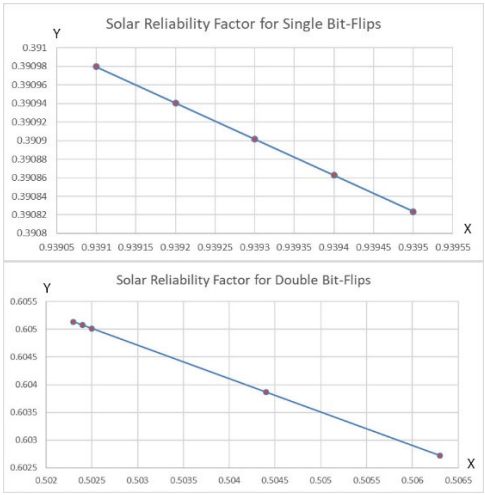

FIGURE 7. SRF graphical representation for single and double bit-flip errors.

without single bit-flip error correction as presented in (21):

$$\begin{cases} STF_{SBF} = \dfrac{E_{SBF}}{T_P \cdot 2^D} \\ STF_{DBF} = \dfrac{E_{DBF}}{T_P \cdot 2^D} \\ STF_{SaF} = \dfrac{E_{SaF}}{T_P \cdot 2^D} \\ STF_{SBF} = \dfrac{61,557}{65,535 \cdot 5} = \dfrac{61,557}{327,675} = 0,1878 \\ STF_{DBF} = \dfrac{33,002}{65,535 \cdot 5} = \dfrac{33,002}{327,675} = 0,1007 \\ STF_{SaF} = \dfrac{65,535}{65,535 \cdot 5} = \dfrac{65,535}{327,675} = 0,2 \\ STF_G = \dfrac{E_{SBF} + E_{DBF} + E_{SaF}}{T_P \cdot 2^D} \\ STF_G = \dfrac{61,557 + 33,002 + 65,535}{65,535 \cdot 5} = \dfrac{160,094}{327,675} \\ = 0.4885 \end{cases} \tag{21}$$

Consequently, the global *SRF* parameter will be computed with the equation system (22):

$$\begin{cases} SRF_{SBF} = e^{-STF_{SBF}} \\ SRF_{DBF} = e^{-STF_{DBF}} \\ SRF_{SaF} = e^{-STF_{SaF}} \\ SRF_{SBF} = e^{-0,1878} = 0.8287 \\ SRF_{DBF} = e^{-0,1007} = 0.9042 \\ SRF_{SaF} = e^{-0,2} = 0.8187 \\ SRF_G = e^{-STF_G} \\ SRF_G = e^{-0.4885} = \dfrac{1}{e^{0.4885}} = 0.6135 \end{cases} \tag{22}$$

The graphical representations of the global *STF* and *SRF* parameters are generated according to equation systems (21) and (22), as illustrated in Fig. 8.

Secondly, by using the same metric system, the single bit-flip correction depicted in the improved OBIST diagram

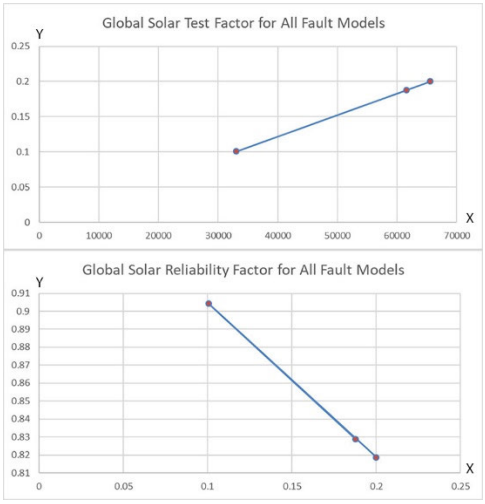

FIGURE 8. Global STF and SRF graphical representations without single-bit error correction.

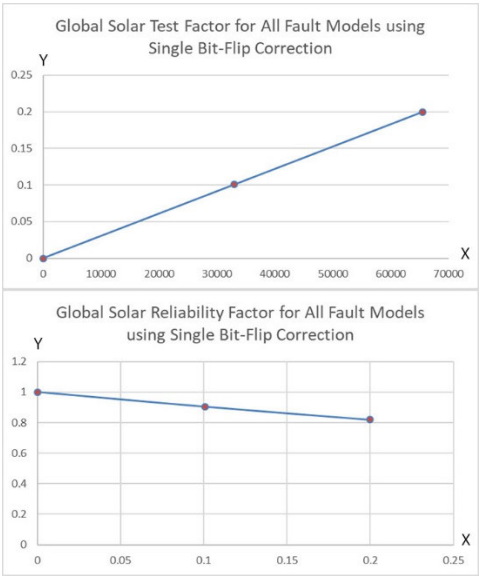

FIGURE 9. Global STF and SRF graphical representations without single-bit error correction.

from Fig. 4 can be applied and obtain the following results for the global *STF* parameter, as presented in equation set (23):

$$\begin{cases} STF_{SBF} = \dfrac{E_{SBF}}{T_P \cdot 2^D} \\ STF_{DBF} = \dfrac{E_{DBF}}{T_P \cdot 2^D} \\ STF_{SaF} = \dfrac{E_{SaF}}{T_P \cdot 2^D} \\ STF_{SBF} = \dfrac{0}{65,535 \cdot 5} = \dfrac{0}{327,675} = 0 \\ STF_{DBF} = \dfrac{33,002}{65,535 \cdot 5} = \dfrac{33,002}{327,675} = 0,1007 \\ STF_{SaF} = \dfrac{65,535}{65,535 \cdot 5} = \dfrac{65,535}{327,675} = 0,2 \\ STF_G = \dfrac{E_{SBF} + E_{DBF} + E_{SaF}}{T_P \cdot 2^D} \\ STF_G = \dfrac{33,002 + 65,535}{65,535 \cdot 5} = \dfrac{160,094}{327,675} = 0.3007 \end{cases} \tag{23}$$

The corresponding global *SRF* parameter will be then computed with equation set (20) using the results from (21), thus obtaining the results in equation system (24):

$$\begin{cases} SRF_{SBF} = e^{-STF_{SBF}} \\ SRF_{DBF} = e^{-STF_{DBF}} \\ SRF_{SaF} = e^{-STF_{SaF}} \\ SRF_{SBF} = e^{0} = 1 \\ SRF_{DBF} = e^{-0,1007} = 0.9044 \\ SRF_{SaF} = e^{-0,2} = 0.8192 \\ SRF_G = e^{-STF_G} \\ SRF_G = e^{-0.10} = \dfrac{1}{e^{0.10}} = 0.9048 \end{cases} \tag{24}$$

The graphical representations associated with equation sets (23) and (24) can be seen in Fig. 9.

By comparing Figs. 8 and 9, it is easy to observe that the global *SRF* parameter was increased, through single bit-flip error correction, by 47.48%, showing that the improved OBIST schematic efficiently extends the lifespan of modern solar tracking devices. As seen in the experimental results, compared to the standard probabilistic model presented in [30], the proposed metric system is more efficient regarding the reliability evaluation of automated systems.

VI. LIMITATIONS OF THIS STUDY AND FUTURE WORK

Regarding the scope of fault scenarios, this study has a few limitations, such as focusing primarily on single bit-flip errors and stuck-at-faults, particularly emphasizing the former. While it achieves 100% correction for single bit-flip errors, thereby reducing fault coverage to 0% and significantly increasing the SRF, the study does not extensively cover other types of faults that may affect solar tracking systems, such as bridging or open-circuit faults.

The OBIST architecture and its implementation are tailored to specific hardware components, including the Atmega328 Microcontroller Unit (MCU), Optocoupler LTV 847, and L298N Dual-H Bridges ICs. This specificity may limit the direct applicability of the proposed solution to solar tracking systems with different hardware configurations or newer components with different fault characteristics.

While the study acknowledges the impact of environmental conditions such as temperature, humidity, and prolonged sunlight exposure on the operation of solar tracking systems, it does not explicitly incorporate these factors into the reliability analysis or the OBIST architecture. The effects of these environmental factors on the long-term performance and fault occurrence in the system components could be significant. Also, the study provides a detailed theoretical and simulation-based analysis of the proposed OBIST architecture and its impact on the SRF. However, it needs more extensive experimental validation in real-world conditions, which would be crucial to comprehensively assess

the effectiveness of the OBIST architecture in improving the reliability of solar tracking systems over extended periods of operation.

The introduction of an improved OBIST architecture, while beneficial for reliability, may have implications for the overall cost and complexity of the solar tracking system. The study needs to provide a detailed cost-benefit analysis considering the additional hardware and computational resources required for the OBIST implementation versus the potential savings from reduced maintenance and increased system uptime.

Finally, this study presents a specific implementation of the OBIST architecture for a dual-axis solar tracker. The scalability of this approach to larger or more complex solar tracking systems and its adaptability to future advancements in solar tracking technology and fault diagnosis methods remain to be fully explored. Despite this work significantly improving the reliability of solar tracking devices through an innovative and enhanced OBIST architecture, addressing these limitations in future research could further enhance the practical applicability and effectiveness of reliability-enhancing strategies in solar tracking systems.

Regarding future work, the results of this study could have a profound impact on the development and adoption of the proposed methodology across various industries, particularly those reliant on solar energy technologies. For example, our methods could lead to more reliable and efficient solar tracking systems, potentially increasing the overall energy yield from solar installations. This could accelerate the adoption of solar energy, contributing to the transition towards more sustainable energy sources. Another example of future work is in precision agriculture, where solar-powered devices are increasingly used for various applications; the improved reliability of solar trackers could enhance the efficiency of solar-powered irrigation systems, drones, and other agricultural technologies. Finally, the methodology could be adapted for space applications, where reliability is critical. Solar trackers on satellites and space probes could benefit from increased fault tolerance, potentially extending mission lifespans.

VII. CONCLUSION

This paper introduces an OBIST architecture that incorporates an LFSR as a TPG and a MISR for gathering results, designed to test dual-axis solar tracking equipment consisting of an Optocoupler, an Arduino UNO, and two L298N Dual-H Bridges ICs. The investigation reveals that all four CUTs are prone to hardware faults, necessitating the implementation of software and hardware solutions within the OBIST framework. The architecture is enhanced with bit-flip correction capabilities through extended Hamming codes, effectively locating and repairing single bit-flip errors. This advancement, alongside a modified metrics system that reduces the fault coverage of single bit-flip errors to 0%, significantly increases solar tracking devices' SRF. Experimental results demonstrate the efficiency of the software in injecting test vectors and collecting MISR device signatures, achieving comprehensive coverage rates for various error types and demonstrating a total global error coverage of 72.15%. Implementing the improved OBIST strategy and metrics system results in a 100% correction rate for single bit-flip errors, eliminating fault coverage and enhancing the overall SRF by 47.48%. This research underscores the importance of maintenance operations in reliability assessment and establishes the OBIST approach as a cost-effective solution with high coverage efficiency.

By reducing maintenance and downtime costs due to improved reliability, industries might find the cost-benefit ratio favorable, leading to broader adoption. The methodology could also influence the development of new solar tracking system reliability standards, pushing manufacturers to integrate similar OBIST architectures into their products.

ABBREVIATIONS AND SYMBOLS

Abbreviation/ Symbol	Definition
SRF	Solar Reliability Factor
OBIST	Online Built-In Self-Test
PV	Photovoltaic
MCU	Microcontroller Unit
DC	Direct Current
ANN	Artificial Neural Networks
SoC	System-on-Chip
MM	Mathematical Morphology
FTC	Fault Tolerant Control
DFT	Design for Testing
UN	United Nations
API	Application Programming Interface
IC	Integrated Circuit
ESD	Electrostatic Discharge
CUTs	Circuits Under Test
STF	Solar Test Factor
BIST	Built-In Self-Test
CMOS	Complementary Met-al-Oxide-Semiconductor
RAM	Random Access Memory
ReBISR	Reconfigurable Built-In Self-Repair
DeBISR	Dedicated Built-In Self-Repair
BIRA	Built-In Redundancy Analysis
ReBIRA	Reconfigurable BIRA
RPU	Reconfigurable Processing Unit
CCs	Clock Cycles
RTR	Runtime Reconfiguration
CTR	Compile-time Reconfiguration
LFSR	Linear Feedback Shift Register
CLFSR	Configurable Linear Feedback Shift Register
CRC	Cyclic Redundancy Check
TPG	Test Pattern Generator
ORA	Output Response Analyzer
ROM	Read-Only Memory

RTL	Register Transfer Level
MSIC	Multiple Input Single Input Change
CSTP	Circular Self-Test Path
DCSTP	Deterministic CSTP
SVM	Support Vector Machine
TCT	Total-Cross-Tied
PSC	Partial Shading Conditions
MPP	Maximum Power Points
MPPT	Metaheuristic Maximum Power Point Tracking
EGWO	Enhanced Grey Wolf Optimization
MPA	Marine Predator Algorithm
8 S	Eight Series-Connected Panels
OCPD	Overcurrent Protection Devices
GFPD	Ground Fault Protection Devices
KNN	K-Nearest Neighbors
TR	Top Right
TL	Top Left
BR	Bottom Right
BL	Bottom Left
AVR	Average Voltage Right
AVL	Average Voltage Left
AVT	Average Voltage Top
AVD	Average Voltage Down
TAV	Total Average Value
RR	right rotation
LR	left rotation
UR	upward rotation
DR	downward rotation
A/I	active/inactive states
TMR	Triple Module Redundancy
PTM	Probabilistic Transfer Matrix
ITM	Ideal Transfer Matrix
ECCs	Error Correcting Codes
SECDED	Single-Error Correction and Double-Error Detection
SEC-DED-TAED	Single Error Detection, Double Error Detection, Triple Adjacent Error Detection
SISEC	Soft Information Single Error Correction
PRPG	Pseudo-Random Pattern Generator
MISR	Multiple-Input Signature Register
ns	nanoseconds
nm	nanometer
W	Watts
Wh	Watt-hour
mW	milliWatts
r	*single fault probability*
G1	Gate 1
G2	Gate 2
p	probability
k	associated message length
FC_{SBF}	epresents the fault coverage of detected single bit-flip errors
FC_{DBF}	stands for the fault coverage of detected double bit-flip errors
FC_{SaF}	represents the fault coverage for detected stuck-at faults
FC_G	represents the global fault coverage for both the single bit-flips and single stuck-at faults
T_V	represents the number of test vectors
T_P	designates the number of tested patterns
E	a hardware error factor
FC	is the hardware fault coverage
N_P	represents the total number of test patterns
D	represents the number of D flip-flops employed in the MISR s hardware design
STF	Solar Test Factor

REFERENCES

[1] A. Adouni, K. Elmellah, D. Chariag, and L. Sbita, "DC–DC converter fault diagnostic in PV system," in *Proc. Int. Conf. Green Energy Convers. Syst. (GECS)*, Mar. 2017, pp. 1–7, doi: 10.1109/GECS.2017.8066136.

[2] K. Kasar and P. C. Tapre, "A new fast detection module for short-circuit current detection in PV grid system," in *Proc. 2nd Int. Conf. Inventive Syst. Control (ICISC)*, Jan. 2018, pp. 468–472, doi: 10.1109/ICISC.2018.8399116.

[3] P. Jain, J. Xu, S. K. Panda, J. Poon, C. Spanos, and S. R. Sanders, "Fault diagnosis via PV panel-integrated power electronics," in *Proc. IEEE 17th Workshop Control Model. Power Electron. (COMPEL)*, Jun. 2016, pp. 1–6, doi: 10.1109/COMPEL.2016.7556716.

[4] M. Weerasekara, M. Vilathgamuwa, and Y. Mishra, "Detection of high impedance faults in PV systems using mathematical morphology," in *Proc. IEEE Int. Conf. Ind. Electron. Sustain. Energy Syst. (IESES)*, Jan. 2018, pp. 357–361, doi: 10.1109/IESES.2018.8349902.

[5] S. Thukral, O. P. Mahela, and B. Kumar, "Detection of transmission line faults in the presence of solar PV system using Stockwell's transform," in *Proc. IEEE 7th Power India Int. Conf. (PIICON)*, Nov. 2016, pp. 1–6, doi: 10.1109/POWERI.2016.8077304.

[6] V. Schmidt, A. Luccioni, S. Karthik Mukkavilli, N. Balasooriya, K. Sankaran, J. Chayes, and Y. Bengio, "Visualizing the consequences of climate change using cycle-consistent adversarial networks," 2019, *arXiv:1905.03709*.

[7] *UN Sustainable Development Goals*. [Online]. Available: https://www.un.org/sustainabledevelopment/sustainable-development-goals/

[8] R. Schwartz, J. Dodge, N. A. Smith, and O. Etzioni, "Green AI," 2019, *arXiv:1907.10597*.

[9] P. Rani, O. Singh, and S. Pandey, "An analysis on Arduino based single axis solar tracker," in *Proc. 5th IEEE Uttar Pradesh Sect. Int. Conf. Electr., Electron. Comput. Eng. (UPCON)*, Nov. 2018, pp. 1–5, doi: 10.1109/UPCON.2018.8596874.

[10] T. Kaur, S. Mahajan, S. Verma, and J. Gambhir, "Arduino based low cost active dual axis solar tracker," in *Proc. IEEE 1st Int. Conf. Power Electron., Intell. Control Energy Syst. (ICPEICES)*, Jul. 2016, pp. 1–5, doi: 10.1109/ICPEICES.2016.7853398.

[11] Y. Bekakra, L. Zellouma, and H. Serhoud, "Design and implementation of a solar tracker system with dual axis for photovoltaic panels in El Oued region of Algeria," in *Proc. 1st Int. Conf. Sustain. Renew. Energy Syst. Appl. (ICSRESA)*, Dec. 2019, pp. 1–6, doi: 10.1109/ICSRESA49121.2019.9182599.

[12] A. K. Suria and R. M. Idris, "Dual-axis solar tracker based on predictive control algorithms," in *Proc. IEEE Conf. Energy Convers. (CENCON)*, Oct. 2015, pp. 238–243, doi: 10.1109/CENCON.2015.7409546.

[13] A. Y. Pratama, A. Fauzy, and H. Effendi, "Performance enhancement of solar panel using dual axis solar tracker," in *Proc. Int. Conf. Electr. Eng. Informat. (ICEEI)*, Jul. 2019, pp. 444–447, doi: 10.1109/ICEEI47359.2019.8988902.

[14] R. Rotar, S. L. Jurj, F. Opritoiu, and M. Vladutiu, "Position optimization method for a solar tracking device using the cast-shadow principle," in *Proc. IEEE 24th Int. Symp. Design Technol. Electron. Packag. (SIITME)*, Oct. 2018, pp. 61–70.

[15] M. Tadeusiewicz and S. Halgas, "Parametric fault diagnosis of very high-frequency circuits containing distributed parameter transmission lines," *Electronics*, vol. 10, no. 5, p. 550, Feb. 2021, doi: 10.3390/electronics10050550.
[16] M. Lee, S. Han, J. Sim, H. Park, and B. Kim, "A 10-GHz multi-purpose reconfigurable built-in self-test circuit for high-speed links," in *Proc. IEEE Asian Solid-State Circuits Conf. (A-SSCC)*, Seoul, (South) Korea, Nov. 2017, pp. 73–76, doi: 10.1109/ASSCC.2017.8240219.
[17] A. S. Nair and P. L. Bonifus, "An efficient built-in self-repair scheme for multiple RAMs," in *Proc. 2nd IEEE Int. Conf. Recent Trends Electron., Inf. Commun. Technol. (RTEICT)*, Bangalore, India, May 2017, pp. 2076–2080, doi: 10.1109/RTEICT.2017.8256965.
[18] T. Tseng, J. Li, C. Hsu, A. Pao, K. Chiu, and E. Chen, "A reconfigurable built-in self-repair scheme for multiple repairable RAMs in SOCs," in *Proc. IEEE Int. Test Conf.*, Santa Clara, CA, USA, Oct. 2006, pp. 1–9, doi: 10.1109/TEST.2006.297688.
[19] K. George and C.-I.-H. Chen, "Logic built-in self-test for core-based designs on system-on-a-chip," *IEEE Trans. Instrum. Meas.*, vol. 58, no. 5, pp. 1495–1504, May 2009, doi: 10.1109/TIM.2008.2009417.
[20] M. Elbadri, V. Groza, R. Abielmona, and M. Assaf, "A reconfigurable processing unit for digital circuit testing using built-in self-test techniques," in *Proc. IEEE Instrum. Meas. Technol. Conf.*, Sorrento, Italy, Apr. 2006, pp. 239–244, doi: 10.1109/IMTC.2006.328406.
[21] V. Groza, R. Abielmona, M. H. Assaf, M. Elbadri, M. El-Kadri, and A. Khalaf, "A self-reconfigurable platform for built-in self-test applications," *IEEE Trans. Instrum. Meas.*, vol. 56, no. 4, pp. 1307–1315, Aug. 2007, doi: 10.1109/TIM.2007.900134.
[22] S. Wane, M. Ranaivoniarivo, B. Elkassir, C. Kelma, and P. Gamand, "Towards cognitive built-in-self-test (BIST) for reconfigurable on-chip applications, and contact-less measurement," in *Proc. IEEE Radio Freq. Integr. Circuits Symp.*, Baltimore, MD, USA, Jun. 2011, pp. 1–4, doi: 10.1109/RFIC.2011.5940716.
[23] S. Mishra, R. R. Tripathi, and D. Kr. Tripathi, "Implementation of configurable linear feedback shift register in VHDL," in *Proc. Int. Conf. Emerg. Trends Electr. Electron. Sustain. Energy Syst. (ICETEESES)*, Sultanpur, India, Mar. 2016, pp. 342–346, doi: 10.1109/ICETEESES.2016.7581406.
[24] K. N. Devika and R. Bhakthavatchalu, "Design of efficient programmable test-per-scan logic BIST modules," in *Proc. Int. Conf. Microelectronic Devices, Circuits Syst. (ICMDCS)*, Vellore, India, Aug. 2017, pp. 1–6, doi: 10.1109/ICMDCS.2017.8211609.
[25] V. S. Kumar and J. Mohan, "Multiple single input change test vector for BIST schemes," in *Proc. Int. Conf. Green Comput. Commun. Elect. Eng. (ICGCCEE)*, Coimbatore, India, 2014, pp. 1–4, doi: 10.1109/ICGCCEE.2014.6922320.
[26] Y. Ouahab, R. Rashidzadeh, and R. Muscedere, "A secure scan chain using a phase locking system and a reconfigurable LFSR," in *Proc. IEEE 30th Can. Conf. Electr. Comput. Eng. (CCECE)*, Windsor, ON, Canada, Apr. 2017, pp. 1–4, doi: 10.1109/CCECE.2017.7946810.
[27] K. Wen, Y. Hu, and X. Li, "Deterministic circular self test path," *Tsinghua Sci. Technol.*, vol. 12, no. 1, pp. 20–25, Jul. 2007, doi: 10.1016/S1007-0214(07)70078-0.
[28] S. L. Jurj, R. Rotar, F. Opritoiu, and M. Vladutiu, "Online built-in self-test architecture for automated testing of a solar tracking equipment," in *Proc. IEEE Int. Conf. Environ. Electr. Eng. IEEE Ind. Commercial Power Syst. Eur.*, Jun. 2020, pp. 1–7, doi: 10.1109/EEEIC/ICPSEurope49358.2020.9160850.
[29] H. Lim, S. Jang, S. Kim, and S. Kang, "An efficient scan chain diagnosis for stuck-at and transition faults," in *Proc. Int. SoC Design Conf. (ISOCC)*, Oct. 2019, pp. 295–296, doi: 10.1109/ISOCC47750.2019.9078517.
[30] R. B. Schivittz, D. T. Franco, C. Meinhardt, and P. F. Butzen, "A probabilistic model for stuck-on faults in combinational logic gates," in *Proc. 17th Latin-American Test Symp. (LATS)*, Apr. 2016, pp. 39–44, doi: 10.1109/LATW.2016.7483337.
[31] A. Sánchez-Macián, P. Reviriego, and J. A. Maestro, "Hamming SEC-DAED and extended Hamming SEC-DED-TAED codes through selective shortening and bit placement," *IEEE Trans. Device Mater. Rel.*, vol. 14, no. 1, pp. 574–576, Mar. 2014, doi: 10.1109/TDMR.2012.2204753.
[32] J. Metzner, "Correction of two (or often more) vector symbol errors with the outer structure of a Hamming single error correcting code," *IEEE Commun. Lett.*, vol. 18, no. 12, pp. 2069–2072, Dec. 2014, doi: 10.1109/LCOMM.2014.2363665.
[33] R. Rotar, S. L. Jurj, F. Opritoiu, and M. Vladutiu, "Fault coverage-aware metrics for evaluating the reliability factor of solar tracking systems," *Energies*, vol. 14, no. 4, p. 1074, Feb. 2021, doi: 10.3390/en14041074.
[34] S. L. Jurj, R. Rotar, F. Opritoiu, and M. Vladutiu, "Improving the solar reliability factor of a dual-axis solar tracking system using energy-efficient testing solutions," *Energies*, vol. 14, no. 7, p. 2009, Apr. 2021, doi: 10.3390/en14072009.
[35] R. Rotar, S. L. Jurj, R. Susany, F. Opritoiu, and M. Vladutiu, "Global energy production computation of a solar-powered smart home automation system using reliability-oriented metrics," *Energies*, vol. 14, no. 9, p. 2541, Apr. 2021, doi: 10.3390/en14092541.
[36] E. Collins, M. Dvorack, J. Mahn, M. Mundt, and M. Quintana, "Reliability and availability analysis of a fielded photovoltaic system," in *Proc. 34th IEEE Photovoltaic Spec. Conf. (PVSC)*, Philadelphia, PA, USA, Jun. 2009, pp. 2316–2321, doi: 10.1109/PVSC.2009.5411343.
[37] K. Natarajan, P. B. Kumar, and V. Kumar, "Fault detection of solar PV system using SVM and thermal image processing," *Int. J. Renew. Energy Res.*, vol. 10, no. 3, pp. 967–977, 2020.
[38] S. Alwar, D. Samithas, M. S. Boominathan, P. K. Balachandran, and L. Mihet-Popa, "Performance analysis of thermal image processing-based photovoltaic fault detection and PV array reconfiguration—A detailed experimentation," *Energies*, vol. 15, no. 22, p. 8450, Nov. 2022, doi: 10.3390/en15228450.
[39] M. Patil, B. P. Kumar, and N. Karuppiah, "Detection and location of faults in photo VoltaicSystems," in *Proc. 2nd Int. Conf. Smart Technol. Syst. Next Gener. Comput. (ICSTSN)*, Villupuram, India, Apr. 2023, pp. 1–6, doi: 10.1109/ICSTSN57873.2023.10151647.
[40] S. Kumar Vankadara, S. Chatterjee, and P. K. Balachandran, "Applications of metaheuristic algorithms for MPPT under partial shaded condition in PV system," in *Proc. 4th Int. Conf. Emerg. Technol. (INCET)*, Belgaum, India, May 2023, pp. 1–5, doi: 10.1109/INCET57972.2023.10170183.
[41] S. Ganesan, P. W. David, P. Murugesan, and P. K. Balachandran, "Solar photovoltaic system performance improvement using a new fault identification technique," *Electric Power Compon. Syst.*, vol. 52, no. 1, pp. 42–54, Jan. 2024, doi: 10.1080/15325008.2023.2237013.
[42] G. M. El-Banby, N. M. Moawad, B. A. Abouzalm, W. F. Abouzaid, and E. A. Ramadan, "Photovoltaic system fault detection techniques: A review," *Neural Comput. Appl.*, vol. 35, no. 35, pp. 24829–24842, Dec. 2023, doi: 10.1007/s00521-023-09041-7.

SORIN LIVIU JURJ received the Ph.D. (Dr.-Ing.) degree in computers and information technology from the Politehnica University of Timisoara, Romania, in 2020, where his research focused on powering and evaluating deep learning-based systems using green energy. Then, he worked for two years as a Postdoctoral Researcher with the German Aerospace Center (DLR e.V.), Institute of Systems Engineering for Future Mobility, Germany, where his research focused on developing methods for integrating knowledge into artificial intelligence (AI) algorithms used in autonomous vehicles to increase their safety and sustainability. Currently, he is a Researcher in neuromorphic computing with the Institute for Print and Media Technology, Chemnitz University of Technology, Germany, where his research focuses on designing neuromorphic 3D circuits using printed memristors.

RAUL ROTAR received the B.Sc. degree in automation and applied informatics from the Aurel Vlaicu University of Arad, Romania, in 2015, and the M.Sc. degree in computer engineering from the Politehnica University of Timisoara, Romania, in 2017, where he is currently pursuing the Ph.D. degree in computers and information technology. His research interest includes developing energy-efficient testing solutions for automation systems to increase their safety and dependability.