Hamming Code with Reversible Logic

Shashidhar
PG Student ECE
Malla Reddy Engineering College
(Jawaharlal Nehru Technological University)
Hyderabad, India

Dr P Joel Josephson
Associate Professor ECE
Malla Reddy Engineering College
(Jawaharlal Nehru Technological University)
Hyderabad, India

Dr M Jagadeesh Chandra Prasad
Professor and Head ECE
Malla Reddy Engineering College
(Jawaharlal Nehru Technological University)
Hyderabad, India

***Abstract*— Recently, quantum-dot cellular automata (QCA) technology is widely using in variety of applications, which is prominent alternative solution to conventional CMOS technology. Further, the built-in self-test (BIST) environments developed using CMOS technology consuming the higher area, delay, and power measures. Further, the conventional BIST modules are unable to restore the perfect errors in the circuit under test (CUT). Therefore, this work implemented the QCA based BIST module using reversable logic gates (RLG) technology. Further, the CUT considered in the proposed RLG-BIST module is single error correction with double error detection (SEC-DED) based hamming system. Initially, hardware pattern generation (HPG) is implemented using linear feedback shift registers (LFSR), which generates the random test patterns. Then, the test patterns are applied to SEC-DED based hamming system. Further, the output response comparator (ORC) is used to verify the output from SEC-DED with reference values. The RLG-BIST environment is repeated for multiple number of times and applies the corrects the possible errors for all test cases. The simulation results shows that the proposed RLG-BIST resulted in superior performance as compared to conventional BIST methods.**

Keywords*— *quantum-dot cellular automata, built-in self-test, reversable logic gates, CUT, double error detection, single error correction, hamming system.

I. INTRODUCTION

The design of testing of VLSI circuits needs to be improved for the development of VLSI technology [1]. In order to improve the controllability and observability methods of testing, partitioning and random test pattern are necessary [2]. The testing methods should cover 90% of fault coverage. The testing of VLSI circuits with more than 20000 gates can be implemented in simulation. The cost of fault simulation is also to be reduced. Certain guidelines need to be followed in designing of VLSI circuits [3] to make the testing of VLSI simple and fault free. The factors such as test strategy, initialization, synchronous system, test mode logic, wired logic, floating signals, one shot, clock control, power on reset and analog modules are to be considered for the best design of VLSI circuits [4]. The BIST methodology is implemented in IC. There are four parameters to be considered for developing the IC for BIST methodology. They are fault coverage, test set size, hardware overhead and performance overhead. If at all there are some errors in the test pattern produced by the HPG [5], the ORC will not produce and error signal and indicates that the CUT is fault free. This is an undesirable and it is sometimes called as aliasing or masking [6]. The extra hardware required for the implementation of BIST is referred as hardware overhead. It is not desirable to have more hardware for implementing BIST in IC. Less amount of hardware for large CUT is more desirable in BIST methodology. The implementation of BIST methodology may sometimes affect the normal performance of the CUT [7]. There may be delay in response during the normal operation of the CUT. This undesirable characteristic is called as performance overhead. This may sometimes lead to severe than the hardware overhead. The above said four parameters are to be considered for the better implementation of BIST methodology in IC [8]. The test strategy is to be identified at the time of designing itself. Proper test strategy is not determined at the beginning it will be too difficult to identify the fault at later stage. In order to test the VLSI circuit, it is to be driven from a known state [9]. If a reset signal is given then the digital circuits such as flip flops, counters, latches, registers etc. are to be loaded with initial value for proper testing [10]. These initial values are to be loaded by a separate driven circuit. It reset signal is given the driven circuit will load the initial values [11]. This initialization is to be done before test vectors is passed to the CUT. All the components in the VLSI circuit are to be synchronized to avoid race condition. The clock and reset signals are the two types of signals specifically designed for proper synchronization of VLSI circuits. Certain hardware is specifically designed in the chip for testing purpose [12]. This hardware is separated from the normal functions of the chip. In the test mode logic various components of chip is tested. Therefore, the major contributions of this work are as follows:

- Implementation of RLG-BIST using QCA technology with SEC-DED based hamming system as CUT.
- Generation of random test patterns using LFSR mechanism in HPG.
- Verification of SEC-DED based hamming system through ORC, which compares resultant data with reference data.
- RLG-BIST environment is repeated for multiple number of times and applies the corrects the possible errors for all test cases.

Rest of the article is organized as follows: section 2 deals with literature survey, section 3 deals with the proposed RLG-BIST implementation, section 4 deals with analysis of results with performance comparison, section 5 concludes the article with possible future directions.

II. LITERATURE SURVEY

In [13] authors proposed bounds on the length of pseudo exhaustive tests for HPGs for pseudo exhaustive testing and cellular automata pseudo-exhaustive HPGs. A method for cutting down on the amount of power that is wasted by scan-based structures that are used for testing digital circuits was presented in the article [14]. This technique reduces the amount of switching activity as well as the static power. They present a scan structure in addition to a method for finding the optimal configuration of the structure, which leads to the lowest possible dynamic and static power consumptions when the structure is in scan mode. Random-Access Scan (RAS) is a technique that has been suggested by the authors of [15] as a means of simultaneously reducing the amount of power used for testing, the volume of data being analyzed, and the amount of time spent applying the results. The output response of the microchip has been implemented with the help of a Multiple Input Signature Register. In the paper [16], the authors developed a strategy called a post Automatic HPG (ATPG) for the process of lowering power. The hybrid BIST method was suggested by the authors in the article [17] as a method for evaluating the VLSI circuit. The tests that are created by this ATPG may be used to test the chips at their highest possible speed, and there is very little chance that the chips will be damaged as a result of the extreme heat.

The authors of [18] described a technique that modifies the scan chain by introducing logic gates in the spaces between the scan cells. The incorporation of gate delays into the scan route is an idea put forth by this approach. A great number of digital circuits include flaws known as Random Pattern Resistant (RPR) [19], which restrict the area that may be tested using pseudorandom methods. RPR faults are defects that have a low probability of being detected (few patterns detect them). The fault coverage that may be attained using BIST has been recommended to be improved using a number of different methods.

These techniques consist of Weighted pseudorandom patterns, where the random patterns are biased using extra logic to increase the probability of detecting RPR faults, Mixed-mode testing, where the circuit is tested in two phases, Modifying the CUT by test point insertion [20] or by redesigning the CUT, and Weighted pseudorandom patterns, where the random patterns are biased using extra logic to increase the probability of detecting RPR faults [21]. The first process involves the application of pseudorandom patterning. In the second step, deterministic patterns are used to zero in on the flaws that have not yet been discovered [22]. The test wrapper infrastructure circuit consists of digital test control circuit, digital TAM clock, serial to parallel conversion registers. This Analog Test Wrapper (ATW) circuit can be combined with other BIST techniques. The requirement of the analog signal for the BIST can be obtained from the ATW [23].

This ATW approach reduces the testing time and overall cost of the system. This TAM optimization easily handles the analog cores and digital circuits easily. The area of the system is also reduced. The experimental result of testing the ITC'02 SOC with the analog cores is presented [24]. In [25] authors proposed branch and bound algorithm for open defects in the VLSI circuits. The open circuit may take place in two layers of the digital circuits such as inter layer opens and intra layer opens. A branch and bound algorithm were designed, which reduces the functions of ATPG. The flow is designed for detecting the open circuit fault and analyzing the parameters of the digital circuits. By tracking the fanout structure the interconnect open can be detected. Aggressor victim open defect model is designed for the inter layer and intra layer open defects. Commercial parameter extraction tool is used for this model. This model functions by extracting the layout capacitances values obtained from the circuit layout. The open circuit defect model is combined with automatic HPG to improve the testing of VLSI circuits.

III. PROPOSED METHOD

The proposed RLG-BIST is a design for test supplies the testing programmed to the SEC-DED based hamming system CUT. The RLG-BIST architecture consists of three main components such as HPG, ORC and test controller as shown in Figure 1. The HPG generates test vector of the CUT. Actually, the test patterns are stored in the memory with counter and LFSR. The ORC is basically a comparator with stored responses, LFSR, which is used as a signature analyzer. The ORC checks the responses and tests the correctness of the CUT. A controller circuit is required to control and coordinate the HPG, ORC and the CUT. The primary input to multiplexer and wires from output to display devices cannot be covered by the test. During the test of CUT various test patterns are applied to the CUT. In this step, the output signatures are scrutinized in light of the reference golden signatures. Based on the comparison an error signal is generated. The error indicates the CUT is good or faulty one.

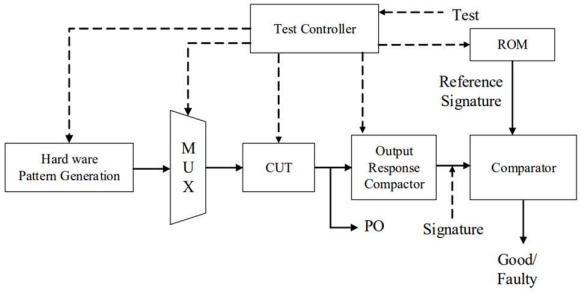

Fig. 1. Proposed RLG-BIST block diagram.

The RLG-BIST is performed in online or offline. In online RLG-BIST the chip is tested during normal operation of the chip, whereas the offline BIST is performed when the chip is not in normal operation.

A. HPG

The number of test patterns produced by the HPG is called as test set size. If the test set size is large, then the fault coverage is also more. If the test set size is small then it will not cover all the faults. The time shifting and repeating operation is to be done on the outputs of LFSR. Figure 2 shows the architecture diagram of LFSR, which is developed by using RLG.

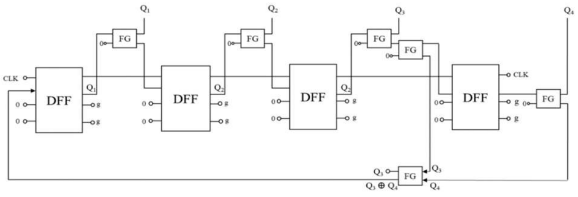

Fig. 2. LFSR using RLG.

It is possible to create an LFSR by performing an exclusive-OR operation on the outputs of two or more flip-flops together and then applying this output to one of the flip-flops. Although the LFSR is used to generate periodic sequence, it does not produce all zero. Figure 2 depicts the layout of a reversible 3-bit low frequency shift register. In order to replicate the output of any two flip-flops, a Feynman gate must be placed in between them. In addition to that, the exclusive-OR operation on the feedback route is performed with its help. It is important that Q1, Q2, and Q3 do not begin with all zeros at the beginning; otherwise, the LFSR would create a pattern of output that is all zeros whenever a clock pulse is applied. LFSR has a pattern count equal to 2n minus 1, where n is the total number of flip-flops in the circuit. The LFSR and phase shift registers increases the effectiveness of fault detection. The phase shift register is designed by XOR gates. The output of the CUT will compact by multiple input shift register. The output of the CUT will be compared with fault free signature to test the circuit. Large amount of VLSI chips is produced every day for various applications. There may some defects or fault during the manufacturing process. If there is fault that needs to be cleared before it is used. The test patterns are generated and sent to the fault less VLSI chip and actual VLSI chip of same architecture. The outputs of fault less VLSI chip is compared with actual VLSI chip by EXOR gate. The EXOR gate produce low output if there is no fault. It produces high output when there is difference between the outputs of two VLSI chip and means the actual VLSI chip is faulty one.

B. CUT

This work considers the CUT as hamming encoder and decoder with SEC-DED properties. Figure 3 shows the hamming encoder circuit with RLG technology. The hamming encoder contains the A, B, C, D as inputs, and A, B, C, D, P1, P2, P4 as outputs, where P1, P2, P4 are parity bits. The XOR operation is performed on the input vectors to produce the encoded outcomes using Fredkin gate (F), Double Feynman gate (F2G) and Feynman gate (FG) based RLG. The consequence of this hamming encoder circuit output is taken into consideration for the check bit generation (CBG) cell input.

Figure 4 shows the block diagram of CBG implemented using CBG. The hamming encoded outcomes such as A, B, C, D, P1, P2, P4 attacked by different types of errors. So, CBG performs the error location detection operation by performing the parity matching. The CBG operation verifies the received parity with even parity rules and generates the check bits (C1, C2, C4). These check bits are applied to error detection and correction (EDC) unit. Figure 5 shows the block diagram of RLG based EDC. It has been noted that the data that was received differs from the data that was originally sent from the block that is above. To correct the problem, we must first use the C4C2C1 combination to make a value change in the location of the error bit. Here, the EDC unit contains inputs as A, B, C, D, P1, P2, P4, C1, C2, C4. The EDC unit compares the C1, C2, C4 with P1, P2, P4 and corrects the various errors in A, B, C, D.

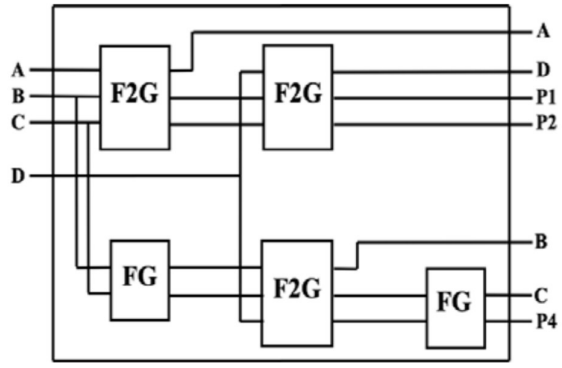

Fig. 3. Hamming encoder using RLG.

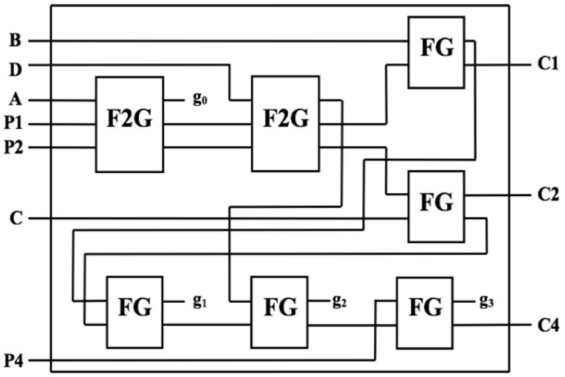

Fig. 4. Block diagram of CBG.

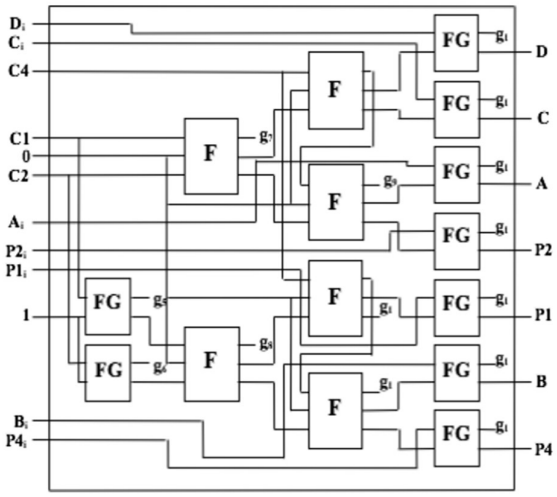

Fig. 5. RLG based EDC block diagram.

C. Test Controller

BIST is a design technique of testing the CUT by placing the testing functions within the CUT itself. The three important structures of BIST are HPG, output analyzer and test controller. The function of the HPG is to generate required test pattern for the CUT. Some of the examples of HPGs are LFSR, counter or ROM with already stored testing data. ORC is a type of Comparator, which stores model outputs for comparison with present outputs of the CUT. The rest controller is circuit to give command signal to the HPG to release test patterns for testing. It also sends signals for the ORC to analyses the present output with the already stored

output. The test related functions are carried out in the test controller.

D. ORC

The test patterns are generally generated to test the chip at three different levels. They are application level, functional level and structure level. The application level of HPG is done by customer. The customer can generate pattern to test the chip. This application level HPG may not guarantee to be 100 percent defect free. The function level HPG used to test at subunits, modules and black boxes level. In this the input patterns are tested with the output pattern. The function level HPG will test the full capability of the chip, but it will not give 100 percent assurance of defect free. It the structure of the chip is available then tools such as CAD and algorithms are available to test the chip. In VLSI there are certain guidelines are available to test the entire capabilities of the chip. The guidelines are available in the areas of test strategy, initialization, test mode logic, wired logic, floating signals, one shot, clock control, power on reset and analog modules. In the normal mode of operation, the CUT will receive signals from the input sources and generates output signals to other devices. It will not take any BIST related signals. During BIST operation only the CUT will receive signals from HPG. The response of the CUT will be tested in the ORC. The signals from ORC are compared with reference signals already stored in the chip. The comparator generates error signals. The error signals indicate the CUT is a faulty or good.

IV. RESULTS AND DISCUSSIONS

Xilinx ISE software was used to create all of the BIST designs. This software programmed gives two types of outputs: simulation and synthesis. The simulation results provide a thorough examination of the BIST architecture in terms of input and output byte level combinations. Decoding procedure approximated simply by applying numerous combinations of inputs and monitoring various outputs through simulated study of encoding correctness. The use of area in relation to the transistor count will be accomplished as a result of the synthesis findings. In addition, a time summary will be obtained with regard to various path delays, and a power summary will be prepared utilizing the static and dynamic power consumption.

Device Utilization Summary (estimated values)			[-]
Logic Utilization	**Used**	**Available**	**Utilization**
Number of Slice Registers	512	35200	1%
Number of Slice LUTs	12142	17600	68%
Number of fully used LUT-FF pairs	348	12306	2%
Number of bonded IOBs	24	100	24%
Number of BUFG/BUFGCTRLs	2	32	6%
Number of DSP48E1s	1	80	1%

Fig. 6. Design summary.

Figure 6 shows the design (area) summary of proposed method. Here, the proposed method utilizes the low area in terms of slice LUTs i.e., 12142 out of available 17600. Further, the proposed method utilizes the slice registers as 512, out of available 35200. Further, the proposed method utilizes fully used LUT-FF as 348, out of available 12306. Further, the proposed method utilizes buffers (BUF) as 2, out of available 32.

```
Data Path: u6/out to bist_out
                                   Gate     Net
  Cell:in->out      fanout   Delay   Delay  Logical Name (Net Name)
  ----------------------------------------  ------------
   FD:C->Q               1   0.232   0.279  u6/out (u6/out)
   OBUF:I->O                 0.000          bist_out_OBUF (bist_out)
  ----------------------------------------
  Total                      0.511ns (0.232ns logic, 0.279ns route)
                                     (45.4% logic, 54.6% route)
```

Fig. 7. Time summary.

Figure 7 shows the time summary of proposed method. Here, the proposed method consumed total 0.511ns of time delay, where 0.232ns of delay is logical and 0.279ns of delay is route.

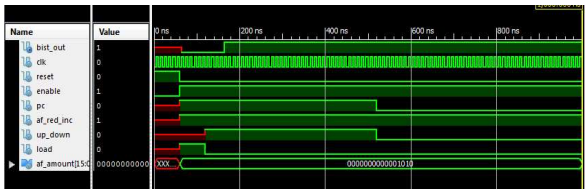

Fig. 8. Simulation outcome.

Figure 8 presents the simulation outcome of proposed system. Here, clock (clk), reset, enable, program_counter(pc), activity_factor_reduction_increment (af_red_inc), up_down, load, and af_amount are the input data pins. Further, bist_out is the output pin. During the active high reset, system is intilized to zero and during active low reset, system starts work. The system is disable during the active low enable, pc, and system starts work during active high enable, pc. If the load is active high, then the af_amount is loaded in BIST environment. Further, active high of af_red_inc resulted in increment of activity factor and active low of af_red_inc resulted in decrement of activity factor. In addition, active high of up_down resulted in increment counting and active low of up_down resulted in decrement of counting. Finally, active low of bist_out shows BIST fail condition and active high of bist_out shows BIST pass condition.

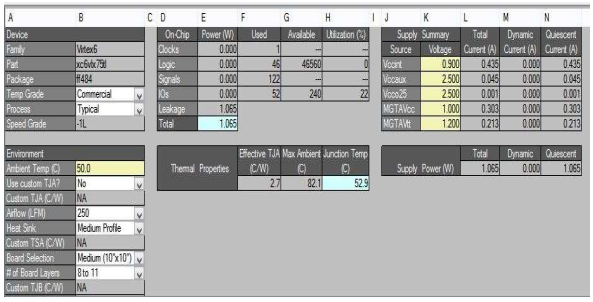

Fig. 9. Power summary.

Figure 9 shows the power consumption report of propsoed RLG-BIST. Here, the propsoed RLG-BIST consumed power as 1.065 watts. Table 1 compares the performance evaluation of various BIST controllers. Here, the propsoed RLG-BIST resulted in superior (reduced) performance in terms of LUTs, slice registers, LUT-FFs, time-sdelay, and power consumption as compared to conventional approaches such as MBIST [22], PSRG-BIST [23], and FT-BIST[24].

TABLE I. PERFORMANCE EVALUTION.

Metric	**MBIST [22]**	**PSRG-BIST [23]**	**FT-BIST [24]**	**Proposed RLG-BIST**
Slice Registers	784	734	673	512
LUTs	18367	17352	15327	12142

LUT-FFs	826	736	635	348
Time delay (ns)	0.927	0.836	0.726	0.511
Power consumption (w)	32.482	24.1939	16.937	1.1065

CONCLUSION

The RLG-BIST module was implemented with QCA technology as a result of this work. In addition, the CUT that has been taken into consideration for the proposed RLG-BIST module is a hamming system that is based on SEC-DED. LFSR is initially used to implement HPG because it generates random test patterns and is therefore essential to the process. After that, the test patterns are utilized by a hamming system that is based on SEC-DED. In addition, the output from the SEC-DED is fed into the ORC so that reference values can be checked against it. The RLG-BIST environment is run a large number of times, and each and every test case has any and all potential errors fixed that were found during those runs. The findings from the simulation show that the RLG-BIST method proposed resulted in superior performance when compared to traditional BIST methods. This work can be extended with advanced BIST architectures for more bit patterns generation.

REFERENCES

[1] Ahmed, Mohammed Altaf, and Suleman Alnatheer. "Deep Q-Learning with Bit-Swapping-Based Linear Feedback Shift Register fostered Built-In Self-Test and Built-In Self-Repair for SRAM." Micromachines 13.6 (2022): 971.

[2] Mrozek, I., N. A. Shevchenko, and V. N. Yarmolik. "Universal Address Sequence Generator for Memory Built-in Self-test." arXiv preprint arXiv:2208.05325 (2022).

[3] Sun, Yang, and Spencer K. Millican. "Applying Artificial Neural Networks to Logic Built-in Self-test: Improving Test Point Insertion." Journal of Electronic Testing (2022): 1-14.

[4] Ince, Mehmet, Bora Bilgic, and Sule Ozev. "Digital Fault Based Built-in Self-Test and Evaluation of Low Dropout Voltage Regulators." ACM Journal on Emerging Technologies in Computing Systems (JETC) (2022).

[5] Madhulatha, Kamma, et al. "Reconfigurable and Parameterizable Pseudorandom Pattern Generators for Built-in Self Test." 2022 7th International Conference on Communication and Electronics Systems (ICCES). IEEE, 2022.

[6] Gopalan, Karthy, and Sivakumar Pothiraj. "Retraction Note to: A saboteur and mutant based built-in self-test and counting threshold-based built-in self repairing mechanism for memories." Journal of Ambient Intelligence and Humanized Computing (2022): 1-1.

[7] Suriyan, Kannadhasan, et al. "Power analyzer of linear feedback shift register techniques using built in self test." Bulletin of Electrical Engineering and Informatics 11.2 (2022): 713-721.

[8] Bilgic, Bora, and Sule Ozev. "Performance Degradation Monitoring for Analog Circuits Using Lightweight Built-in Components." 2022 IEEE 40th VLSI Test Symposium (VTS). IEEE, 2022.

[9] Pavlidis, Antonios, et al. "SymBIST: Symmetry-based analog and mixed-signal built-in self-test for functional safety." IEEE Transactions on Circuits and Systems I: Regular Papers 68.6 (2021): 2580-2593.

[10] Chaudhuri, Arjun, et al. "Built-in Self-Test and Fault Localization for Inter-Layer Vias in Monolithic 3D ICs." ACM Journal on Emerging Technologies in Computing Systems (JETC) 18.1 (2021): 1-37.

[11] Emara, A. S., Romanov, D., Roberts, G. W., Aouini, S., Ziabakhsh, S., Parvizi, M., & Ben-Hamida, N. (2021). An Area-Efficient High-Resolution Segmented ΣΔ-DAC for Built-In Self-Test Applications. IEEE Transactions on Very Large Scale Integration (VLSI) Systems, 29(11), 1861-1874.

[12] Ahmed, Mohammed Altaf, and Suleman Alnatheer. "Deep Q-Learning with Bit-Swapping-Based Linear Feedback Shift Register fostered Built-In Self-Test and Built-In Self-Repair for SRAM." Micromachines 13.6 (2022): 971.

[13] Xie, C., Li, J., & Wang, W. (2022). A package for piezoresistive pressure sensors with embedded built-in self-test function based on bimetallic actuator. Sensors and Actuators A: Physical, 113817.

[14] Maity, D. K., Roy, S. K., & Giri, C. (2022). A Cost-Effective Built-In Self-Test Mechanism for Post-Manufacturing TSV Defects in 3D ICs. ACM Journal on Emerging Technologies in Computing Systems (JETC).

[15] Mrozek, I., N. A. Shevchenko, and V. N. Yarmolik. "Universal Address Sequence Generator for Memory Built-in Self-test." arXiv preprint arXiv:2208.05325 (2022).

[16] Choi, S., Aoki, Y., Park, H. C., Yang, S. G., & Song, H. J. (2021, June). Sequential loopback built-in self-test algorithm for dual-polarization millimeter-wave phased-array transceivers. In 2021 IEEE Radio Frequency Integrated Circuits Symposium (RFIC) (pp. 55-58). IEEE.

[17] Sun, Yang, and Spencer K. Millican. "Applying Artificial Neural Networks to Logic Built-in Self-test: Improving Test Point Insertion." Journal of Electronic Testing (2022): 1-14.

[18] Murugan, S. Vijay, and B. Sathiyabhama. "Bit-swapping linear feedback shift register (LFSR) for power reduction using pre-charged XOR with multiplexer technique in built in self-test." Journal of Ambient Intelligence and Humanized Computing 12.6 (2021): 6367-6373.

[19] Sparkman, B., Smith, S. C., & Di, J. (2022). Built-In Self-Test for Multi-Threshold NULL Convention Logic Asynchronous Circuits using Pipeline Stage Parallelism. Journal of Electronic Testing, 1-14.

[20] Lee, H., Oh, H., & Kang, S. (2021). On-Chip Error Detection Reusing Built-In Self-Repair for Silicon Debug. IEEE Access, 9, 56443-56456.

[21] Shoer, S., Karady, T., Keshet, A., Shilo, S., Rossman, H., Gavrieli, A., ... & Segal, E. (2021). A prediction model to prioritize individuals for a SARS-CoV-2 test built from national symptom surveys. Med, 2(2), 196-208.

[22] Murugan, S. Vijay, and B. Sathiyabhama. "Bit-swapping linear feedback shift register (LFSR) for power reduction using pre-charged XOR with multiplexer technique in built in self-test." Journal of Ambient Intelligence and Humanized Computing 12.6 (2021): 6367-6373.

[23] Garbolino, Tomasz. "A New, Fast Pseudo-Random Pattern Generator for Advanced Logic Built-In Self-Test Structures." Applied Sciences 11.20 (2021): 9476.

[24] Zaree, M., & Raji, M. (2021, March). FT-LFSR: A Fault Tolerant Architecture for Linear Feedback Shift Registers. In 2021 26th International Computer Conference, Computer Society of Iran (CSICC) (pp. 1-6). IEEE.

[25] Ince, M., Yilmaz, E., Fu, W., Park, J., Nagaraj, K., Winemberg, L., & Ozev, S. (2021). Fault-based Built-in Self-test and Evaluation of Phase Locked Loops. ACM Transactions on Design Automation of Electronic Systems (TODAES), 26(3), 1-18.