ISSN: 2278 – 909X
International Journal of Advanced Research in Electronics and Communication Engineering (IJARECE)
Volume 6, Issue 4, April 2017

An Efficient Design for Testability Implementation of Sleep Convention Logic S-Box Based on Scan Cell Design

M.S.Kavitha[1] M.E.VLSI Design, P.Selvakumar[2] M.Tech(Assistant Professor),

***Abstract*— Design for testability (DFT) refers to an added hardware that reduces test generation complexity and test cost, also increases test quality. Sleep Convention Logic (SCL) is an asynchronous logic style which is based on Null Convention Logic (NCL). In the SCL the combinational blocks are made of threshold gates. SCL utilizes power gating method to further reduce the power consumption by incorporating the sleep signal in every single gate. There are currently no DFT methodologies existing for SCL. But in the current NCL, specific DFT methods cannot be directly used due to the sleep mechanism for power gating. The aim of this paper is to implement Dual rail sleep convention logic S-BOX and to analyze the various stuck at faults within the SCL pipeline and also improve the fault coverage. To analyze the power consumption during normal AES S-Box and Dual rail AES S-Box. Hence the project stands for analyzing the stuck at faults and improving the fault coverage by using scan based testing methodology.**

Index Terms— ***Dual Rail,Sleep convention logic, Null convention logic, Design for testability, power gating technique, AES, S-BOX.***

I. INTRODUCTION

Design for testability (DFT) consists of IC design techniques that add testability features to a hardware product design. The tests are generally driven by test programs that execute using automatic test equipment (ATE). The diagnostic information can be used to locate the source of the failure. The automatic test equipment is an instrument used to apply test patterns to device-under-test (DUT), analyze the responses from the DUT, and mark the DUT as good or bad. The DUT is also called as the circuit-under-test (CUT).

Sleep convention logic (SCL) is a self-timed asynchronous pipeline logic style that offers inherent power-gating, resulting in ultra-low power consumption **S**leep convention logic (SCL), is also known as a variant of NULL convention logic (NCL) [1], [2] that takes the advantage of the MTCMOS power-gating technique [3], [4] to further reduce the power consumption. Most of these advantages are the direct result of applying the sleep mechanism to the circuit through high-*V*th transistors. The aim of this paper is to analyze the various stuck-at faults within an SCL pipeline and propose a comprehensive scan-based testing methodology that provides high fault coverage by introducing the scan chain.

Level Sensitive Scan Design (LSSD) is the DFT method used to test the sleep convention logic.

In Cryptography to provide confidentiality and integrity, encryption is used. Hence by using dual rail encoding for data transfer communication the confidentiality is increased. Encryption transforms original information, called plaintext, into transformed information, called cipher text, code text or simply cipher, which usually has the appearance of random, unintelligible data. There are number of cryptography methods which involve different encryption and decryption techniques. One such method is AES [5], called Advance Encryption Standard. It was published by National Institute of Standards and Technology (NIST).

II. SLEEP CONVENTION LOGIC

Overview

SCL is an asynchronous logic style [6] based on the NCL. SCL was originally developed in [7]. SCL combines the idea of the NCL with early completion [8] and fine-grained MTCMOS power-gating [9]. During normal operation, each pipeline stage alternates between set and reset phases. In the set phase, data change from a spacer (called NULL) to a proper codeword (called DATA), and in the reset phase it changes back to NULL. SCL uses delay-insensitive encoded data for data communication. The most popular delay-insensitive encoding is dual rail. .An SCL gate is generally denoted as TH*mnWw*1 . . . , *wn* where *n* is the number of inputs, *m* is the threshold of the gate, and w1, *w*2, . . . , *wn* are the weights of inputs when the weights are > 1.

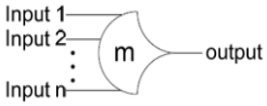

Fig.1 Threshold Gate

A dual-rail encoded signal *D* consists of two wires, *D*0 and *D*1. *D* is logic 1 (DATA1) when *D*1 = 1 and *D*0 = 0, is logic 0 (DATA0) when *D*0 = 1 and D1 = 0, and is NULL when both *D*0 and *D*1 are 0The SCL framework is shown in Fig. 2. Similar to the NCL, each pipeline stage contains a combinational logic function block (*Fi)*, a register block (*Ri)*, and a completion detector block (*CDi)*.

ISSN: 2278 – 909X
International Journal of Advanced Research in Electronics and Communication Engineering (IJARECE)
Volume 6, Issue 4, April 2017

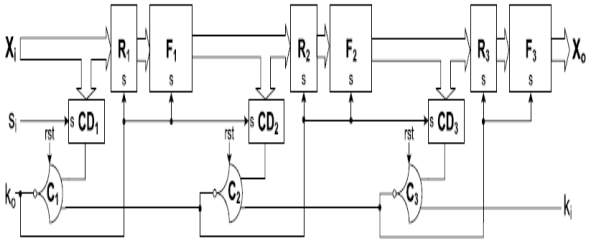

Fig.2 SCL Pipeline

SCL requires an extra gate to synchronize between DATA and NULL phases. This extra gate is a simple resettable *C*-element with inverted output, which will be called the completion *C*-element (*Ci)* hereafter. SCL utilizes fine-grained power-gating by incorporating a sleep signal, *S*, in every single gate. Similar to the NCL gates [10], each SCL gate is made of a *set* block and a *hold0* block (denoted as *set).* In the SCL circuits, however, since all the gates within the combinational blocks are forced to reset by asserting the sleep signal, input-completeness with respect to NULL is inherently ensured and NULL wave front propagation is no longer needed.

Endowments

SCL circuits have several advantages over traditional NCL circuits. These advantages are the direct result of applying the sleep mechanism to the circuit. Since the NULL phase is now forced through the sleep signal rather than waiting for the NULL wave front to propagate through the circuit, the gates no longer need hysteresis, because input completeness with respect to NULL is inherently ensured by explicitly sleeping all the gates. Removing hysteresis from the NCL gates results in a significant amount of area saving. As a result, no extra logic is required to be added to a combinational block to make it input complete with respect to the DATA. Finally, observability in the SCL circuits is also ensured via the sleep mechanism since any potential orphan is explicitly cleared between two adjacent data phases by asserting the sleep signal. In summary, the following contributions are made.

1) Stuck-at faults within various components in the SCL pipeline and how they impact the pipeline are analyzed.

2) A comprehensive scan-based DFT methodology is proposed based on the fault analysis.

Organization

The remaining part of this paper is organized as follows. The related DFT work for the dual rail SCL is discussed in Section III. Section IV analyzes various stuck-at faults within the SCL AES S-Box. Then a scan-based DFT methodology is proposed. The proposed methodology is eventually validated by applying various testing metrics and the experimental results are shown in section V. Finally, the conclusions are drawn in Section VI. Power comparison for the normal s-box operation and the dual rail sleep convention logic based S-box are made it in the table.

III. RELATED DFT WORK

Related SCL (Dual Rail) Specific DFT Methodology

Current ATPG tools do not support asynchronous circuit styles such as the NCL due to asynchronous feedback paths and absence of a clock signal. There are mainly two approaches in the literature to make the NCL circuits testable: limited insertion of control/observation points to increase fault coverage and synchronous modeling of NCL pipelines to make them compatible with synchronous ATPG tools and using scan chain technique.

Normal Boolean circuit is reconfigured as a dual rail logic for data communication.SCL pipeline with two primary inputs (*A* and *B*) and two primary outputs (*Y* and *Z*) are shown in Fig. 8, when registers are replaced with scan cells.

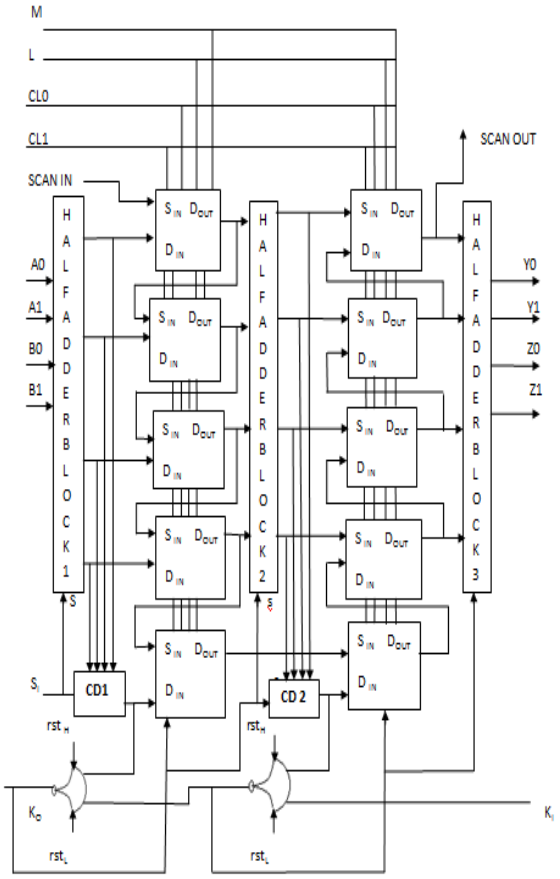

Fig.3 SCL Scan Chain Design

Single scan cell in a scan chain will consist of D-flipflop to form shift register.Similar to a traditional scan chain design, LSSD scan design[11],[12] is used to test the SCL circuit, hence the scan cells form a long shift register in test mode so that the test vectors can be shifted in, and the captured results can be shifted out. The primary output from the scan chain is fed through the SCL pipeline and the final output is obtained and tested by using the true response analyzer. The fault coverage will be improved by using the fault injection techniques.

ISSN: 2278 – 909X
International Journal of Advanced Research in Electronics and Communication Engineering (IJARECE)
Volume 6, Issue 4, April 2017

IV. PROPOSED DESIGN FOR TESTABILITY METHODOLOGY

As discussed before, each stage of the SCL pipeline is made of four separate blocks: combinational logic function (*Fi*), completion detector (CD*i)*, register (*Ri)*, and completion *C*-element (*Ci).* Since the stuck-at faults in each block can impact the SCL pipeline in different ways, each block should be analyzed separately. For the analysis in Section IV-A, it is assumed that the sleep signals are fault-free. The effect of stuck-at faults on sleep signals is analyzed later in Sec. IV-B.

DUAL RAIL AES S-BOX

Dual-rail method is the most promising asynchronous logic style. The benefit of dual-rail logic is that the constant power consumption can be achieved since the signals are implemented by two complementary wires. The power dissipated is independent of the input data in asynchronous logic. This article is based on [13], we propose an asynchronous AES S-Box based on a sleep Convention Logic (SCL), which matches the two important properties mentioned above; dual-rail encoding and clock-free operation. The AES S-Box [14] is constructed by combining the inverse function with an invertible affine transformation in order to avoid attacks based on mathematics. The S-Box [15] is one of the most critical implementation of AES hardware. It consumes the majority of power and is also most vulnerable component to SCAs. A block diagram of the AES S-Box is shown in Fig. 1(a). Asynchronous clock less circuit require less power, generate less noise and produce less electro-magnetic interference compared to their synchronous counterparts. Sleep Convention Logic (SCL) is a delay-insensitive logic which belongs to the asynchronous circuit's categories. SCL circuit utilizes dual-rail and quad-rail logic to achieve this delay insensitivity.

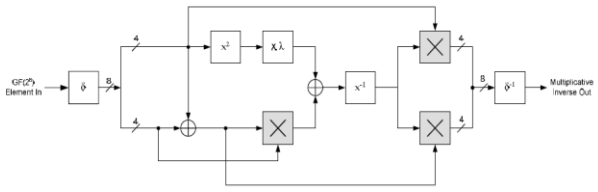

Fig.4 Multiplicative Inverse Module For S Box

A dual-rail signal can Represent [one of available three states, DATA0, DATA1 and NULL, which corresponds to Boolean logic 0 (i.e., DATA0), Boolean logic 1(i.e., DATA1) and control signal NULL for asynchronous handshaking, respectively.

SCAN CHAIN DESIGN FOR S-BOX

This DFT technique is used mainly for testable synchronous circuit. In this scan chain design we assume the use of D-flip-flops only.A mux is placed at the input of each flip-flop in such a way that all flip-flops can be connected in a shift register for one mux selection and to work in a normal mode in the other Connect the SFF in a shift register and test the combinational part.

Level sensitive means that state changes in FSM are independent of delays nor order of changes in input signals (if inputs are set to new values).Scan is defined as an ability to shift into or out of any state.The advantages of scan design are :compatible with multiple clock designs , Shorten test application time,Simplify the stitching of the flip-flops.

The primary inputs are applied to the Dual rail s-box circuit and the corresponding sub byte transformation will be performed to provide the AES encryption output.here the dual rail logic will complicate the configuration of s-box circuit.hence the security will be provided because hacking is complicated in this method more over the testing time will be reduced and the fault coverage is increased in this DFT .

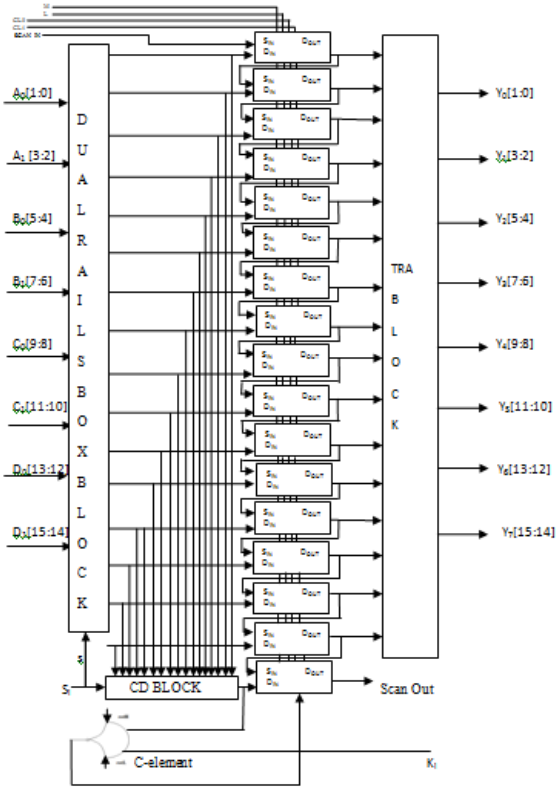

Fig.5 Proposed Scan chain Design For S Box

A.FAULT ANALYSIS

- *Faults on Completion C-Element Signals:*

A *C*-element works as follows: the output is asserted when all inputs are asserted; the output remains asserted until all inputs are deasserted (hysteresis behavior). In the SCL, however, this *C*-element's output is inverted. Since in each DATA/NULL phase all inputs and consequently the output of a completion *C*-element must make a transition for the corresponding DATA/NULL set to propagate through the pipeline. In the SCL pipeline, all stuck-at faults on the inputs and output of all completion *C*-elements can be detected by allowing a single {DATA, NULL} pair to propagate through

the pipeline from primary inputs to primary outputs. Therefore, a complete propagation of a {DATA, NULL} pair ensures that there is no stuck-at-0 or stuck-at-1 fault on the inputs and output of completion *C*-elements.

- *Faults in Completion Detector:*

The stuck-at faults in a completion detector may not necessarily result in a pipeline stall. In NULL phase, even when a gate's output is stuck-at-1, the completion detector will still produce a 0 at its output once the sleep signal is asserted, as long as the output of the last gate in the completion detector is not stuck-at-1. This is in fact a consequence of using the sleep signal to force the completion detector to get cleared rather than requiring the propagation of a NULL wave front to clear it. Note that if the output of the last gate in the completion detector is stuck-at-1, the pipeline will stall after a while. Stuck-at-0 faults always result in a deadlock, so detecting them is easy. This is due to the fact that all gates within the completion detector must be asserted in the DATA phase to assert the output of the completion detector. Therefore, if even a single transition does not happen due to a stuck-at-0 fault, the output of the completion detector cannot be asserted, which eventually results in deadlock, since the output of the completion detector is an input of the completion *C*-element. Hence the faults are analyzed by propagating DATA<NULL pair through the pipeline.

- *Faults in Combinational Logic:*

Combinational logic blocks in SCL are unate. In the DATA phase, gates within a combinational block can only make low-to-high transitions; and in the NULL phase, they can only make high-to-low transitions. This might imply that an approach similar to [16] can be used to detect stuck-at faults in the combinational logic blocks as discussed earlier; unfortunately, this is not possible for two reasons. The first is that the SCL combinational logic is not input-complete; so, in contrast to the NCL, a stuck-at-0 fault on a signal may not necessarily stop it from producing a valid output DATA set, and hence the pipeline may not stall. The second reason is that a stuck-at-1 fault on the output of a gate may be hidden by the gates at its fan-out if those gates can be properly put to sleep. Each combinational logic block in the SCL pipeline behaves exactly like a traditional Boolean combinational logic block when its sleep signal is disabled. Therefore, traditional synchronous combinational ATPG techniques can be used to detect its stuck-at faults.

- *Faults in Register:*

The test patterns generated by traditional ATPG tools are applied to each combinational block through a scan chain design similar to a synchronous approach. This implies that the SCL registers must be augmented to have functionalities similar to a traditional scan cell.

B.SLEEP SIGNAL FAULT ANALYSIS

The analysis performed in Section IV-A was based on the assumption that the sleep signals are fault-free. But in reality, the sleep signals are also prone to stuck-at faults. In this section, the effects of stuck-at faults on sleep signals are analyzed. In the SCL pipeline, as shown in Fig. 1, each sleep signal generated by the output of a completion *C*-element is forked to a register block, a combinational logic block, a completion detector block, and the subsequent completion *C*-element.

- *Sleep Signal Fork to Registers:*

A sleep signal that forks to a register block can be either stuck-at-0 or stuck-at-1. In the case of a stuck-at-1 fault, the register outputs remain low, causing the register to output NULL at all times. This can be easily detected since no DATA set can then propagate through the pipeline, causing the pipeline to stall. In the case of a stuck-at-0 fault on a sleep signal, the register outputs will never return to NULL once they are set to DATA. When the outputs of registers do not get properly reset by the sleep signal, it will cause the registers to output an illegal value. If the new DATA set generates a different output than the previous DATA set then the propagation of illegal values through the pipeline can then be interpreted as a sign of a stuck-at-0 fault on the sleep signal.

- *Sleep Signal Fork to Completion Detectors:*

The sleep signal forks to a completion detector are automatically tested for stuck-at-1 faults at the time of testing the completion *C*-elements and are untestable for stuck-at-0 faults due to redundancy.

- *Sleep Signal Fork to Combinational Logic Blocks:*

The stuck-at faults on the sleep signal fork within combinational blocks are either untestable or it can be ignored due to fault collapsing.

- *Fault Analysis Summary:*

By allowing a single {DATA, NULL} pair to propagate through the SCL pipeline, all stuck-at faults on the inputs and output of all completion *C*-elements can be detected.

By disabling the sleep signal, the SCL combinational logic block becomes a normal Boolean circuit that can then be checked for stuck-at faults using the traditional combinational ATPG tools.

The stuck-at faults on the sleep signal forks within a combinational logic block are either untestable (stuck-at-0 faults) or can be ignored through fault collapsing (stuck-at-1 faults).

The stuck-at faults on the sleep signal forks within a completion detector block are either untestable (stuck-at-0 faults) or can be detected during the test of the completion *C*-elements (stuck-at-1 faults).

The stuck-at faults on the sleep signal forks within a register block are best tested through a scan chain design to be discussed.

C.TEST PROCEDURE

After analyzing different fault scenarios and how they impact the SCL pipeline, we can now devise a methodology to perform testing.

- *Replacing Registers With Scan Cells:*

Similar to a synchronous scan-based testing [17] approach, the SCL registers need to be replaced with scan cells in order to shift in the test. Fig. 6 shows the interface of our proposed SCL scan cell. In dual-rail encoding, each register bit is made of two scan cells, one for each rail. *D*in is

ISSN: 2278 – 909X
International Journal of Advanced Research in Electronics and Communication Engineering (IJARECE)
Volume 6, Issue 4, April 2017

the main input, which could be either rail of a dual-rail input signal. *S*in is the scan input, and *D*out is the output of the scan cell. In a scan chain configuration, *D*out of each scan cell is connected to *S*in of the next scan cell. *M* is the test mode selection signal. When *M* = 0, the scan cell is in normal mode; but when *M* = 1, the scan cell enters test mode.

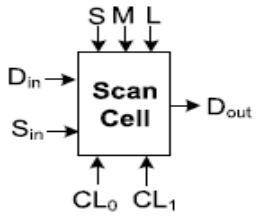

Fig.6 Scan Cell

In normal mode, the scan cell operates exactly like the SCL register; but in test mode, it behaves like a traditional LSSD-type scan cell, where data can be shifted from *S*in to *D*out through the non overlapping clock signals *CL*0 and *CL*1.

In test mode, once the test patterns are applied to a combinational logic block and a sufficient amount of time has passed, the outputs of the combinational logic block can be loaded into scan cells using signal *L*. Finally, *S* is the sleep signal that puts the register in sleep mode when the scan cell is in normal mode.

Fig. 7 shows our proposed implementation of the SCL scan cell. The design is made of two D-latches, one of them being the original SCL register, as shown in Fig. 3, which is reconfigured by signals *M* and *S* to become a D latch.

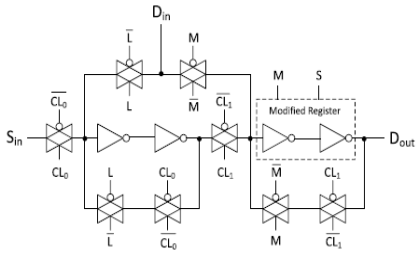

Fig.7 SCL Scancell Design

Performing the Test:

The testing procedure [18] starts with testing the scan cells first. Similar to a synchronous scan chain, a shift test can be initially used to detect most stuck-at faults associated with scan cells and ensure the correctness of the shifting operation. For the shift test, the circuit is first placed in test mode by asserting signals *M* and rst_L.This will disable the sleep signals and set the scan cells to a shift register configuration.

A toggle sequence 00110011…, of length *N* + 4 is then shifted in and out, where *N* is the number of scan cells. The toggle sequence generates various transitions on *S*in and *D*out signals to capture most of the faults associated with the scan cells. The shift test also detects stuck-at-1 faults on the sleep signal forks within the register blocks. Since every bit of the toggle sequence needs to pass all the scan cells, a stuck-at-1 fault on the sleep signal of even a single scan cell causes all the 1s in the toggle sequence to change to 0; therefore, the output sequence will be all 0s.

A similar approach can be used to detect all the stuck-at-0 faults on the sleep signal forks within the register blocks. This time an all-1 sequence, 1111…, is shifted into the scan chain. Then, a signal rst_H is asserted temporarily followed by asserting rst_L again. Asserting rst_H causes the sleep signal of all the scan cells to be asserted, and asserting rst_L returns the circuit to test mode. If there are no stuck-at-0 faults on the sleep signal forks, all the registers must then get cleared, the output sequence to be all 0s. The presence of even a single 1 in the output sequence indicates the existence of a stuck-at-0 fault on a sleep signal fork. In fact, the number of 1s in the output sequence shows how many of the sleep signal forks are stuck-at-0. The second testing step is to apply a single {DATA, NULL} pair to the SCL pipeline in normal mode and it propagate from primary inputs to primary outputs. This will detect all the stuck-at faults on the inputs and output of completion *C*-elements. Additionally, as discussed earlier, this also detects all stuck-at-0 faults on the output of all gates within the completion detector blocks. To detect stuck-at faults in the combinational logic blocks and stuck-at-1 faults on the output of gates within the completion detector blocks remain to be tested. By disabling the sleep signal, the combinational logic blocks become normal Boolean circuits. Therefore, the traditional ATPG tools can be used to generate test patterns to detect the remaining faults.

V. EXPERIMENTAL RESULTS

Proposed SCL S-Box Output:

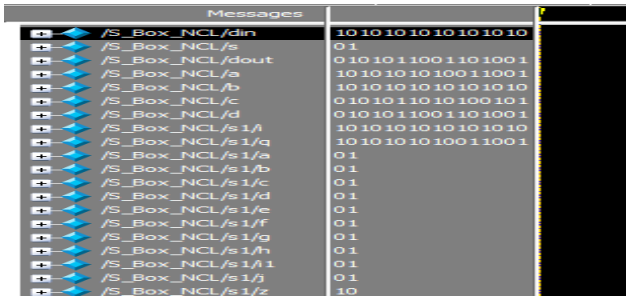

Fig.8 Dual Rail S Box

Test Response Analyzer Output For Dual Rail S Box:

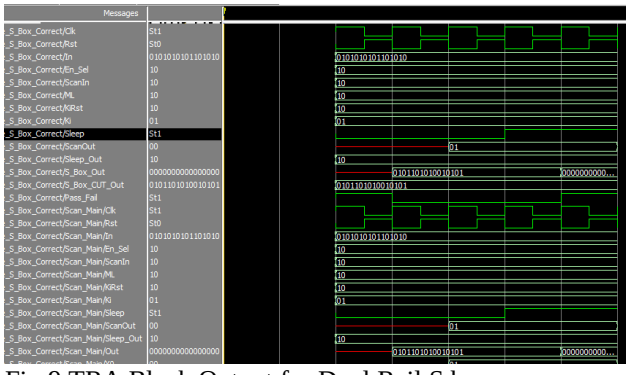

Fig.9 TRA Block Output for Dual Rail S box

POWER COMPARISON

Method name	Input power (mw)	logic power (mw)	Signals (mw)	Output power (mw)
Power consumption during Normal S-Box	32	104	95	11163
Power consumption during Dual rail S-Box	9	25	28	60

VI. CONCLUSION

In this project we can implemented SCL based AES S-box technique successfully by using verilog language. The problem of testing SCL circuits for stuck-at faults was investigated. The faults were initially divided into two separate categories:

1) Faults on logic gates and
2) Faults on sleep signal forks.

The faults within each category were then analyzed separately, and the impact of the faults was discussed. Finally, the proposed DFT methodology was validated through experimental results. By using the fault injection techniques the fault coverage will be improved.

REFERENCES

[1]. J. Di and S. C. Smith, "Ultra-low power multi-threshold asynchronous circuit design," U.S. Patent 8 664 977, Mar. 4, 2014.

[2]. S. C. Smith and J. Di, "Designing asynchronous circuits using NCL," *Synth. Lect. Digit. Circuits Syst.*, vol. 4, no. 1, pp. 1–96, 2009.

[3]. S. Mutoh, T. Douseki, Y. Matsuya, T. Aoki, S. Shigematsu, and J. Yamada, "1-V power supply high-speed digital circuit technology with Multi Threshold-voltage CMOS," *IEEE J. Solid-State Circuits*, vol. 30, no. 8, pp. 847–854, Aug. 1995.

[4] P. Lakshmikanthan, K. Sahni, and A. Nunez, "Design of ultra-low power combinational standard library cells using a novel leakage reduction methodology," in *Proc. IEEE Int. SOC Conf.*, Sep. 2006, pp. 93–94.

[5].Fedral information processing standard publication 197,"Advanced Encryption Standard"nov.26, 2001.

[6].P. A. Beerel, R. O. Ozdag, and M. Ferretti, *A Designer's Guide to Asynchronous VLSI*. Cambridge, U.K.: Cambridge Univ. Press, 2010.

[7]. A. Bailey, A. Al Zahrani, G. Fu, J. Di, and S. C. Smith, "Multi-threshold asynchronous circuit design for ultra-low power," *J. Low Power Electron.*, vol. 4, no. 3, pp. 337–348, 2008.

[8].S. C. Smith, "Speedup of self-timed digital systems using early completion," in *Proc. IEEE Comput. Soc. Annu. Symp. VLSI*, Apr. 2002, pp. 98–104.

[9].G. E. Sobelman and K. Fant, "CMOS circuit design of threshold gates with hysteresis," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, vol. 2. May/Jun. 1998, pp. 61–64.

[10].F. A. Parsan and S. C. Smith, "CMOS implementation comparison of NCL gates," in *Proc. IEEE 55th Int. Midwest Symp. Circuits Syst. (MWSCAS)*, Aug. 2012, pp. 394–397.

[11]. Farhad A. Parsan, S.C. Smith, W.K. Al-Assadi, "Design for testability of sleep convention logic in *IEEE trans on VLSI vol. 24, no. 2,pp-743-753.*

[12] .V. Satagopan, B. Bhaskaran, W. K. Al-Assadi, S. C. Smith, and S. Kakarla, "DFT techniques and automation for asynchronous NULL conventional logic circuits," *IEEE Trans. Very Large Scale Integr. (VLSI)Syst.*, vol. 15, no. 1 0, pp. 1155–1159, Oct. 07.

[13].D.Swathi, Mr.P.Ganagathar, V.Supriya "Designing Of S Box Based On Null Cnvention Logic" in IRJET *vol.02,issue. 08* ,Nov. 2015.

[14].Xinmaio Zhang,Keshab K Parhi"High speed VLSI architecture for the AES algorithm" *IEEE Trans. Very Large Scale Integr. (VLSI)Syst., vol.12,issue.09,* sep. 2004

[15].Edwin NC Mui,Texco Enterprice Pvt Ltd "Practical implementation of rijndael sox using combinational logic"2006.

[16].A. Kondratyev, L. Sorensen, and A. Streich, "Testing of asynchronous designs by 'inappropriate' means Synchronous approach," in *Proc. 8thInt. Symp. Asynchron. Circuits Syst.*, Apr. 2002, pp. 171–180.

[17].W. K. Al-Assadi and S. Kakarla, "Design for test of asynchronous NULL convention logic (NCL) circuits," *J. Electron. Test.*, vol. 25, no. 1, pp. 117–126, 2009.

[18]M.Bushnell And V.D. Agrawal, "Essentials Of Electronic Testing for Digital ,Memory and Mixed Signal VLSI Circuits" ,Vol.17,NY,USA:
Springer-Verlag 2000