

ARCHITECTURAL DESIGN OF BIST USING MULTISTAGE LFSR ALGORITHM IN VLSI TECHNOLOGY

Fariha Fatma, Abu rehan,
M.Tech Scholar, Assistant Professor
Al-Falah University.

ABSTRACT : In this digital world rather than construction, testing the built architecture has become the challenging task. Testing process includes high cost and power consumption. Many studies have taken part in construction of efficient testing circuits, in that BIST is one of the efficient testing circuit. BIST [Built in Self Test] provides a platform for testing the circuit with low power consumption and with spending fewer bucks. The construction of the BIST is done with the MULTISTAGE LFSR decoder circuits, which makes a path in testing the circuit by providing random and complete input sequences to the built architecture. The decoding logic is also employed to make it perfect for fault tolerant architecture. The pavement made of the BIST with MULTISTAGE lfsr is said to be the efficient technique in finding the faults in the working of the circuitry so this is termed as the fault tolerant architecture the construction of the proposed architecture is done in Xilinx ISE proceeding with verilog HDL language.

Index Terms—BIST, MULTISTAGE lfsr , decoding logic, linear-feedback shift register (LFSR), Bench mark circuit.

1. INTRODUCTION

With Recent advances in applications, for example, single-photon location, it has gotten essential to execute large number of arrayed counters in less area. These incorporate time-of-flight (TOF) extending top to bottom cameras where counters are required to check clock cycles and furthermore photon-counting cameras that include the quantity of photons in an interval. Diminishing the region devoured by the counter in these applications is basic in expanding the quantity of pixels in the cameras, as every camera pixel contains a different counter. While linear-feedback shift registers (LFSRs) are normally utilized as pseudorandom number generators, it has been indicated that they are additionally a productive method to actualize random counters and are appropriate to huge arrayed designs, as the move register can go about as a sequential readout component. LFSR counters have been utilized in the CMOS pixel plan and in single-photon detection.

The clock speed of a LFSR is free of the quantity of bits in the counter, and they cross all states in the state space aside from the every one of the zero state. Be that as it may, the tally request of LFSRs is pseudorandom, so additional handling is required to decipher the LFSR state into double request. Three distinct procedures to interpret the LFSR grouping into parallel are thought about in: the iteration technique, the lookup table (LUT) strategy, and a time-memory tradeoff algorithm. The iteration technique emphasizes over the whole tally succession of the LFSR and thinks about each to the counter worth. For a n -bit LFSR, this requires roughly $2n-1$ correlations by and large. The direct LUT technique rather utilizes a $n \times n$ LUT that legitimately translates the LFSR state. The time-memory tradeoff calculation presented in consolidates the two strategies by putting away $2^{(N/2)}$ LFSR include values in a table and emphasizing over the LFSR succession until the include esteem coordinates an incentive in the table. The quantity of emphases is then deducted from the put away an incentive to acquire the decoded esteem. Another calculation dependent on discrete logarithms was presented in and was adjusted for use with ring generator event counters in.

Applications with large arrays require every cell in the array in the exhibit to be decoded to double request for additional processing, and for system on-chip structures, it is important to play out decoding on chip. This requirement directs that the deciphering rationale must be integrable and quick, since numerous transformations need to happen. Be that as it may, the entirety of the previously mentioned techniques develop exponentially in either time or memory with the size of the LFSR. For single-photon identification applications, there are several instances of displayed plans that would not have the option to be executed with LFSR counters without restrictively enormous coordinated LUTs. This venture proposes another counter structure dependent on numerous LFSR stages, which can be decoded with rationale that develops logarithmically with the counter size instead of exponentially. While a direct connection of LFSR counters would cause a critical exhibition decrease, like double wave counters, this paper acquaints a strategy with appropriate the ripple signal in time and makes up for this his paper introduces a technique to distribute the ripple signal in time and compensates for this in a generalized decoding logic scheme.. All through the rest of this paper, a n -bit LFSR will be alluded to as a n -LFSR.

LINEAR-FEEDBACK SHIFT REGISTER (LFSR)

In figuring, a linear-feedback shift register (LFSR) is a shift register whose information bit is a linear function of its past state. The most regularly utilized linear function of single pieces is select or (XOR). In this manner, a LFSR is regularly a shift register whose information bit is driven by the XOR of certain bits of the general shift register esteem. The underlying estimation of the LFSR is known as the seed, and in light of the fact that the activity of the register is deterministic, the flood of qualities delivered by the register is totally controlled by its current (or past) state. In like manner, in light of the fact that the register has a limited number of potential states, it should inevitably enter a rehashing cycle. Nonetheless, a LFSR with an all well chosen picked criticism capacity can deliver an arrangement of pieces that seems irregular and has a long cycle. Utilizations of LFSRs incorporate creating pseudo-random numbers, pseudo- noise sequences, quick advanced counters, and brightening arrangements. Both equipment and programming usage of LFSRs are normal. The arithmetic of a cyclic redundancy check, used to give a quick check against transmission mistakes, are firmly identified with those of a LFSR .LFSRs can be actualized in equipment, and this makes them valuable in applications that require quick generation of a pseudo- random sequences, for example, direct-arrangement spread spectrum radio. LFSRs have likewise been utilized for creating a approximation of background noise different programmable sound generators.

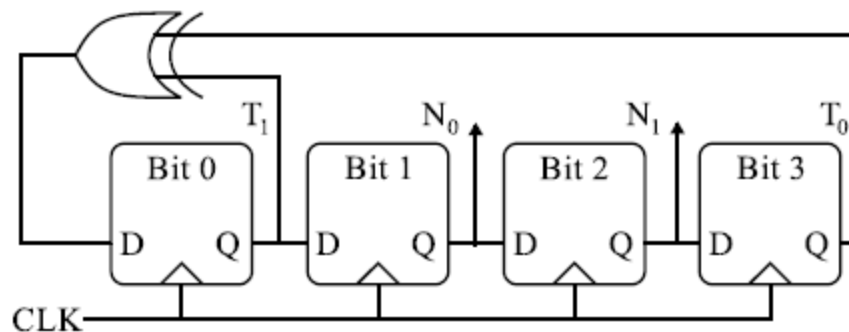


Fig1 : Structure of conventional many-to-one 4-LFSRs.

In built in self-test(BIST) procedures, putting away all the circuit yields on chip is unimaginable, however the circuit yield can be compressed to frame a mark that will later be contrasted with the brilliant mark (of the great circuit) to recognize deficiencies. Since this compression is lossy, there is consistently a like faulty output that yields likewise creates a similar signature as the brilliant mark and the shortcomings can't be distinguished. This condition is called error masking or associating. BIST is practiced with a multiple-input signature register (MISR or MSR), which is a kind of LFSR. A standard LFSR has a solitary XOR or XNOR gate, where the contribution of the gate is associated with a few "taps" and the yield is associated with the contribution of the main flip-flop. A MISR has a similar structure, however the contribution to each flip-flop is taken care of through a XOR/XNOR gate. For instance, a 4-bit MISR has a 4-bit equal yield and a 4-bit equal info. The contribution of the main flip-flop is XOR/XNORd with equal information bit zero and the "taps". Each and every other flip-flop input is XOR/XNORd with the proceeding flip-flop yield and the comparing equal information bit.

2. Literature Survey

Astonishing advancement in Silicon gadgets and circuits and profoundly solid mass manufacturing techniques have incited remarkable upheavals in hardware throughout the previous 4 decades to the degree that hardware is growingly saturating various parts of our life. The application world, consumer and infrastructure, is currently used to exponential execution enhancements and high return. Multifaceted nature and seriousness of present day electronic frameworks interest for advancement over different orders. Joint advancement of gadget, circuit, bundling, and test are progressively significant for superior frameworks. Plan for Manufacturing and Testing is more basic yet progressively testing in complex frameworks. So I don't get optimization's meaning for the fate of ever-developing and complex strong state hardware? An overwhelming test worth spending a discussion on. Allen C. Cheng - Comprehensive Study on Designing Memory BIST: Algorithms, Implementations and Trade-Offs, Memories are the most all inclusive segments today. Practically all framework chips contain some kind of inserted memory, for example, ROM, SRAM, DRAM and glimmer memory. Testing of these recollections is a difficult undertaking as testing time, overhead region and cost of the test assumes a significant work during testing. The direct getting to of installed recollections is profoundly troublesome. The testing of these recollections done through a BIST (Built-in-individual test) Controller The Microcode based Memory BIST regulator can be demonstrated to be perhaps the best strategy to test the recollections. For this we are utilizing March C/C+ calculations, since March based tests are basic and have great flaw inclusion. Microcode based Memory BIST comprises of Program counter, Instruction decoder, Address generation block and Memory interface unit.. It has Address register, Data register, Control register, Multiplexers and Comparator. These blocks are

structured by utilizing Verilog HDL code, recreated by NC Verilog compiler and combined by RTL Compiler. These squares are incorporated with program counter and microcode decoder to frame a Microcode based Memory BIST Controller.

3. Multistage Linear Feedback Shift Register Counters With Reduced Decoding Logic

A) Multistage LFSR Counter

The overall plan of the counter structure is appeared in Fig. 2. There are M indistinguishable n -LFSR blocks that are controlled by an enable signal. When the $(m - 1)$ th n -LFSR experiences a specific state change, the enable signal is declared so that the m th n -LFSR propels one state. This permits the whole $M \times n$ bit state space to be navigated. In large array design, the counter can likewise go about as a rapid sequential readout chain. This is accomplished with insignificant extra rationale that by passes the LFSR input and wave convey squares. The multistage counter plan diminishes the counter into M autonomous modules, permitting every n -LFSR to be decoded independently by a $n \times n$ bit LUT instead of a $(M \times n) \times (M \times n)$ bit LUT. For little n , the LUT can without much of a stretch be implemented on chip.

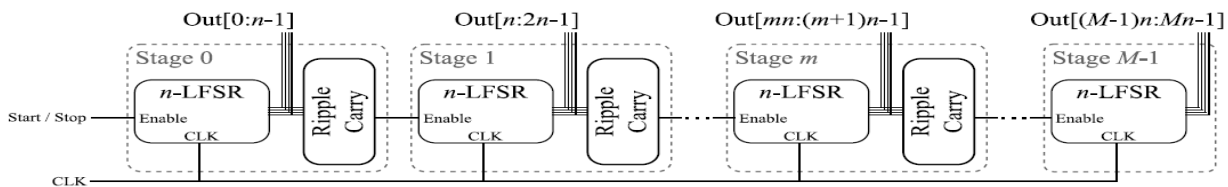


Fig. 2. Block diagram of the multistage LFSR counter.

B). LFSR Block

Each stage of the counter is set off once per time of the past stage, so missing states from the LFSR arrangement will make huge blocks of counter states be absent from the counter state space. Subsequently, it is significant that the n -LFSR is intended for a maximal length. The maximal arrangement length of a n -LFSR is just $2n - 1$, so additional logic is required to join the missing state into the tally grouping. This can be accomplished utilizing a NOR and XOR function to incapacitate the input logic when the $0x000 \dots 1$ state is recognized, as appeared in Fig. 3. This grouping augmentation rationale broadens the arrangement length of the individual segment LFSRs to $2n$ with the goal that the counter covers each state in the $2M \times n$ state space. This additionally permits the multistage counter to be utilized in applications that require each state to be secured, for example, self-beginning counters, where conventional LFSRs would not be pertinent. n -LFSR hinder with arrangement augmentation logic (dotted parts). The whole input square is actualized as a single logic blocks

A few LFSR input styles exist, including many-to-one, one-to-many (then again known as Fibonacci and Galois LFSRs, individually), and ring generators. Ring generators[depicted in Fig.1 ordinary LFSR] are ordinarily viewed as the ideal method to actualize a LFSR, where the move register shapes a ring and taps structure sub loops inside the ring. Be that as it may, the arrangement expansion requires extra logic in the LFSR, commanding the basic way. Rather, many-to-one style LFSRs[Fig1. 4bit ordinary lfsr] are utilized, permitting the input logic and the arrangement augmentation logic to be consolidated into a solitary logic blocks for logic minimization as appeared in Fig. 2. The multistage counter permits adaptability in decision of the size of the n -LFSR, with the goal that little single-tap LFSRs are specially picked. A solitary tap many-to-one LFSR is topologically unclear from the comparing ring generator. Featured states require further preparing by the unraveling logic.

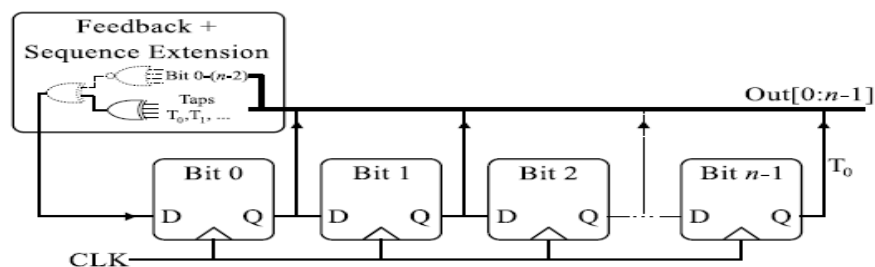


Fig 3: Structure of a proposed multistage

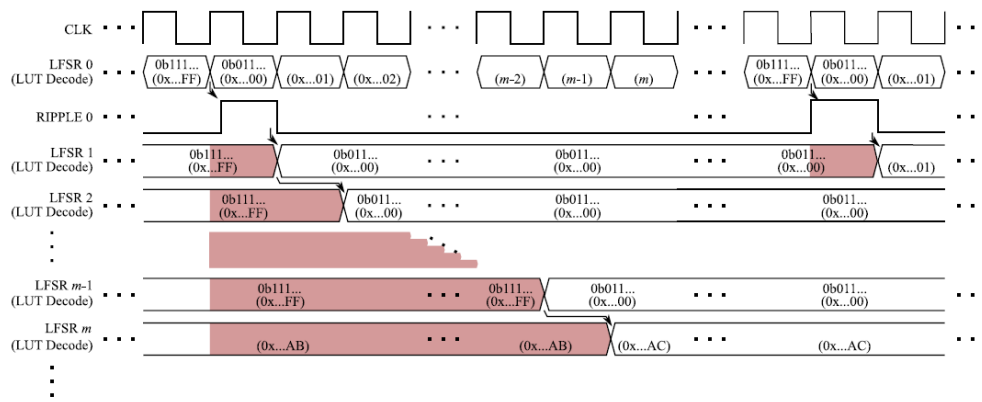


Fig. 4. Timing diagram of the operation of the multistage LFSR counter. Arrows show the operation of the ripple-carry logic.

C). Ripple-Carry Logic

Since the n-LFSR contains each state in the state space, the LFSR must incorporate the 0b1111 . . . _ 0b0111 . . . change. This state is a Gray-code progress and happens in each n-LFSR structure, so it is a perfect wave trigger change. This sets the beginning of the n-LFSR arrangement to 0b0111 . . . so it is decoded by the unraveling logic to 0x . . . 00. If the counter was planned with the goal that a LUT could legitimately interpret each stage accurately in a solitary clock cycle, the wave sign would need to spread through each stage and recognize if each stage will change. Rather, to keep the presentation of the counter from diminishing with each additional stage added to the counter, the wave signal just follows up on the direct next stage and the wave signal for the resulting stages is conveyed to the following clock cycle. This circulates the change edge after some time and, for the mth stage, includes a m clock cycle postponement to the progress edge.

The counter planning outline that exhibits the activity of the wave convey logic is appeared in Fig. 4. Each LFSR state is given as a twofold worth (0b . . .), though the state subsequent to disentangling with a LUT (the LUT Decode signal) is the hex an incentive in sections (0x . . .) for each state. When LFSR 0 changes from the 0b1111 . . . state to the 0b0111 . . . express, the RIPPLE 0 sign is produced. On the following clock edge, the RIPPLE 0 sign follows up on LFSR 1 making it likewise experience the 0b1111 . . . _ 0b0111 . . . progress. This, thusly, likewise creates a wave sign to follow up on LFSR 2 on the following clock edge. Along these lines, the wave convey rationale causes the progress edge to be deferred one clock cycle for each stage. The postponed change makes a blunder triangle structure, appeared by featured states in Fig. 5. These states are decoded erroneously by the LUT and in this way should be revised with a minor measure of interpreting rationale notwithstanding the n x n bit LUT.

D). Decoding Logic

The unraveling logic goes about as a post handling step on the multistage LFSR counter exhibit yield. As each LFSR esteem is perused out of the exhibit, it is gone through a LUT. Fig. 5 shows that the LUT remedies most states to paired request. In any case, extra logic is required to address the blunders brought about by the deferred change. Two sorts of LUT interpret errors happen: starting errors and over flow of errors. Introductory errors happen in the states on the upper edge of the progress mistake triangle when the counter is halted on the clock cycle before the mth stage advances. The decoded estimation of the past stages is likewise the quantity of clock cycles since the beginning of the change edge. Since the wave takes m clock cycles to come to the mth stage, these errors can be distinguished if the decoded estimation of the past stages is equivalent to m-1. Overflow errors happen when the past stage has a errors and is equivalent to 0x . . . FF. These errors demonstrate that a past stage ought to have caused a wave occasion on a prior clock cycle.

The disentangling logic that identifies and adjusts these errors is appeared in Fig. 5. The error recognition of each stage relies upon the decoded estimation of the past stages, so each stage is handled consecutively. On the off chance that a error condition on the following stage is distinguished, the following stage invalid register is set, with the goal that it is revised on the following clock cycle. The blunders are just ever one not exactly the right worth, so the following stage invalid chooses either the LUT yield or adds one to the LUT yield. A over flow error in the following stage will happen if the current stage is a error and furthermore 0x . . . FF. This can be recognized by ANDING the incremter carryout with the following stage invalid register. Beginning errors can be identified by putting away the recently decoded stages in locks and contrasting and a counter that monitors the current stage number. On the off chance that the current stage is equivalent to the recently decoded esteem, the following stage will have an underlying error. The counter just needs to tally to M and along these lines needs $y = \lfloor \log_2(M) \rfloor$ bits. In this manner, just a y-bit correlation should be made between the past decoded state and the counter, so $Z = \lfloor y/n \rfloor$ stages should be stored. The zeros register is utilized to guarantee that every other bit in the recently decoded esteem are zero so the examination is substantial. The last tally worth will be off by one, yet this can be adjusted by adding one to the last check esteem whenever required. An opportunity to disentangle the tally worth will rely upon both M and n. The deciphering

logic takes one clock cycle for every phase of the counter, so M clock cycles are required to translate the whole check esteem. The critical path of the translating logic is execution subordinate however conceivably incorporates the LUT, the incrementer, and the correlation with the stage counter, all of which have decreased execution for enormous n . Along these lines, the most extreme clock rate will be lower for bigger n esteems, however more stages will be required for an identically measured counter with a bigger n esteem, requiring less clock cycles to translate the counter. In the event that the readout of the cluster is performed sequentially, at that point the deciphering logic can be set up as a pipeline in the readout chain. This lone includes a M clock cycle pipeline deferral to the whole exhibit readout, however all individual counter qualities will be perused out with a decoded esteem.

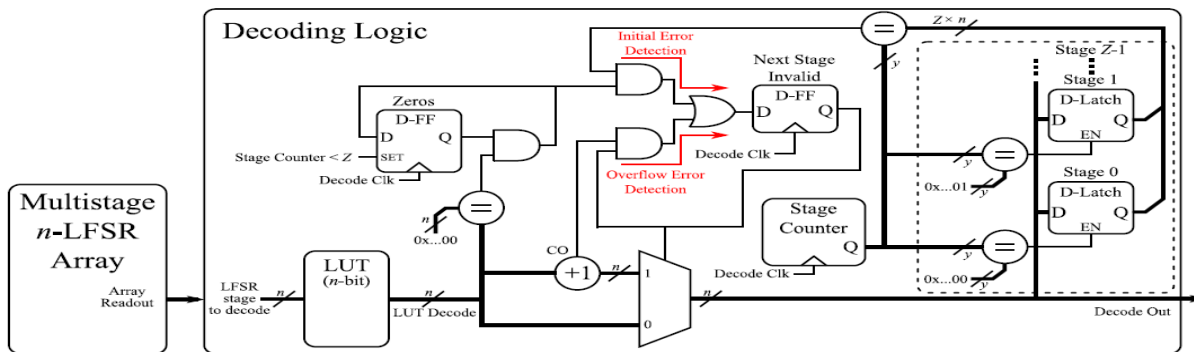


Fig.5. Logic to decode the multistage LFSR counter state order into binary. Each stage is decoded separately in sequence.

4. PROPOSED WORK (FAULT COVERAGE BY USING LFSR DECODER)

The test design generator produces test vectors that are applied to the tried circuit during pseudo-arbitrary testing of combinational circuits. The idea of the generator hence straightforwardly impacts the fault coverage accomplished. The impact of the kind of pseudo-random example generator on stuck-at fault coverage. Linear feedback shift registers (LFSRs) cluster counters are generally utilized as test design generators, and the producing polynomial is crude to guarantee the most extreme period. Built-in self test empowers the chip to test itself and to assess the circuit's reaction. Hence, the extremely unpredictable and costly outer ATE (Automatic Test Equipment) might be totally excluded, or its multifaceted nature altogether diminished. Also, BIST empowers a simple access to inward structures of the tried circuit, which are very difficult to reach from outside. There have been proposed numerous BIST equipment design methods. In the majority of the cutting edge techniques an a pseudorandom design generator (PRPG) is utilized to deliver vectors to test the circuit. These vectors are applied to the circuit either as they may be, or the vectors are adjusted by some extra hardware so as to get better fault coverage. At that point the circuit's reaction to these vectors is assessed in a reaction analyzer. Generally, linear feedback shift registers (LFSRs) or cell automata (CA) are utilized as PRPGs, for their effortlessness. Examples created by basic LFSRs or CA regularly don't give an acceptable fault coverage. Hence, these examples must be adjusted by one way or another. One of the most realized methodologies is the weighted irregular example testing. Here the LFSR code words are changed by a weighting rationale to create a test with given probabilities of event of 0's and 1's at the specific circuit under test (CUT) inputs. Numerous papers managing the calculation of the loads and the structure of the weighting rationale have been distributed. There are two issues in the weighted testing included: the path how to process the loads and the route how to structure the weighting rationale. Presently the LFSR width might be subjectively scaled. The adequacy of this scaling and its potential degrees are reported on BIST equipment structure models for a portion of the standard ISCAS benchmarks. The fundamental rule of the calculation comprises in attempting to "coordinate" tried plan yield with the same number of decoder yields with its contributions, by finding a reasonable vector requesting. On the off chance that a yield is coordinated with yield, here comparator utilized as a test reaction analyzer, when there is no fault then the yield of TRA is 1 in any case the tried plan is faulty. A ordinary BIST architecture comprises of 1. TPG - Test Pattern Generator, 2. TRA - Test Response Analyzer and 3. Control Unit. It generates test pattern for CUT. It will be devoted circuit or a microchip. Example produced might be pseudo arbitrary numbers or deterministic succession. Here we are utilizing a linear feedback shift Register cluster counter for producing random numbers.

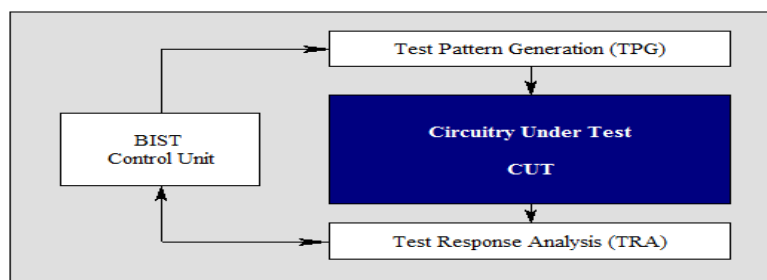


Fig 6. Test Pattern Generator

Test Response Analyzer (TRA): TRA will check the output of MISR & verify with the input of LFSR & give the result as error or not.

BIST Control Unit: Control unit is used to control all the operations. Mainly control unit will do configuration of CUT in test mode/Normal mode, feed seed value to LFSR, Control MISR & TRA. It will generate interrupt if an error occurs.

Circuit under Test (CUT): CUT is the circuit or chip in which we are going to apply BIST for testing stuck at zero or stuck at one error.

Need for using BIST technique

Today’s highly integrated multi-layer boards with fine-pitch ICs are virtually impossible to be accessed physically for testing. Traditional board test methods which include functional test, only accesses the board’s primary I/Os, providing limited coverage and poor diagnostics for board-network fault. In circuit testing, another traditional test method works by physically accessing each wire on the board via costly "bed of nails" probes and testers. The proposed design is reliable testing method which will reduce the cost of test equipment, a research to verify each VLSI testing problems has been conducted.

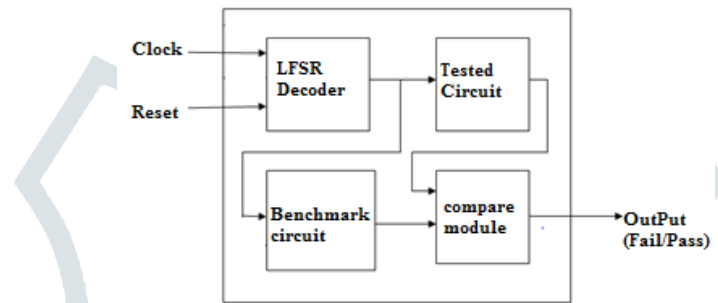


Fig 7: circuit diagram of proposed design

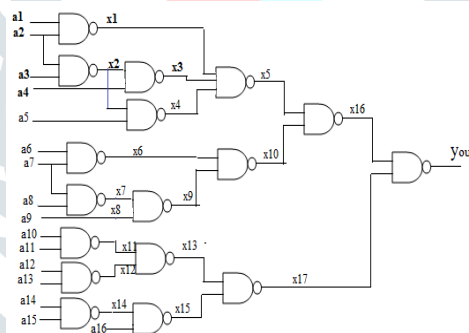


Fig8: Benchmark Circuit

5. Results

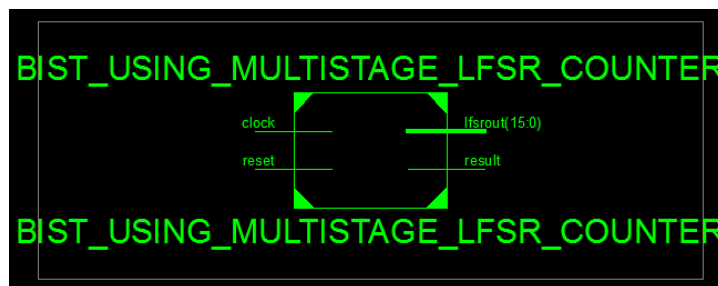


Fig9 : RTL schematic of proposed design

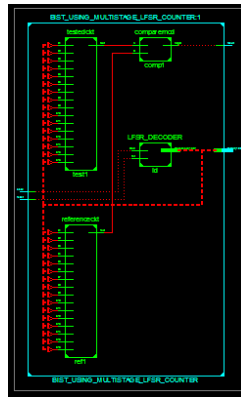


Fig 10: Internal Structure RTL schematic

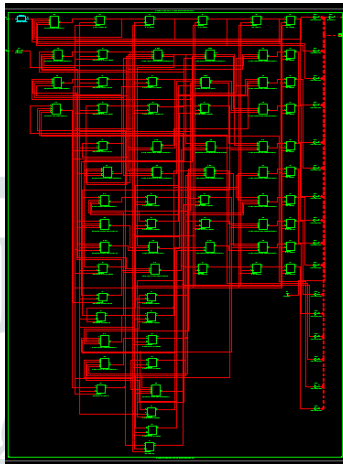


Fig 11: View technology Schematic Of proposed design

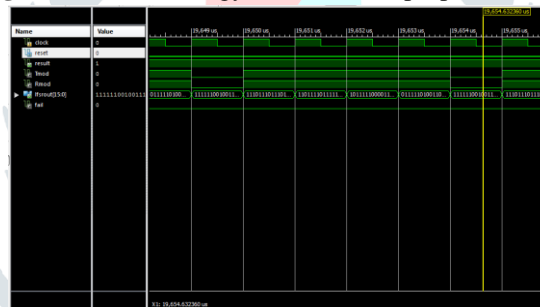


Fig12 : simulated wave forms of proposed design

6. Conclusion

This paper presents a generalized design and a practical implementation in VLSI technology of multistage LFSR counters as well as the decoding logic required to convert the count sequence into binary order. The proposed counter is composed of multiple smaller LFSR stages that are triggered by a specific state transition of the previous stage. This configuration allows the decoding logic to be based on a constant sized LUT for any number of stages, rather than requiring the LUT to scale with the size of the counter. An extension of this paper would be to generalize this multistage counter design to allow the BIST architecture for performing the testing of the circuit with in low cost and less power consumption.

Reference

- [1] D. Bronzi, Y. Zou, F. Villa, S. Tisa, A. Tosi, and F. Zappa, "Automotive three-dimensional vision through a single-photon counting SPAD camera," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 3, pp. 782–795, Mar. 2016.
- [2] I. Vornicu, R. Carmona-Galán, and A. Rodríguez-Vázquez, "A CMOS 0.18 µm 64×64 single photon image sensor with in-pixel 11 b timeto- digital converter," in *Proc. Int. Semiconductor Conf. (CAS)*, 2014, pp. 131–134.
- [3] M. Perenzoni, D. Perenzoni, and D. Stoppa, "A 64×64-pixels digital silicon photomultiplier direct TOF sensor with 100-MPhotons/s/pixel background rejection and imaging/altimeter mode with 0.14% precision up to 6 km for spacecraft navigation and landing," *IEEE J. Solid-State Circuits*, vol. 52, no. 1, pp. 151–160, Jan. 2017.

- [4] J. M. Pavia, M. Scandini, S. Lindner, M. Wolf, and E. Charbon, "A 1×400 backside-illuminated SPAD sensor with 49.7 ps resolution, 30 pJ/sample TDCs fabricated in 3D CMOS technology for near-infrared optical tomography," *IEEE J. Solid-State Circuits*, vol. 50, no. 10, pp. 2406–2418, Oct. 2015.
- [5] C. Niclass, M. Soga, H. Matsubara, M. Ogawa, and M. Kagami, "A 0.18- μm CMOS SoC for a 100-m-range 10-frame/s 200×96 -pixel time-of-flight depth sensor," *IEEE J. Solid-State Circuits*, vol. 49, no. 1, pp. 315–330, Jan. 2014.
- [6] R. Ballabriga, M. Campbell, E. Heijne, X. Llopart, L. Tlustos, and W. Wong, "Medipix3: A 64 k pixel detector readout chip working in single photon counting mode with improved spectrometric performance," *Nucl. Instrum. Methods Phys. Res. A, Accel. Spectrom. Detect. Assoc. Equip.*, vol. 633, pp. S15–S18, May 2011.
- [7] J. Kim, S. Park, M. Hegazy, and S. Lee, "Comparison of a photoncounting- detector and a CMOS flat-panel-detector for a micro-CT," in *Proc. IEEE Nucl. Sci. Symp. Med. Imag. Conf. (NSS/MIC)*, 2013, pp. 1–4.
- [8] F. Villa et al., "SPAD smart pixel for time-of-flight and time-correlated single-photon counting measurements," *IEEE Photon. J.*, vol. 4, no. 3, pp. 795–804, Jun. 2012.
- [9] H. Mo and M. P. Kennedy, "Masked dithering of MASH digital deltasigma modulators with constant inputs using multiple linear feedback shift registers," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 64, no. 6, pp. 1390–1399, Jun. 2017.
- [10] K. J. Sham, S. Bommalingaihnappallya, M. R. Ahmadi, and R. Harjani, "A 3×5 -Gb/s multilane low-power 0.18- μm CMOS pseudorandom bit sequence generator," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 55, no. 5, pp. 432–436, May 2008.

