applied sciences

MDPI

Review

A Review of Test Stimulus Compression Methods for Ultra-Large-Scale Integrated Circuits

Liang Zhou [1], Daming Yang [2,*], Lei Chen [2], Wei Zhuang [2], Shiyuan Zhang [2] and Yuanyuan Xiong [3]

[1] National Elite Institute of Engineering, Northwestern Polytechnical University, Xi'an 710072, China
[2] Beijing Microelectronics Technology Institute, Beijing 100076, China
[3] China Astronautics Standards Institute, Beijing 100166, China
[*] Correspondence: dada19930921@126.com

Abstract: With the development of system-on-chip (SoC) and chiplet technology in the post-Moore era, an increasing number of chiplets are being integrated into a single chip. Consequently, the functions and complexity that can be realized are growing daily. Simultaneously, the volume of test data required for ultra-large-scale integrated circuits (ULSIs) has risen significantly. However, traditional automatic test equipment (ATE) is constrained by its data storage and bandwidth limitations, and its long technology iteration cycle. These cannot keep pace with the rapid development of design technology. This discrepancy leads to challenges in ULSI testing, such as excessively long test time and difficulties in completing the tests. Test compression technology can effectively address these issues by reducing the performance requirements of the test equipment, which in turn can lower test costs. This paper summarizes the classifications of chip test compression technology and, based on their current development, provides a detailed analysis of key technologies. It includes test compression-oriented coding methods, optimization of scan chain structures, and enhancements in coding for compression efficiency. Finally, a forward-looking perspective on the development of chip test compression technology is presented. The aim is to offer a reference for subsequent research in this field and related areas, as well as to provide technical support for the advancement of ULSI testing in the post-Moore era.

Keywords: integrated circuits; testing technique; test compression

Citation: Zhou, L.; Yang, D.; Chen, L.; Zhuang, W.; Zhang, S.; Xiong, Y. A Review of Test Stimulus Compression Methods for Ultra-Large-Scale Integrated Circuits. *Appl. Sci.* **2024**, *14*, 10769. https://doi.org/10.3390/app142310769

Academic Editor: Alexander Barkalov

Received: 8 October 2024
Revised: 11 November 2024
Accepted: 18 November 2024
Published: 21 November 2024

Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

As a strategic industry for the development of the national economy and technology, integrated circuits have become a vital support for ensuring economic security and information security. Integrated circuits are diverse in type and widely applied, encompassing key areas such as military, telecommunications, aerospace, and consumer electronics. They serve as the cornerstone for accelerating modernization and the core technology for empowering the upgrade of traditional industries. In recent years, with the application of advanced integrated circuit design and manufacturing technologies, the scale of integrated circuit design has been expanding [1]. Karl [2] has provided the development trend of microprocessors over the past 48 years and the history of integrated circuit manufacturing processes in the last two decades, as shown in Figure 1. As can be seen from the figure, with the empowerment of new materials and processes, the number of transistors integrated in integrated circuits has shown exponential growth.

Affected by factors such as manufacturing design complexity, process precision, material purity, and operator handling, risks exist at every step of the integrated circuit industry chain. Each step has the potential to introduce irreversible physical defects into the product, leading to circuit failures and subsequently causing integrated circuit failure. Integrated circuit testing can ensure the reliability of batch products by eliminating non-conforming

products. As an effective method to guarantee the reliability of integrated circuits, integrated circuit testing technology is facing unprecedented challenges, mainly encompassing the following four aspects.

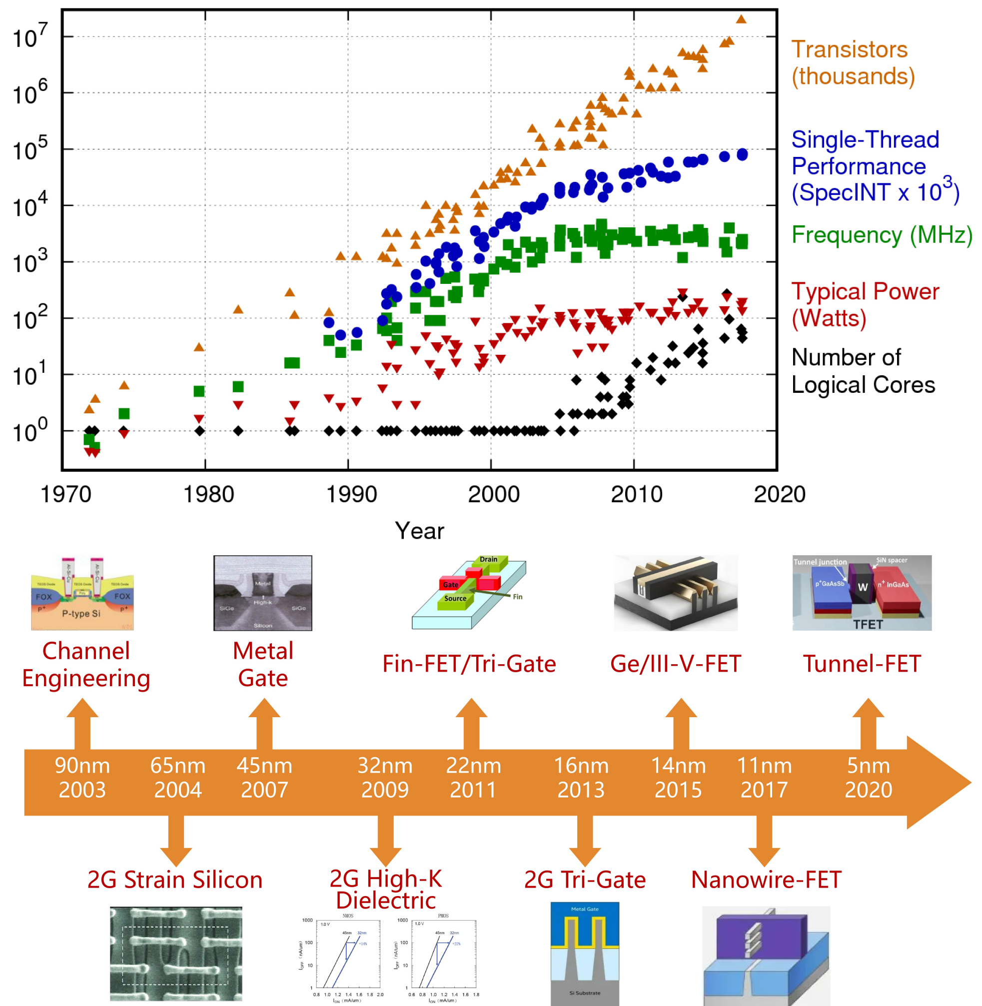

Figure 1. Trends in microprocessor development and the evolution of manufacturing processes [2].

1.1. Limited Testing Resources

With the advancement of integration technology, in order to meet the requirements of shortening development cycles and reducing costs, a variety of Intellectual Property (IP) cores are integrated into a single integrated circuit, forming an SoC device. Due to the continuous increase in the scale of integrated circuits and the addition of functions, the types of integrated IP cores are diverse. Each type of IP core has different testing conditions, which undoubtedly increases the complexity of the integrated circuit testing schemes. As the number of integrated cores on the chip continues to increase, while the number of I/O pins remains largely unchanged, accessing the internal circuits during integrated circuit testing becomes extremely difficult. Considering that IP cores do not have packaged physical pins, in order to ensure the effective transmission of test stimuli and test responses, it is necessary to design dedicated Test Access Mechanisms (TAM) and Test Wrappers (TW) for the IP cores to be tested. The transmission between the TW and the test interface adheres to the IEEE P1500 [3] and P1450 [4] standards. When designing TAM and TW, a significant amount of testing resources is required, presenting new challenges for the effective utilization of testing resources [5].

1.2. Increase in Testing Costs

As chip integration technology progresses, there has been a significant rise in the number of transistors, accompanied by a proportional increase in the volume of test data required. These test data need to be input into the chip through the test channels on the ATE. However, the number of I/O test channels on the ATE, the I/O transmission bandwidth, the operating frequency, and the depth of data storage are all limited. According to the International Technology Roadmap for Semiconductors (ITRS) report [6], the operating frequency of integrated circuits has averaged an increase of 30% over the past 30 years, while the corresponding growth rate for ATE has only been 12%. As for the throughput, in 2007, testing GDDR5 at 5 Gbps remained a challenge for ATE [7]. However, by 2023, ATE has developed testing solutions for PCIe 6.0 at 32 Gbps [8]. Current ATE is capable of addressing previous testing issues, yet new challenges persist, such as the testing of the upcoming PCIe 7.0. The performance improvement of ATE lags far behind the development of integrated circuit performance, leading to extended testing times for integrated circuits and increased demand for ATE test data storage, which in turn further increases the testing costs. For example, in the context of the emerging 2 nm technology, the application of sophisticated algorithms, such as March C- for memory testing, along with complex fault models including resistive-open and bridge faults, is crucial for precise fault detection and localization. These enhanced testing criteria result in a substantial increase in test duration. Specifically, a 2 nm chip requires eight times the test time compared to a 14 nm chip, which is nearly 800 s when assessed against the same PPA (Performance, Power, Area) criteria [9].

1.3. Increase in Test Power Consumption

The design of scan circuits can effectively enhance the controllability and observability of the Circuit Under Test (CUT). However, correspondingly, during the process of test stimulus input into the scan circuits, the flip-flops within the scan chains generate additional test power consumption due to their charging and discharging operations [10]. This test power consumption can impact the reliability and operational performance of the chip. During testing, in order to reduce test time, the switching activity of circuit nodes within the chip is significantly increased. This results in test power consumption that is substantially higher than that of normal operating modes [11]. Excessive test power consumption not only affects the quality of chip testing but also introduces issues of heat dissipation, thereby reducing test efficiency. When the heat generated by increased test power consumption exceeds the thermal tolerance of the chip design, it can potentially lead to chip failure or permanent damage [12]. Especially for emerging devices, e.g., spintronic and SiNW FETs devices. High temperatures can lead to instability of the spin state, thereby affecting spintronic device reliability and longevity [13]. SiNW FETs have a higher density of surface states and defect states, which can vary with temperature changes, affecting carrier scattering and recombination [14]. As the scale of chip integration continues to expand, the corresponding chip cooling measures remain limited, making the issue of test power consumption increasingly prominent and a significant factor affecting the quality and efficiency of chip testing.

Among the aforementioned challenges, the increase in testing costs has garnered substantial attention and concern from researchers. Increased transistor density necessitates extensive structural, functional, and parametric tests. The iterative updates of ATE struggle to keep pace with the advancements in integrated circuit manufacturing processes. During testing, the number of test channels and the test bandwidth are constrained by the physical limitations of the test equipment, resulting in limited data transmission rates [15,16]. Given the larger number of test patterns required, efficient test stimulus compression techniques are crucial. Test stimulus compression methods, as an effective means to shorten test time, directly impact the efficiency and cost of integrated circuit testing [17] and have been widely applied in chips with multi-chiplet integration. The advantages of its application can be summarized as follows:

1. **Optimization of Test Data Volume:** Reducing the volume of test data is crucial for decreasing testing costs and increasing test efficiency. Through test stimulus compression, the number of required test vectors and the volume of data can be significantly reduced without compromising the quality of chiplet testing.
2. **Reduction of Test Power Consumption:** Test stimulus compression can decrease power consumption during testing by reducing the transmission of test vectors, which is particularly important for 3D heterogeneous integrated circuits with multi-chiplet integration and emerging devices such as SiNW FETs and spintronic devices.
3. **Improvement of Test Efficiency:** In chiplet design, test stimulus compression can reduce the transmission and processing time of test vectors, thereby accelerating the test speed and enhancing overall test efficiency.
4. **Strong Portability:** Chiplet technology allows for the "LEGO-like" assembly of modules with different functions. Test stimulus compression methods need to be adaptable to this modular design, providing flexible test solutions for different chiplet combinations.

In this paper, test stimulus compression methods are categorized into three types based on their compression principles: coding-based compression methods, scan chain structure optimization-based compression methods, and compression methods that enhance the efficiency of coding compression. Compared to previous reviews, as shown in Table 1, the contributions of this paper are as follows:

1. Based on the existing classification structure, this paper summarizes the latest research and proposes a new category, Enhancing Encoding Compression Efficiency, for subsequent scholarly research and reference.
2. The paper analyzes the four mainstream compression methods from five aspects—Principle, Dominance, Limitation, Application Scenarios, and Compressed Objects—providing a reference for subsequent scholars in selecting compression methods for testing.
3. From the perspective of the interplay between the four methods, as shown in Figure 2, the paper organizes the application scenarios and compressed objects of these methods, offering a reference for scholars to adopt a hybrid approach of multiple compression methods.
4. The paper thoroughly reviews and discusses the latest research under this classification method and presents the advantages and development of various sub-methods within this classification framework.

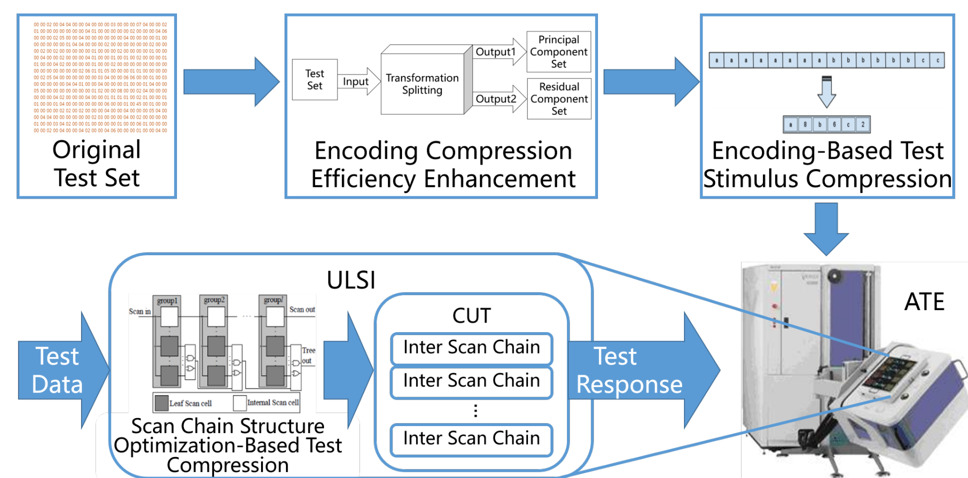

Figure 2. Schematic diagram of test stimulus compression techniques.

Table 1. Comparative analysis of previous reviews and this paper.

Article	Year	No. of Ref.	Compressed Objects	Researches Overview
[18]	2006	35	Code-based, Linear-Decompression-based, and Broadcast-Scan-based	Yes
[19]	2006	49	Code-based, Linear-Decompression-based, and Broadcast-Scan-based	Yes
[20]	2009	24	Code-based	Yes
[21]	2010	6	Code-based, Linear-Decompression-based, and Broadcast-Scan-based	No
[10]	2013	38	LFSR Based, BIST Based, Low Power Scan Based, and Low-Power DFT	Yes
[22]	2020	29	Code-based, Linear-Decompression-based, and Broadcast-Scan-based	Yes
This paper	-	102	Code-based, Scan Chain Structure Optimization-Based, and Enhancing Encoding Compression Efficiency	Yes

2. Overview of Test Stimulus Compression Techniques

2.1. Encoding-Based

Encoding-based test stimulus compression techniques initially divide the original test set into distinct symbol blocks, which are then represented by code words. These code words constitute new test vectors, achieving the objective of compressing the original test data [23]. During the decoding process, the decoder interprets the original symbol blocks based on the code words to reconstruct the test vectors [24]. The length of the symbol blocks in the original test set and the compressed code words can be either fixed or variable. Consequently, encoding-based test stimulus compression methods can be categorized into four types, which are fixed-length to fixed-length, fixed-length to variable-length, variable-length to variable-length, and variable-length to fixed-length [20].

2.2. Scan Chain Structure Optimization-Based

2.2.1. Linear Decompression Architecture

Stimulus compression methods based on a linear decompression architecture typically utilize a linear decompression mechanism composed of XOR gates and flip-flops to expand and populate the test data output from the ATE into the scan chains within the CUT [18].

The linear decompression architecture effectively leverages the large number of don't-care bits in the test vector set to achieve test compression [25]. However, this method is ineffective for the fixed bits within the test vectors. Consequently, the compression efficiency of such stimulus compression methods is limited by the number of fixed bits in the test vectors, and the compressed test data volume must be at least equal to the number of fixed bits in the original test set [26]. To achieve higher test efficiency, researchers have combined the linear decompression architecture with nonlinear encoding compression to deeply compress both the fixed and don't-care bits within the test vectors [27].

2.2.2. Broadcast Scan

When multiple sub-modules within the CUT have test vectors with interdependencies, test stimulus compression can be achieved through the optimization of a broadcast scan architecture [19]. This method involves broadcasting the same test data to multiple scan chains, effectively using a single test channel to drive multiple scan chains. The broadcast scan architecture includes fan-out circuits, which are typically implemented in hardware using a shared scan input structure [28]. This approach allows the same set of test vectors

to test different sub-circuits within the CUT, significantly enhancing test efficiency and achieving test vector compression [29]. However, due to the varying fault models of different sub-circuits, the fault coverage of this technique is limited, necessitating the input of additional test vectors in a serial scan mode to ensure comprehensive coverage [30].

2.3. Enhancing Encoding Compression Efficiency

Test compression methods aimed at enhancing encoding compression efficiency involve decomposing the original test set into multiple sub-component sets, followed by compression processing of these sub-component sets [31]. The compression process maps the original test set to a representation domain of sub-component sets that is more amenable to compression, after which encoding or other compression methods are applied to the sub-component sets [32]. This approach effectively utilizes the extensive contiguous data blocks within the original test set, allowing the encoding-based test stimulus compression methods to more efficiently partition symbol blocks and achieve higher compression efficiency.

2.4. Comparative Analysis of Test Stimulus Compression Techniques

Encoding-based test compression methods leverage the correlation among the fixed bits in test vectors to achieve compression of the test set. Since these methods compress the test set itself, they impose no constraints on Automatic Test Pattern Generation (ATPG) and can be widely applied to various integrated circuits without requiring knowledge of the structure of the CUT. However, the efficiency of encoding-based compression is limited by the number of don't-care bits in the test vectors; when there are many don't-care bits, the compression efficiency is reduced.

The linear decompression structure based on combinational logic requires minimal control logic, making it easy to design and implement in hardware. This method utilizes don't-care bits in the test vectors to generate free variables for each scan slice during each clock cycle. The compression efficiency is limited when there are a large number of fixed bits in the scan slice. The linear decompression structure based on sequential logic can effectively overcome this limitation by selectively using free variables from the previous cycle when compressing the current cycle's scan slice. Thereby, it can enhance the flexibility of the compression encoding and increase the likelihood of successful compression.

Broadcast scan-based test compression technology broadcasts the same test data to multiple scan cells within the CUT. Compared to the linear decompression structure method, it has a shorter test time but can encode fewer test vectors and has lower encoding flexibility. This issue can be mitigated by constraining ATPG through static or dynamic reconstruction. Static reconstruction requires less control information and has relatively lower encoding flexibility, while dynamic reconstruction requires more control information and offers relatively higher encoding flexibility.

Among the two stimulus compression methods based on scan chain structure optimization, the compression method based on linear decompression structure requires solving a set of equations corresponding to the circuit relationships, while the broadcast scan-based compression method involves broadcasting test vectors to multiple scan chains. The application of both methods presupposes that the test personnel are familiar with the structure of the CUT. However, in general, third-party IP core suppliers do not provide internal structural information. Under these circumstances, test vector generation and fault simulation tools cannot be used normally, and the correctness and comprehensiveness of the analysis results cannot be guaranteed. In such cases, the only viable option is the encoding-based compression method.

However, after years of development, this technology has matured, and there is limited room for improvement in compression efficiency, which is typically lower than that of the other two methods. To further enhance the efficiency of encoding compression, researchers have proposed preprocessing the original test set, leading to methods that enhance encoding compression efficiency. These methods can effectively improve encoding efficiency and are usually used in conjunction with encoding-based compression methods.

A comparative analysis of the three test stimulus compression techniques is presented from the perspectives of compression principle, technical advantages, limitations, application scenarios, and compression objects, summarized in Table 2.

Table 2. Comparative analysis of three test stimulus compression techniques.

Test Stimulus Compression	Principle	Dominance	Limitation	Application Scenarios	Compressed Objects
Encoding-Based	Using test data to determine compatibility and similarity between bits.	No constraints are imposed on ATPG, making it applicable to any test set.	The compression efficiency is relatively low when the vector contains a large number of don't-care bits.	Testing IP cores with unknown structure information.	Original dataset or dataset processed with enhanced encoding compression efficiency methods.
Scan Chain Structure Optimization-Based with Linear Decompression	Using irrelevant bits in the measured data and populating the scan chain with a linear decompression structure.	Effectively utilizing the don't-care bits in test vectors, with minimal hardware resource consumption and ease of implementation.	Subject to the constraints of ATPG, it requires solving the relevant system of equations.	Testing IP cores with known structure information.	Test data output from the test interface.
Scan Chain Structure Optimization-Based with Broadcast Scan	Utilizing the correlation between test vectors of different sub-circuits, the same data are broadcasted to multiple scan chains.	Capable of simultaneously measuring multiple sub-circuits.	Subject to the constraints of ATPG, the fault coverage is limited.	Testing IP cores with known structure information.	Test data output from the test interface.
Enhancing Encoding Compression Efficiency	Dividing the test set and projecting it into a domain space to enhance encoding efficiency.	No constraints are imposed on ATPG, making it applicable to any test set.	Requires co-utilization with encoding compression, as the compression efficiency is low when used independently.	Testing IP cores with unknown structure information.	Original dataset

3. Encoding-Based Test Stimulus Compression Methods

In integrated circuit testing, the test data for the chip are pre-stored in the ATE. To reduce the storage space and test time of the ATE, it is common to employ encoding-based test data compression techniques. Encoding compression methods have been widely used as a key generic technology in various scenarios such as image processing, video special effects, and speech recognition. This method has also been maturely applied and developed in-depth in the field of test data compression [23,24,33,34].

The working principle of encoding compression methods involves dividing the original test dataset into multiple symbols or strings, with each symbol or string being replaced by a code word. The length of the code word is typically shorter than that of the original symbol or string, resulting in a compressed data volume that is lower than the original test data volume. The compressed test data are then stored in the ATE. When the ATE outputs test data to the CUT, it requires the use of a decoder to restore the test data to its original form and input it into the scan chains within the CUT. This decoder typically requires a portion of the hardware resources on the CUT. Additionally, Chun [35] compared the hardware overhead of six encodings including SC, Golomb, VIHC, FDR, SHC, and MICRO, taking the ISCAS'89 and ITC'99 benchmark circuits as test circuits with a single scan chain. Among them, SHC exhibited the highest hardware overhead, up to 20%, while the MICRO encoding had the lowest, less than 0.1%. If the decoder implementation is defective, it will incur significant hardware overhead, thereby increasing the power consumption of the chip during testing.

According to the difference in data length before and after encoding, encoding compression methods can be subdivided into four categories: variable-length to variable-length, variable-length to fixed-length, fixed-length to variable-length, and fixed-length to fixed-length. The differences among these four methods are shown in Table 3 [24].

Table 3. Comparative analysis of four encoding compression methods.

Encoding Category	Typical Encoding Techniques	Compression Effectiveness	Hardware Overhead	Control Protocol
Fixed-Fixed	Dictionary Encoding	Poor	Small	Simple
Fixed-Variable	Huffman Coding	Medium	Medium	Medium
Variable-Variable	Golomb Coding FDR Coding	Good	Large	Complicated
Variable-Fixed	Classic Run-Length Encoding	Medium	Medium	Medium

3.1. Fixed-Length to Fixed-Length

The typical encoding method for fixed-length to fixed-length is dictionary encoding. In the test stimulus compression method based on dictionary encoding, the data in the original test set are divided into segments of n bits each. These segments are then encoded using b-bit code words, where $b < n$. The b-bit code words act as indices in a dictionary, while the n-bit segments correspond to the content of the dictionary entries, with each index corresponding to a unique entry. This indexing relationship is stored in the decoder. During testing, the ATE sends the index to the decoder, which then outputs the corresponding content to the scan chains within the CUT. As the number of test vectors increases, not every possible segment is indexed in the dictionary. If all segments in the original test set can be indexed in the dictionary, the dictionary encoding is considered complete.

Reddy's [36] method based on multi-scan chain structure for complete dictionary encoding is shown in Figure 3. The scan chains for testing consist of n chains, and every clock cycle inputs an n-bit segment into the scan chains. In dictionary encoding, each scan slice is filled according to the principle of the fewest segment types and stored in the dictionary. The index count in the dictionary is $b = \log_2 N$, where N is the number of segment types after the scan slice is filled. The more scan chains, the more bits in the scan slices and the shorter the test time. However, the complete dictionary encoding method is limited by the number of segment types after filling, and the more types there are, the more indexes there are and the higher the design complexity of the decoder, thus occupying more hardware resources.

In response to these limitations, Li [37] and Wurtenberger [38] proposed a partial dictionary encoding compression method. The segments after the scan slices are filled are sorted by their occurrence frequency, and the high-frequency segments are stored in the dictionary while the low-frequency ones are not encoded and are input directly to the scan chains as a bypass. An additional flag bit is added to distinguish whether the segment has been compressed. On this basis, to allow more segments to be encoded into the dictionary under strict control of the dictionary size, Seok-Won [39] and Kanad Basu [40] utilized masking technology to encode more segments through masking operations. The physical implementation of the masking technology is simply an XOR network, which is easy to implement and has minimal overhead. Additionally, Sismanoglou [41] proposed an index reuse method to increase the compression efficiency of dictionary encoding.

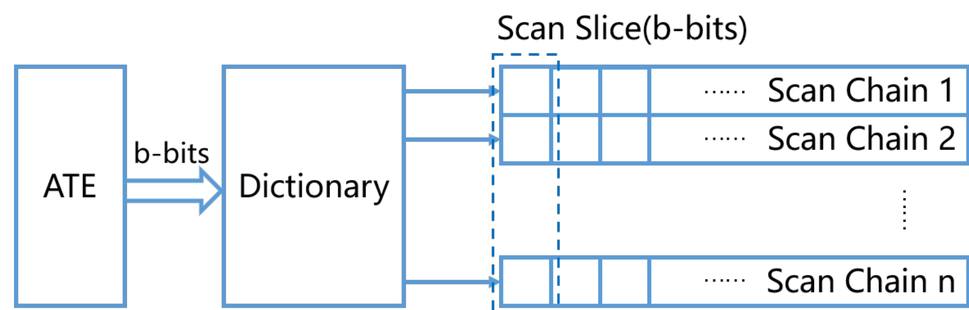

Figure 3. Dictionary encoding for testing [17].

3.2. Fixed-Length to Variable-Length

The encoding method from fixed-length to variable-length divides the original test dataset into equal-length data blocks and then selects variable-length code words to replace each data block. The length of the code words needs to be chosen based on the data characteristics of the data blocks, which directly affects the data compression efficiency of this compression method. Typical technologies for this type of encoding include Huffman encoding [42], followed by the development of selected Huffman encoding [43], optimal selected Huffman encoding [44], and other improved Huffman encoding [45].

As a data block partition-based encoding technique, Huffman encoding first divides the original test dataset into fixed equal-length data blocks, then counts the occurrences of

each data block. Finally, it encodes the data blocks with higher occurrences with shorter code words and those with fewer occurrences with longer code words. The decoder used in Huffman encoding technology is based on a finite state machine. The complexity of the decoder is directly proportional to the number of data block types. When there are more data block types in the original test set, the number of states in the state machine increases, and the decoder structure becomes more complex. To reduce the complexity of the decoder, selected Huffman encoding only encodes data blocks with higher occurrence counts and does not compress other data blocks. It adds information bits to mark whether each code word is used for encoding. This technology can effectively reduce the design complexity of the decoder and also reduce the hardware resource occupation of the compression technology. Kavousianos proposed optimal selected Huffman encoding for information bit selection optimization.

Furthermore, many researchers have further improved and optimized Huffman encoding. Kavousianos [46] combined linear feedback shift register (LFSR) with selected Huffman encoding. It first encodes the data blocks with LFSR; then, it encodes the data blocks that cannot be encoded with LFSR using selected Huffman encoding [47]. In addition, Kavousianos further divided the data blocks that cannot be selected for Huffman encoding into smaller data blocks and then performed secondary selected Huffman encoding on the smaller data blocks [48]. These improved technologies have significantly increased the compression efficiency of Huffman encoding and the design complexity of the decoder. For ATE, Usha proposed optimizations and improvements for the hardware implementation of selected Huffman encoding to reduce test power consumption and test time [49]. Recently, Tenentes improved Huffman encoding's detection capability for non-modeled faults for ATE test interfaces [50].

3.3. Variable-Length to Variable-Length

The encoding method from variable-length to variable-length replaces sequences of continuous 0 s or 1 s, known as runs, with variable-length code words. Based on this, Chandra proposed Golomb encoding [24] and Frequency-Directed Run-length (FDR) encoding [33]. Golomb encoding groups 0 runs and 1 runs for encoding, with the code words having a prefix and a suffix. The prefix indicates the group of runs, and the suffix indicates the position of the run within the group. Once the group is determined, the suffix code length is constant, while the prefix code length varies, and the number of code words in each group is equal. The length of the suffix code is related to the size of the prefix and the group.

FDR encoding also consists of a prefix and a suffix, with the prefix encoding method being the same as Golomb's. However, the length of the FDR suffix is variable and is equal to the prefix length, and the number of code words in each group is not equal, increasing exponentially. To optimize the FDR encoding process, Chandra proposed Extended Frequency-Directed Run-Length (EFDR) encoding [33], which adds an additional bit to the code word to distinguish between 0 runs and 1 runs. For EFDR encoding, Fang Hao [51] proposed the optimal filling method for the irrelevant bit X. It improves the compression efficiency of EFDR encoding without adding extra hardware resource occupation. When the number of 0 runs and 1 runs in the dataset is approximately equal, the efficiency of FDR encoding is limited. In this case, Chandra encoded 0 runs and 1 runs alternatively [52], forming the Adaptive Linear Feedback Register (ALT-FDR) encoding. This method is based on 0 and 1 dual runs, and does not require an additional identifier bit. On this basis, other researchers proposed symmetric run encoding [53], equal run encoding [54], Improved Frequency-Directed Run-Length (IFDR) encoding based on equal runs [55], Pattern Run-Length (PRL) encoding [56], and 2nPRL encoding [57]. In the past decade, An adaptive Extended Frequency-Directed Run-Length (EFDR) code method for test data compression was presented by Kuang [58]. It can achieve a compression rate of 69.87%, 4.07% higher than the original EFDR code method. Meanwhile, the proposed

Count Compatible Pattern Run-Length (CCPRL) coding compression method can achieve a compression ratio of 71.73% [59].

Additionally, some researchers proposed Variable-Length Input Huffman Coding (VIHC) encoding [60], which considers only single runs, and RL-Huffman encoding [45], which considers dual runs, based on run encoding and Huffman encoding. In these cases, the code word length depends on the frequency of run lengths occurring.

3.4. Variable-Length to Fixed-Length

This class of methods divides the original dataset with variable lengths, but the encoded code words have fixed lengths. The fixed length of the code words is relative, meaning within a certain range, the length of the code words does not change with the variation of the data blocks. Representative technologies of this class include traditional run-length encoding, but with the advancement of technology and the increasing demands of testing, this technology has been gradually replaced by variable run-length techniques. In addition, Wolff [61] and Knieser [62] used the LZ77 algorithm from the data compression field to process test stimuli, successively proposing test stimulus compression algorithms based on LZ77 and LZW, achieving variable-to-fixed-length encoding compression.

4. Scan Chain Optimization-Based Compression Methods

4.1. Linear Decompression Structure

The first type of optimized scan chain structure is the linear decompressor, which is a combinational logic circuit consisting of XOR gates and flip-flops. Its output response space is a linear space formed by the expansion of a Boolean matrix [25]. Typical linear decompression structures include Linear Feedback Shift Registers (LFSR), Cellular Automata, and Ring Generators, as shown in Figure 4. Compared to LFSRs, RGs demonstrate enhanced encodability, albeit with an associated hardware overhead that translates to a chip area increase of up to 14% [26].

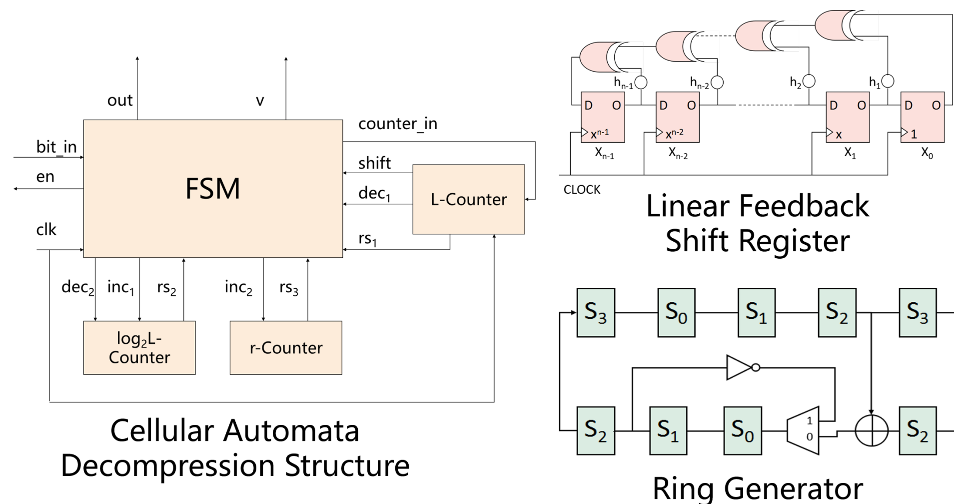

Figure 4. Schematic diagram of linear decompression structure [26].

The test stimulus compression method based on a linear decompression structure achieves the transformation of compressed test vectors X stored in the ATE into the test vectors Y inputted into the circuit under test by solving a system of linear equations, MX = Y. In this equation, M represents the characteristic matrix of the linear decompression structure, which symbolizes the structure. X is a free variable that can be reassigned and is treated as a seed, with the process of inputting the variable X into the decompression structure being referred to as reseeding. The linear decompression structure can only output the target vector if and only if the system of linear equations has a solution [27].

Based on the differences in reseeding methods, the reseeding compression technique is categorized into static reseeding and dynamic reseeding. Static reseeding involves re-inputting a new seed after testing a scan chain is completed. Taking LFSR as an example, the seed is transmitted from the ATE to the LFSR, where it decompresses the test vectors and inputs them into the scan chain. Once the scan chain outputs the response, the LFSR retrieves a new seed from the ATE for the next round of decompression. In static reseeding, the structure of the LFSR is constrained by the test data, meaning that the length of the LFSR must be no shorter than the maximum number of fixed bits in the test cube.

To reduce the length of the LFSR [63,64], Krishna and Wohl encoded the relationship between the seed and the scan slice, allowing each test vector to be determined by multiple seeds. During static reseeding, the ATE is idle during the linear decompression process, with the seed reading process being carried out serially with the generation of test vectors. This serial operation mode results in long test time and low test efficiency.

To address this issue, Rajski [65] and others proposed dynamic reseeding, which allows the ATE to read the seed and generate the test vectors in parallel, thereby increasing the utilization of the ATE and reducing test time. Cheng [66] utilized a ring generator as the decompression structure and optimized the dynamic reseeding technique based on the Embedded Deterministic Test (EDT). This method can generate test cubes of a fixed length, improving encoding flexibility and subsequently reducing test time. Koenemann [67] utilized the redundant test channels of the ATE to control the seed timing and employed variable-length seed coding for test cubes. Zhang [68] used multiple polynomials for reseeding, reducing the dependency probability while also reducing the length of the seed. Kim [69] further reduced the seed length for test sets with fewer determinants to improve test efficiency. Additionally, Krishna proposed partial reseeding [70], Li reseeded the overlapping parts of the scan slices [71], and Kongtim proposed parallel LFSR reseeding [72]. Recently, Yoneda optimized LFSR for delay faults [73] and Wang proposed a mixed-mode LFSR reseeding [74], both of which can effectively increase test efficiency. Acevedo [75] also proposed an algorithm for the shortest order of the LFSR, further reducing the order being limited by the test cube.

4.2. Broadcast Scan Structure

The test stimulus compression method based on broadcast scan is another type of scan chain optimization, allowing the test stimuli to be shared simultaneously with multiple scan chains in the circuit under test. Compared to the linear decompression structure, a broadcast scan structure can effectively reduce the number of test vectors in certain cases but exhibit inferior compression capabilities for the overall data volume [28]. Lee was the first to propose this broadcast-sharing scan chain optimization method [29]. This optimization method is simple in structure and efficient in compression, but when applying this method to test the circuit under test, the fault coverage is limited and cannot cover some rare faults. To improve the fault coverage of broadcast scan testing, researchers from the University of Illinois in the United States proposed the famous Illinois scan architecture [30], which includes a broadcast scan mode and a serial scan mode. Faults that cannot be detected in the broadcast scan mode need to be detected in the serial scan mode to increase the fault coverage of the testing method.

The broadcast scan mode of the Illinois scan architecture is depicted in Figure 5. The original single scan chain can be divided into four scan chains of length L, and the test vectors are input to these four scan chains in a shared broadcast manner. In this mode, a single test channel can simultaneously drive four scan chains, and the responses from the scan chains are also transmitted to the Multiple Input Signature Register (MISR). Although the detection efficiency is high in the broadcast scan mode, there are some special non-common fault structures that cannot be detected by this mode. In such cases, it is necessary to switch to the serial scan mode and input additional test vectors to detect these non-common faults.

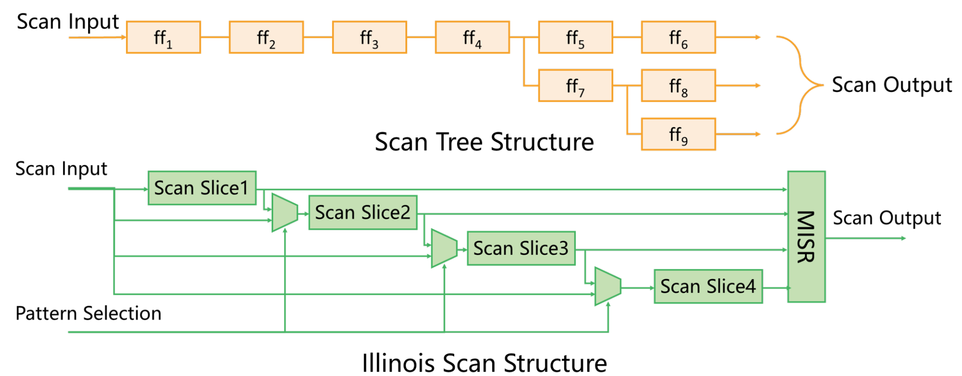

Figure 5. Schematic diagram of broadcast scanning structure [30,76].

The serial scan mode of the Illinois scan architecture involves connecting all the scan chains in series to form a single 4L serial scan chain. The test time in this mode is four times that of the broadcast scan mode. As the circuit scale of integrated circuits continues to expand, to ensure effective testing, the length of a single scan chain also increases. In this situation, the test time cost of the serial scan mode increases significantly, severely affecting the testing efficiency of the Illinois scan architecture. Therefore, it is necessary to switch between test modes in real-time through selectors between scan slices according to the testing requirements.

The Illinois architecture relies on a fixed connection relationship between the scan chains and the scan inputs, and under this structural limitation, the test time cannot be further compressed. Therefore, researchers have improved the Illinois structure to form a reconfigurable Illinois scan structure, which allows the connection relationship between the scan chains and the original scan inputs to be reconfigured. In this structure, the fault coverage can be improved by reconstructing the connection relationship, reducing the need for mode switching and subsequently shortening the test time.

According to the different stages of reconfiguring the connection relationship, the reconfigurable Illinois scan structure can be further categorized into static reconfiguration and dynamic reconfiguration. Static reconfiguration refers to the reconfiguration of the connection relationship during the switching process of test vectors, i.e., before the next test vector is inputted, the connection structure between the input terminal and the scan chains is changed through a multiplexer. In dynamic reconfiguration, the reconfiguration of the connection relationship occurs during the input process of the test vector into the scan chain. Compared to static reconfiguration, dynamic reconfiguration offers higher flexibility and is suitable for circuits with complex functions and structures, enabling the detection of more faults. However, its control circuitry is more complex to achieve accurate structural changes.

Donglikar utilized heuristic algorithms to reorder scan cells to enhance the fault coverage of the broadcast scan mode [77]. Pandey introduced an additional broadcast scan method and proposed a reconfiguration method to improve the compression efficiency of testing [78]. To simultaneously enhance the compression efficiency and fault coverage of testing, Gupta proposed a new layout and routing method for the broadcast scan mode [79]. Subsequently, Wang made improvements for channel delay faults under the broadcast mode [80]. In light of the limitations of the physical location and layout of scan cells, Banerjee optimized the wiring of the cells [81]. Considering that the physical location of scan cells is fixed before scan vector input, the above reconfiguration and optimization methods can effectively improve compression efficiency but introduce the consumption of wiring resources. Therefore, some research scholars have made other design improvements around the Illinois scan structure, including multi-fan-in optimization [82], pin optimization [83], RTL-level optimization [84], and low-power optimization [85].

The scan tree structure is another typical structure used in broadcast scan, which is also effective for compressing test stimuli and reducing test time. Compared to Illinois scan

structure, scan tree can effectively reduce test application cost but not test power [86]. In the scan tree structure, scan cells are connected in a tree-like structure, with compatible cells connected in parallel and incompatible cells connected in series. During testing, test data are input into the root node of the scan tree, and then propagated to every node, or scan cell, of the scan tree. The test data of the scan cells at the same level are the same. Compared to multiple scan chains, the probability of broadcasting test vectors in the scan tree is higher. Compared to a single scan chain, the longest scan chain in the scan tree has a shorter length, resulting in shorter test time. Miyase and others preprocessed the original test set according to the characteristics of the scan tree to achieve better compression effects [76]. Banerjee [87] and others proposed a coarse compatibility method based on the compatibility between scan cells, and You [88] proposed an extended compatibility tree method, both of which can effectively reduce test time. Other researchers have optimized the scan tree structure to improve compression effects, including a hybrid structure of reconfigurable scan chains and scan trees [89], scan forest structures [90], and low-power scan forest structures [91].

5. Enhancing Encoding Compression Efficiency

5.1. Traditional Preprocessing Methods

Due to the limited scope for improving the efficiency of encoding compression, extensive research has been conducted on test data preprocessing methods for enhancing encoding compression efficiency. Traditional preprocessing methods involve studying the inter-row similarity, inter-column similarity, and two-dimensional similarity of test data in the test set. For inter-row similarity, researchers employ techniques such as test vector segmentation [92], differencing [93], and reordering [94] to compactly arrange similar test vectors, which aids in extending the length of symbols between consecutive test vectors to achieve the purpose of enhancing encoding efficiency. However, this method overlooks the length of symbols that are not encoded in the test vectors. For inter-column similarity, researchers partition and reorder scan cells [95], placing similar columns next to each other in the test set to extend the length of encoded symbols within the test vectors but neglecting the length of symbols between consecutive test vectors. For two-dimensional similarity, which refers to both inter-row and inter-column similarity, researchers propose a two-dimensional reordering of the test machine, namely, rearranging all rows first based on inter-row similarity and then rearranging all columns based on inter-column similarity. This method optimizes the encoding length from the overall perspective of the test data, significantly improving the effectiveness of double run-length encoding. However, these preprocessing methods also have significant drawbacks, such as introducing signal delays when the order of scan cells is changed and reconnected, which can affect test quality.

The traditional preprocessing methods described above do not consider the limitation of test power consumption. However, during testing, the significant switching activity of integrated circuits can lead to a surge in power consumption, which not only affects operational reliability but also performance. When the test power consumption exceeds the tolerance range of the integrated circuit, it can cause irreversible damage to the circuit. Therefore, researchers have proposed techniques such as scan chain reordering [96], test compression [97], and ATPG [98] to address the power consumption issues in the reordering process. These preprocessing methods for test data can effectively reduce test power consumption and optimize encoding compression [10], but the issue of signal delays introduced by wiring modifications between scan cells still persists. To address this problem, Yuan [99] proposed a two-dimensional reordering of tests based on Hamming distance and completed a two-dimensional re-adjustment based on irrelevant bits. This approach can effectively reduce the number of logic transitions for test data with a high proportion of irrelevant bits, but the adjustment effect is not ideal for test data with a large number of deterministic bits.

5.2. Preprocessing Methods Based on Spectral Analysis

In recent years, to further enhance the efficiency of encoding compression, researchers have conducted preprocessing of test data based on spectral analysis. In test compression, spectral analysis is used to extract key test vector information components that determine fault coverage from the test vectors. Researchers have found that high-quality test vector sets exhibit characteristics related to the spectrum. By performing preprocessing based on spectral analysis, it is possible to generate test sets with high fault coverage while effectively reducing the amount of test data, achieving test compression. In practical applications, Yogi [100] proposed the use of Walsh–Hadamard transform for digital circuit analysis, which can detect fixed faults in circuits. Giani applied this transformation to the Built-In Self-Test (BIST) and ATPG for timing circuits [31]. Yogi applied spectral analysis to test vector preprocessing at the RTL (Register Transfer Level) [32]. Subsequently, Yogi completed test compression for fixed faults [101] and transition faults [102] in microprocessor testing. In 2010, Yogi extended the applicability of this method to very large-scale digital circuits, demonstrating that the Walsh–Hadamard transform can effectively analyze the principal components and residual components in the test set, and the inverse transform can recover the principal components into the original test vectors [100] (Figure 6).

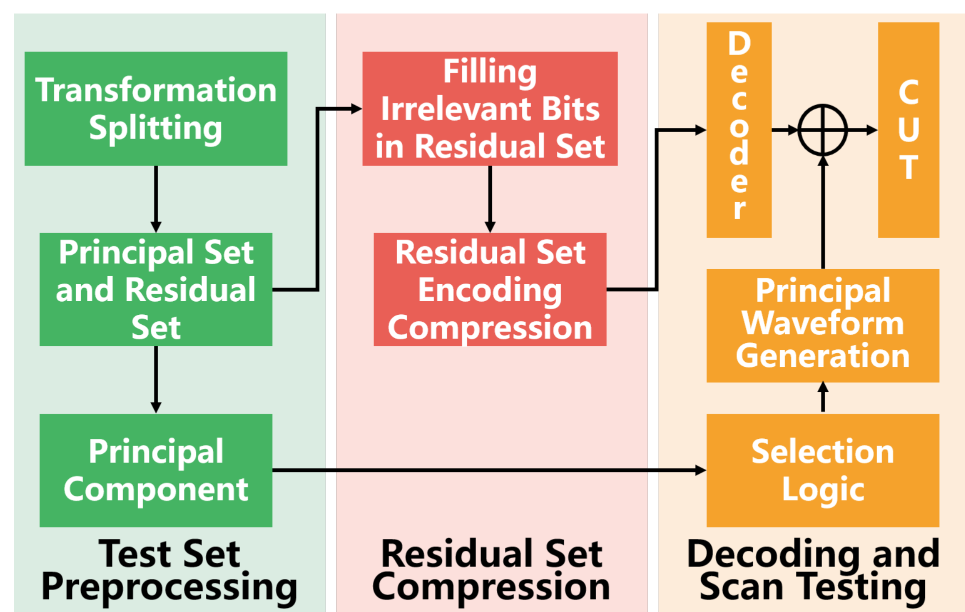

Figure 6. Test process flow based on spectral analysis preprocessing [100].

The decompressor based on Walsh–Hadamard transform consists of counters and the XOR network, where the bit length of the counters is determined by the order of the Hadamard matrix. Logic gates are used to selectively control the outputs of the counters; then, the Walsh function, or the principal component, is generated through the XOR network. The compressed data stored on the ATE are output through the decoder as the residual component. The principal component and the residual component are restored to the original test vectors through XOR gates and then input into the scan circuit.

6. Future Directions and Prospects

1. In methods aimed at enhancing coding compression efficiency, the higher the similarity between the principal component set and the original test set, the better the encoding compression effect of the residual set data. Considering that principal components are outputs after transformation operations, future efforts can explore various transformation operations to enhance compression effectiveness. Research can focus on matrix transformations to investigate how to increase the similarity between vectors in the transformation matrix and the bitstream. Simultaneously, it is also feasible

to analyze the impact of bit transformations in test sets on the compression effect of the transformation matrix without compromising fault coverage.

2. While coding-based compression methods offer numerous advantages, the development of coding techniques themselves is relatively mature, leaving limited room for further research. Future work should consider integrating these methods with other emerging technologies to achieve higher compression efficiency. For instance, combining machine learning with coding techniques could involve using machine learning to analyze the impact of different preprocessing methods in conjunction with various coding techniques on compression efficiency. By characterizing the data in the original dataset, learning models can be trained. When compressing new test vector sets, these models can suggest appropriate methods and parameters.
3. Test data compression and test power optimization are two prominent research topics in test optimization. Currently, these topics are often studied in isolation without collaborative optimization research. In coding-based test compression methods, filling irrelevant bits "X" in the original test data can achieve higher compression efficiency. However, this also introduces additional test power consumption, which can affect test outcomes. Therefore, future research should consider both test compression and test power consumption simultaneously to balance the trade-offs between the two, achieve optimal test results, and reduce test costs.

7. Conclusions

Integrated circuit testing is a critical process for ensuring the quality and reliability of integrated circuit products. In the post-Moore's Law era, with the continuous evolution of integrated circuit fabrication technologies and the development of SoC and chiplet technologies, the number of IP cores and chiplets integrated on a single chip is increasing, leading to more complex functionalities. This has resulted in a significant surge in the volume of test data, while the performance improvement in test equipment lags behind the advancements in design technology, creating a bottleneck in the testing of very large-scale integrated circuits. To address this bottleneck, this paper summarizes and analyzes the existing integrated circuit test compression techniques from three aspects: test compression-oriented coding methods, scan chain structure optimization, and enhancement of coding compression efficiency. In view of the limitations of current research, the paper also prospects the future development trends of integrated circuit testing, with the aim of providing guidance and recommendations for the advancement of test compression technology.

Author Contributions: Conceptualization, D.Y.; methodology, S.Z.; investigation, L.C.; writing—original draft preparation, L.Z.; writing—review and editing, W.Z.; visualization, Y.X. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not Applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Zhang, M.; Kuang, J.; Huang, J. Improving Compression Ratios for Code-Based Test Pattern Compressions through Column-Wise Reordering Algorithms. *J. Circuits Syst. Comput.* **2020**, *43*, 39–46. [CrossRef]
2. 42 Years of Microprocessor Trend Data . Available online: https://www.karlrupp.net/2018/02/42-years-of-microprocessor-trend-data (accessed on 15 February 2018).
3. *IEEE P1500*; Standard Testability Method for Embedded Core-Based Integrated Circuits. Institute of Electrical and Electronics Engineers: Piscataway, NJ, USA, 2022
4. *IEEE P1450*; Standard Test Interface Language. Institute of Electrical and Electronics Engineers: Piscataway, NJ, USA, 2018

5. Ansari, M.A. Network Load Analysis During Test Mode for the Network-on-Chip Reused Test Access Mechanisms. In Proceedings of the 2018 International Conference on Computing, Electronics and Communications Engineering (iCCECE), Southend, UK, 16–17 August 2018; pp. 323–327.
6. Allan, A. *International Technology Roadmap for Semiconductors (ITRS)*; Springer: Berlin/Heidelberg, Germany, 2015._7. [CrossRef]
7. Werkmann, H.; Kim, D.-M.; Fujita, S. GDDR5 Training-Challenges and Solutions for ATE-Based Test. In Proceedings of the 2008 17th Asian Test Symposium, Washington, DC, USA, 24–27 November 2008; pp. 423–428.
8. Advantest Rolls Out Pin Scale Multilevel Serial-Next-Generation High-Speed ATE Instrument. Available online: https://www.advantest.com/en/news/2023/20231114.html (accessed on 14 November 2023).
9. Challenges and Outlook of ATE Testing For 2nm SoCs. Available online: https://semiengineering.com/challenges-and-outlook-of-ate-testing-for-2nm-socs (accessed on 8 August 2024).
10. Kumar, G.S.; Paramasivam, K. Test power minimization of VLSI circuits: A survey. In Proceedings of the 2013 Fourth International Conference on Computing, Communications and Networking Technologies (ICCCNT), Tiruchengode, India, 4–6 July 2013; pp. 1–6.
11. Iwata, H.; Maeda, Y.; Matsushima, J. A Power Reduction Method for Scan Testing in Ultra-Low Power Designs. In Proceedings of the 2021 IEEE 30th Asian Test Symposium (ATS), Virtual, 22–24 November 2021; p. 141.
12. Voyiatzis, I. Embedding test patterns into Low-Power BIST sequences. In Proceedings of the 13th IEEE International On-Line Testing Symposium (IOLTS 2007), Washington, DC, USA, 8–11 July 2007; pp. 197–198.
13. Barla, P.; Joshi, V.K.; Bhat, S. Spintronic devices: A promising alternative to CMOS devices. *J. Comput. Electron.* **2021**, *20*, 805–837. [CrossRef]
14. Alasad, O.; Yuan, J.; Shuinsubramanyan, P. Strong Logic Obfuscation with Low Overhead against IC Reverse Engineering Attacks. *ACM Trans. Des. Autom. Electron. Syst.* **2020**, *25*, 1–31. [CrossRef]
15. Lee, K.; Chen, B.; Kochte, M.A. On-Chip Self-Test Methodology With All Deterministic Compressed Test Patterns Recorded in Scan Chains. *IEEE Trans. Comput. Des. Integr. Circuits Syst.* **2019**, *38*, 309–321. [CrossRef]
16. Rooban, S.; Manimegalai, R. Prediction of Theoretical Limit for Test Data Compression. In Proceedings of the 2018 International Conference on Recent Trends in Advance Computing (ICRTAC), Chennai, India, 10–11 September 2018; pp. 41–46.
17. Rooban, S.; Manimegalai, R. A Dictionary-Based Test Data Compression Method Using Tri-State Coding. In Proceedings of the 2018 IEEE 27th Asian Test Symposium, Hefei, China, 15–18 October 2018.
18. Touba, N.A. Survey of test vector compression techniques. *IEEE Des. Test Comput.* **2006**, *23*, 294–303. [CrossRef]
19. Li, X.; Lee, K.J.; Touba, N.A. Test Compression *VLSI Test Principles and Architectures;* Morgan Kaufmann: Burlington, MA, USA, 2006; pp. 341–396
20. Mehta, U.S.; Dasgupta, K.S.; Devashrayee, N.M. Survey of test data compression technique emphasizing code based schemes. In Proceedings of the 2009 12th Euromicro Conference on Digital System Design, Architectures, Methods and Tools, Patras, Greece, 27–29 August 2009.
21. Mehta, U, Survey of VLSI Test Data Compression Methods. *Nirma Univ. J. Eng. Technol.* **2010**, *1*, 38–41.
22. Sudandhira Veeran, R.; Rajkumar, M. Test Data Compression Methods: A Review. In Proceedings of the International Conference on Artificial Intelligence, Smart Grid and Smart City Applications, Coimbatore, India, 3–5 January 2019; Springer: Berlin/Heidelberg, Germany, 2019, pp. 791–799.
23. Hu, M.; Hu, J.; Wang, H. Research on Test Data Compression Method Based on Transformation. In Proceedings of the 2020 5th International Conference on Mechanical, Control and Computer Engineering, Harbin, China, 25–27 December 2020.
24. Mabel, D.J.J.; Mary, M.C.V.S. A Proficient Test Data Compression and Decompression System for Enhanced Test Competence in SOC Testing. In Proceedings of the 2023 International Conference on Inventive Computation Technologies, Lalitpur, Nepal, 26–28 April 2023.
25. Lee, D.; Roy, K. Viterbi-Based Efficient Test Data Compression. *IEEE Trans. Comput. Aided Des. Integr. Circuits Syst.* **2012**, *31*, 610–619. [CrossRef]
26. Novák, O.; Jenícek, J.; Rozkovec, M. Test Decompressor Effectivity Improvement. In Proceedings of the 2016 Euromicro Conference on Digital System Design (DSD), Limassol, Cyprus, 31 August–2 September 2016; pp. 661–664.
27. Bushnell, M.; Agrawal, V.D. Essentials of electronic testing for digital, memory, and mixed-signal VLSI circuits. In *Electronic Services*; Springer: Berlin/Heidelberg, Germany, 2000; pp. 498–511.
28. Balakrishnan, K.J.; Touba, N.A. Improving encoding efficiency for linear decompressors using scan inversion. In Proceedings of the 2004 International Conferce on Test, Charlotte, NC, USA, 26–28 October 2004; pp. 936–944.
29. Novák, O. Deterministic Search Strategy of Compression Codes. In Proceedings of the IEEE/ACM International Conference on Computer-aided Design, Golem, Albania, 6–8 September 2023; pp. 198–205.
30. Hsu, F.F.; Butler, K.M.; Patel, J.H. A case study on the implementation of the Illinois scan architecture. In Proceedings of the IEEE International Test Conference, Baltimore, MD, USA, 10 October 2002; pp. 538–547.
31. Giani, A.; Sheng, S.; Hsiao, M.S. Novel spectral methods for built-in self-test in a system-on-a-chip environment. In Proceedings of the 19th IEEE VLSI Test Symposium (VTS), Marina Del Rey, CA, USA, 29 April–3 May 2001; pp. 163–168.
32. Yogi, N.; Agrawal, V.D. Spectral RTL Test Generation for Gate-Level Stuck-at Faults. In Proceedings of the 15th Asian Test Symposium, Fukuoka, Japan, 20–23 November 2006; pp. 83–88.

33. Jas, A.; Pouya, B.; Touba, N.A. Test data compression technique for embedded cores using virtual scan chains. *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.* **2004**, *12*, 775 – 781. [CrossRef]
34. Kuang, J.; Zhang, L.; You, Z. Improve the compression ratios for code-based test vector compressions by decomposing. In Proceedings of the 20th IEEE European Test Symposium, Cluj-Napoca, Romania, 25–29 May 2015; pp. 1–6.
35. Chun, S.; Kim, Y.; Im, J.B.; Kang, S. MICRO: A new hybrid test data compression/decompression scheme. *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.* **2006**, *14*, 649–654 [CrossRef]
36. Reddy, S.M.; Miyase, K.; Kajihara, S. On Test Data Volume Reduction forMultiple Scan Chain Designs. *IEEE VLSI Test Symp.* **2002**, *8*, 103–108.
37. Li, L.; Chakrabarty, K.; Touba, N.A. Test Data Compression Using Dictionaries with Selective Entries and Fixed-Length Indices. *Proc VLSI Test Symp.* **2003**, *8*, 219–224. [CrossRef]
38. Wurtenberger, A.; Tautermann, C.S.; Hellebr, S. Data compression for multiple scan chains using dictionaries with corrections. In Proceedings of the IEEE International Test Conference, Charlotte, NC, USA, 26–28 October 2004; pp. 926–935.
39. Seong, S.; Mishra, P. Bitmask-based Code Compression for Embedded Systems. *IEEE Trans. Comput. Aided Des. Integr. Circuits Syst.* **2008**,*27*, 673–685. [CrossRef]
40. Basu, K.; Mishra, P. Test data compression using efficient bit mask and dictionary election methods. *IEEE Trans. Very Large Scale Integr. Syst.* **2010**, *18*, 1277–1286. [CrossRef]
41. Sismanoglou, P.; Nikolos, D. Input Test Data Compression Based on the Reuse of Parts of Dictionary Entries: Static and Dynamic Approaches. *IEEE Trans. Comput. Aided Des. Integr. Circuits Syst.* **2013**, *18*, 1762–1775. [CrossRef]
42. Wang, N.; Yan, W.; Jiang, H.; Lin, S.J.; Han, Y.S. Prefix Coding Scheme Supporting Direct Access Without Auxiliary Space. *IEEE Trans. Knowl. Data Eng.* **2023**, *35*, 12917–12931. [CrossRef]
43. Jas, A.; Ghosh-Dastidar, J.; Mom-Eng, N. An efficient test vector compression scheme using selective Huffman coding. *IEEE Trans. -Comput.-Aided Des. Integr. Circuits Syst.* **2003**, *22*, 797–806. [CrossRef]
44. Kavousianos, X.; Kalligeros, E.; Nikolos, D. Optimal Selective Huffman Coding for Test-data Compression. *IEEE Trans. Comput.* **2007**,*56*, 1146–1152. [CrossRef]
45. Nourani, M.; Tehranipour, M. RL-Huffman Encoding for Test Compression and Power Reduction in Scan Applications. *IEEE Int. Symp. Circuits Syst.* **2005**, *10*, 11681–11684. [CrossRef]
46. Kavousianos, X.; Kalligeros, E.; Nikolos, D. Multilevel Huffman Coding: An Efficient Test-Data Compression Method for IP Cores. *IEEE Trans. Comput. Aided Des. Integr. Circuits Syst.* **2007**, *26*, 1070–1083. [CrossRef]
47. Kavousianos, X.; Kalligeros, E.; Nikolos, D. Multilevel-Huffman Test-Data Compression for IP Cores With Multiple Scan Chains. *IEEE Trans. Very Large Scale Integr. (Vlsi) Syst.* **2008**, *16*, 926–931. [CrossRef]
48. Kavousianos, X.; Kalligeros, E.; Nikolos, D. Test Data Compression Based on Variable-to-Variable Huffman Encoding With Code word Reusability. *IEEE Trans. Comput. Aided Des. Integr. Circuits Syst.* **2008**, *27*, 1333–1338. [CrossRef]
49. Mehta, U.S.; Dasgupta, K.S.; Devashrayee, N.M. Modified Selective Huffman Coding for Optimization of Test Data Compression, Test Application Time and Area Overhead. *J. Electron. Testing: Theory Appl.* **2009**, *26*, 679–688. [CrossRef]
50. Tenentes, V.; Kavousianos, X. High-Quality Statistical Test Compression With Narrow ATE Interface. *IEEE Trans. -Comput.-Aided Des. Integr. Circuits Syst.* **2013**, *32*, 1369–1382. [CrossRef]
51. Fang, H.; Yao, B.; Song, X.-D. The Algorithm of Filling X Bits in Dual-Run-Length Coding. *Acta Electron. Sin.* **2009**, *37*, 1–6.
52. Chandra, A.; Chakrabarty, K. A unified approach to reduce SOC test data volume, scan power and testing time. *IEEE Trans. -Comput.-Aided Des. Integr. Circuits Syst.* **2003**, *22*, 352–363. [CrossRef]
53. Liu, J.; Xu, S. Integrative test compression scheme based on symmetrical coding. *Chin. J. Entific Instrum.* **2012**, *33*, 2130–2136.
54. Zhan, W.; El-Maleh, A. A new collaborative scheme of test vector compression based on equal-run-length coding. In Proceedings of the 2009 13th International Conference on Computer Supported Cooperative Work in Design, Santiago, Chile, 22–24 April 2009; Volume 5, pp. 91–98.
55. You, Z.Q.; Luo, Q.J. A test compression method based on IFDR code. *J. Hunan Univ. (Nat. Sci.)* **2016**, *43*, 130–134.
56. Ruan, X.; Katti, R.S. Data-independent pattern run-length compression for testing embedded cores in SoCs. *IEEE Trans. Comput.* **2007**, *56*, 545–556. [CrossRef]
57. Lee, L.J.; Tseng, W.D.; Lin, R.B. Pattern run-length for test data compression. *IEEE Trans. Comput. Aided Des. Integr. Circuits Syst.* **2012**, *31*, 644–648. [CrossRef]
58. Kuang, J.; Zhou, Y.; Cai, S. Adaptive EFDR Coding Method for Test Data Compression. *J. Electron. Inf. Technol.* **2015**, *37*, 2529–2535.
59. Yuan, H.; Mei, J.; Song, H. Test Data Compression for System-on-a-Chip using Count Compatible Pattern Run-Length Coding. *J. Electron. Test.* **2014**, *30*, 237–242. [CrossRef]
60. Gonciari, P.T.; Al-Hashimi, B.M.; Nicolici, N. Variable-length input Huffman coding for system-on-a-chip test. *IEEE Trans. Comput. Aided Des. Integr. Circuits Syst.* **2003**, *22*, 783–796. [CrossRef]
61. Wolff, F.G.; Papachristou, C. Multiscan-based test compression and hardware decompression using LZ77. In Proceedings of the International Test Conference, Baltimore, MD, USA, 10 October 2002; pp. 331–339.
62. Knieser, M.J.; Wolff, F.G.; Papachristou, C.A. A technique for high ratio LZW compression logic test vector compression. *Des. Autom. Test Eur. Conf. Expo. (Date)* **2003**, *31*, 116–121.

63. Wohl, P.; Waicukauski, J.A.; Patel, S. Efficient compression of deterministic patterns into multiple PRPG seeds. In Proceedings of the IEEE International Conference on Test, Austin, TX, USA, 8 November 2005; pp. 910–925.
64. Krishna, C.V.; Touba, N.A. Reducing test data volume using LFSR reseeding with seed compression. In Proceedings of the International Test Conference, Baltimore, MD, USA, 10 October 2002; pp. 321–330.
65. Rajski, J.; Tyszer, J.; Kassab, M. Embedded deterministic test. *IEEE Trans. Comput. Aided Des. Integr. Circuits Syst.* **2004**, *23*, 776–792. [CrossRef]
66. Cheng, W.T.; Mrugalski, G.; Rajski, J.; Trawka, M.; Tyszer, J. On Cyclic Scan Integrity Tests for EDT-based Compression. In Proceedings of the 2019 IEEE 37th VLSI Test Symposium (VTS), Monterey, CA, USA, 23–25 April 2019; pp. 1–6.
67. Koenemann, B.; Barnhart, C.; Keller, B. A Smart BIST variant with guaranteed encoding. In Proceedings of the 10th Asian Test Symposium, Kyoto, Japan, 19–21 November 2001; pp. 325–330.
68. Zhane, S.; Seth, S.C.; Bhattacharya, B.B. On finding consecutive test vectors in a random sequence for energy-aware BIST design. In Proceedings of the 18th International Conference on VLSI Design held jointly with 4th International Conference on Embedded Systems Design, Kolkata, India, 3–7 January 2005; pp. 491–496.
69. Kim, H.S.; Kim, Y.; Kang, S. Test-decompression mechanism using a variable-length multiple-polynomial LFSR. *IEEE Trans. Very Large Scale Integr. (Vlsi) Syst.* **2003**, *11*, 687–690.
70. Krishna, C.V.; Jas, A.; Touba, N.A. Test vector encoding using partial LFSR reseeding. In Proceedings of the IEEE International Test Conference, Baltimore, MD, USA, 1 November 2001; pp. 885–893.
71. Li, J.; Han, Y.; Li, X. Deterministic and low power BIST based on scan slice overlapping. In Proceedings of the 2005 IEEE International Symposium on Circuits and Systems (ISCAS), Kobe, Japan, 23–26 May 2005; pp. 5670–5673.
72. Kongtim, P.; Reungpeerakul, T. Parallel LFSR Reseeding with Selection Register for Mixed-Mode BIST. In Proceedings of the 19th IEEE Asian Test Symposium, Shanghai, China, 1–4 December 2010; pp. 153–158.
73. Yoneda, T.; Inoue, M.; Taketani, A. Seed Ordering and Selection for High Quality Delay Test. In Proceedings of the 19th IEEE Asian Test Symposium, Shanghai, China, 1–4 December 2010; pp. 313–318.
74. Wang, S.; Wei, W.; Wang, Z. A Low Overhead High Test Compression Technique Using Pattern Clustering With n-Detection Test Support. *IEEE Trans. Very Large Scale Integr. (Vlsi) Syst.* **2010**, *18*, 1672–1685. [CrossRef]
75. Acevedo, O.; Kagaris, D. On The Computation of LFSR Characteristic Polynomials for Built-In Deterministic Test Pattern Generation. *IEEE Trans. Comput.* **2016**,*65*, 664-669. [CrossRef]
76. Miyase, K.; Kajihara, S. Optimal scan tree construction with test vector modification for test compression. In Proceedings of the 2003 Test Symposium, Xi'an, China, 16–19 November 2003; pp. 136–141.
77. Donglikar, S.; Banga, M.; Chandrasekar, M. Fast circuit topology based method to configure the scan chains in Illinois Scan architecture. In Proceedings of the International Test Conference, Austin, TX, USA, 1–6 November 2009; Volume 23, pp.1–10.
78. Pandey, A.R.; Patel, J.H. Reconfiguration technique for reducing test time and test data volume in Illinois Scan Architecture based designs. In Proceedings of the 20th IEEE VLSI Test Symposium (VTS 2002), Monterey, CA, USA, 28 April 2002–2 May 2002; pp. 9–15.
79. Gupta, P.; Kahng, A.B.; Mantik, S. Routing-aware scan chain ordering. In Proceedings of the ASP-DAC Asia and South Pacific Design Automation Conference, Kitakyushu, Japan, 21–24 January 2003; Volume 10, pp. 857–862.
80. Wang, S.; Peng, K.; Li, K.S. Layout-Aware Scan Chain Reorder for Skewed-Load Transition Test Coverage. In Proceedings of the 15th Asian Test Symposium, Fukuoka, Japan, 20–23 November 2006; pp. 169–174.
81. Banerjee, S.; Mathew, J.; Pradhan, D.K. Layout-aware Illinois Scan design for high fault coverage coverage. In Proceedings of the 2010 11th International Symposium on Quality Electronic Design (ISQED), San Jose, CA, USA, 22–24 March 2010; pp. 683–688.
82. Shah, M.A.; Patel, J.H. Enhancement of the Illinois scan architecture for use with multiple scan inputs. In Proceedings of the IEEE Computer Society Annual Symposium on VLSI, Lafayette, LA, USA, 19–20 February 2004; pp. 167–172.
83. Chandra, A.; Yan, H.; Kapur, R. Multimode Illinois Scan Architecture for Test Application Time and Test Data Volume Reduction. In Proceedings of the 25th IEEE VLSI Test Symposium (VTS'07), Berkeley, CA, USA, 6–10 May 2007; pp. 84–92.
84. Ko, H.F.; Nicolici, N. Functional Illinois scan design at RTL. In Proceedings of the IEEE International Conference on Computer Design: VLSI in Computers and Processors, San Jose, CA, USA, 11–13 October 2004; pp. 8–81.
85. AlQuraishi, E.; AlTeenan, R. Average power reduction in compression-based scan designs. In Proceedings of the 15th IEEE Mediterranean Electrotechnical Conference (MELECON), Valletta, Malta, 26–28 April 2010; pp. 504–509.
86. Xiang, D.; Gu, S.; Sun, J.; Wu, Y. A cost-effective scan architecture for scan testing with nonscan test power and test application cost In Proceedings of the 2003. Design Automation Conference, Anaheim, CA, USA, 2–6 June 2003; pp. 744–747
87. Banerjee, S.; Chowdhury, D.R.; Bhattacharya, B.B. An Efficient Scan Tree Design for Compact Test Pattern Set. *IEEE Trans. Comput. Aided Des. Integr. Circuits Syst.* **2007**, *26*, 1331–1339. [CrossRef]
88. You, Z.; Huang, J.; Inoue, M. A response compactor for extended compatibility scan tree construction. In Proceedings of the IEEE International Conference on ASIC, Changsha, China, 20–23 October 2009; pp. 609–612.
89. Bonhomme, Y.; Yoneda, T.; Fujiwara, H. An Efficient Scan Tree Design for Test Time Reduction. In Proceedings of the Ninth IEEE European Test Symposium (ETS), Corsica, France, 23–25 May 2004; pp. 174–179.
90. Li, K.S. Multiple Scan Trees Synthesis for Test Time/Data and Routing Length Reduction Under Output Constraint. *IEEE Trans. -Comput.-Aided Des. Integr. Circuits Syst.* **2010**, *29*, 618–626. [CrossRef]

91. Chen, L.; Cui, A. A power-efficient scan tree design by exploring the Q'-D connection. In Proceedings of the 2013 IEEE International Symposium on Circuits and Systems (ISCAS), Beijing, China, 19-23 May 2013; pp. 1018–1021.
92. Vohra, H.; Singh, A. Test data compression using hierarchical block merging technique. *IET Comput. Digit. Tech.* **2018**, *12*, 176–185. [CrossRef]
93. Mehla, U.S.; Dasgupta, K.S.; Devashrayee, N.M. Hamming Distance Based Reordering and Columnwise Bit Stuffing with Difference Vector: A Better Scheme for Test Data Compression with Run Length Based Codes. In Proceedings of the 23rd International Conference on VLSI Design, Bangalore, India, 3–7 January 2010; pp. 33–38.
94. Zhou, Y.; Kuang, J. A sort method to enhance significant spectral components of test set. In Proceedings of the 2016 12th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD), Changsha, China, 13–15 August 2016; pp. 2147–2151.
95. Zhang, L.; Kang, J.; You, Z. Virtual scan chains reordering using a RAM-based module for high test compression. *Microelectron. J.* **2012**, *43*, 869–872.
96. Seo, S.; Lee, Y.; Lim, H. Scan Chain Reordering-Aware X-Filling and Stitching for Scan Shift Power Reduction. In Proceedings of the IEEE 24th Asian Test Symposium (ATS), Mumbai, India, 22–25 November 2015; pp. 1–6.
97. Tyszer, J.; Filipek, M.; Mrugalski, G. New test compression scheme based on low power BIST. In Proceedings of the 18th IEEE European Test Symposium (ETS), Avignon, France, 27–30 May 2013; pp. 1–6.
98. Chung, Y.; Rau, J. Low-capture-power X-filling method based on architecture using selection expansion. In Proceedings of the IEEE International Conference on Applied System Innovation (ICASI), Chiba, Japan, 13–17 April 2018; pp. 102–104.
99. Yuan, H.; Guo, K.; Sun, X. A power efficient BIST TPG method on don't care bit based 2-D adjusting and Hamming distance based 2-D reordering. *J. Electron. Test.* **2015**, *31*, 43–52. [CrossRef]
100. Yogi, N.; Agrawal, V.D. Application of signal and noise theory to digital VLSI testing. In Proceedings of the 2010 28th VLSI Test Symposium (VTS), Santa Cruz, CA, USA, 19–22 April 2010; pp. 215–220.
101. Yogi, N.; Agrawal, V.D. Spectral RTL Test Generation for Microprocessors. In Proceedings of the 20th International Conference on VLSI Design held jointly with 6th International Conference on Embedded Systems (VLSID'07), Bangalore, India, 6–10 January 2007; pp. 473–478.
102. Yogi, N.; Agrawal, V.D. Transition Delay Fault Testing of Microprocessors by Spectral Method. In Proceedings of the Thirty-Ninth Southeastern Symposium on System Theory, Macon, Georgia, 4–6 March 2007; pp. 283–287.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.