Logic Built In Self-Test Verification Statergy For SerDes PHY

[1]Adarsh Malagi , [2]Kariyappa B.S
RV College of Engineering Bengaluru, India
[1]adarshmalagi@gmail.com , [2]kariyappabs@rvce.edu.in

Abstract: VLSI technology has improved rapidly in the past decade. To meet high operating speed, physical layer (PHY) of the system should support high speed communication protocols which made PHY complex. PHY includes Serializer and Deserializer (SerDes), which supports BIST and Loopback as self-testability feature. BIST and Loopback reduces verification time and improves verification quality. Verification strategies for BIST is discussed in this paper. To ensure proper working of BIST module two techniques, BIST flow without error injection and with error injection is presented. Verification flow starts with development of verification plan which is done by Vplanner. To cover all coverage point from the verification plan, 24 test cases are written. NCSIM simulator is used to run test-cases and analysis. Then regression is run by Vmanager tool, here in total 49 test cases are run with multiple seeds. For verification of BIST module 100% functional coverage is achieved.

Keywords: Built-in Self-Test, Physical layer, Serializer, Deserializer.

I. INTRODUCTION

Industry need to keep up with growing demands. It is necessary to meet those demands, which provide great challenges for SOC designers. In competing world, it is necessary to keep time to market factor minimal and verification takes more time of VLSI flow. PHY need to meet high performance and high speed requirement. Physical layer is lowest layer of OSI model. PHY deals with physical interfaces, data is sent and received through physical medium.

PHY include sub-layer Physical Coding Sublayer (PCS) and Physical Medium Attachment (PMA).

The Physical Coding Sublayer is top layer of PHY, it provides interface between Media access control layer and Physical Medium Attachment layer. This layer is included as protocol sub-layer in Fast Ethernet and gigabit Ethernet models. PCS is responsible for lane block synchronization, alignment marker managing, encoding/decoding and deskew.

The PMA is lower sublayer of PHY. It provides interface between physical medium and the device [1]. It deals with reception and transmission of the data through physical medium. It is responsible for signal timing, signal encoding, and interface with physical medium like optical fiber, cables, and wires.

To support high speed protocols, SerDes structure is used. SerDes include serializer and deserializer modules, transmitter block deals with serialization and receiver block deals with deserialization [2]. SerDes PHY structure include single ended registers, encoder, parallel to serial shift register (serializer) output differential buffer, input differential buffer, serial to parallel shift register (deserializer) and decoder. Structure of SerDes PHY is as shown in figure 1.

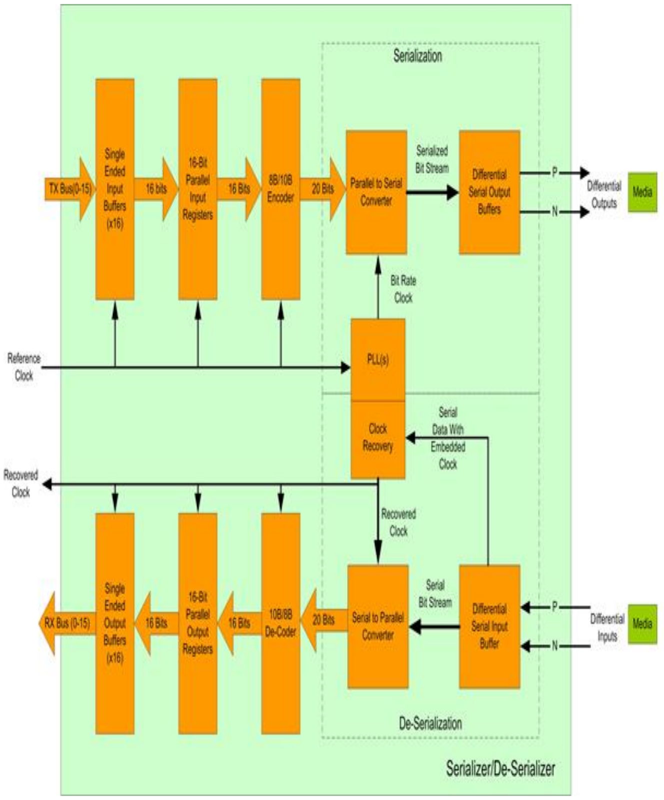


Figure 1 SerDes PHY structure [2]

Verification of PHY is complex process, cannot guarantee complete verification of complex structure. To overcome problems like unreachable corner and cover points, self-test modules are included in design that is included in PHY. BIST and Loopback modules are included to enhance verification. Self-Test modules ease post silicon verification process, reduces verification time and verification cost. With Loopbacks, earlier unreachable corners and cover points are covered as well. Hence BIST and Loopback are necessary for complex designs [3].

In section II the Design is presented along with structure of BIST module in the Design. In section III verification flow of the BIST module is presented with flow chart. In section IV waveforms and regression results are presented. In section V the results are concluded for BIST verification.

doi : 10.17950/ijer/v5s10/1002

II. DESIGN UNDER TEST

Design is the combination of USB 2.0 and USB 3.0. In this paper we concentrate on the verification of USB 3.0 BIST module. USB 2.0 has different modes of operation, they are Low-Speed (1.5 Mbps), Full-Speed (12 Mbps) and Hi-Speed (480 Mbps). Max cable length of the USB 2.0 support is 5meter, supports connection to devices Keyboard, mouse, storage connectivity and bandwidth up to 480Mbps. Feature of USB 3.0 includes upgradation of power management of USB devices. It is available with low speed, high speed and super speed, bandwidth up to 5 Gbps of data transfer, Interrupt driven, Duplex data communication, Link Training and Status State Machine (LTSSM) support.

USB 3.0 PHY has sub-layers PCS and PMA. PMA contain BIST module, Transmitter and receiver block contain separate BIST module with different functionality. BIST module of Transmitter block generates pseudo random binary sequence and that of receiver block checks PRBS sequence. Structure of BIST module in USB 3.0 is as shown in figure 2.

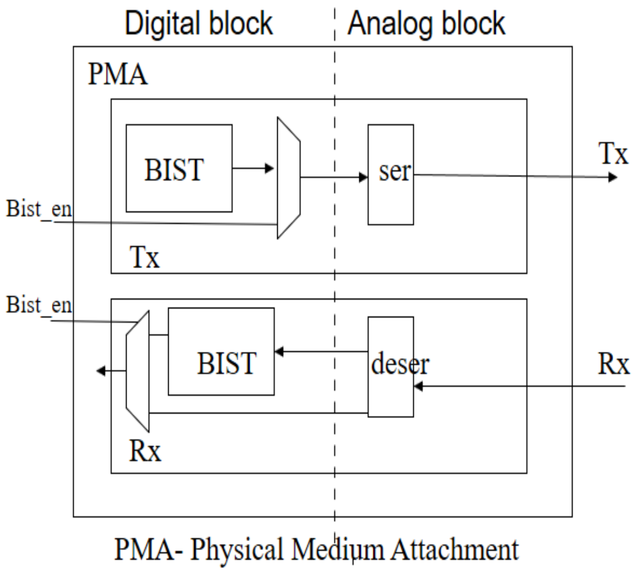


Figure 2 PMA structure including BIST

Transmitter BIST module include PRBS generator and Error injection. PRBS generator generates PRBS sequence and Error injection module can introduce error in PRBS sequence. Receiver BIST module include PRBS sequence checker, error detection and error counter. PRBS checker check received PRBS sequence for any error, if any error gets detected it asserts BIST status signal, and increments Error counter [4].

This design support five modes of BIST, i.e. BIST can operate in PRBS7, PRSB15, PRBS23, PRBS31 and User define mode. To run BIST, both transmitter and receiver BIST modules need to set to same mode. Respective PRBS generator and checker will operate in selected mode. For user defined mode, here FIFO is provided for both BIST modules. User need to write data sequence which need to be transmitted or checked by BIST.

III. VERIFICATION OF BIST MODULE

As discussed in previous section, BIST module support five modes of operation, that is transmitter BIST pattern generator generates random pattern in these PRBS modes and receiver BIST pattern checker supports PRBS modes. Verification need to exercise the design in all modes. If design operates normally in all modes, then verification is said to be complete of BIST module. BIST need to operate in all modes, have to check for supporting modules like error injection, error detection and error counter. So BIST need to run with error injection. There are two test flow, one with BIST run without error injection and another one is BIST flow with error injection. BIST verification is done by connecting transmitter differential pins to receiver differential pins that is external Loopback. Block diagram representation of external Loopback is shown in figure 3.

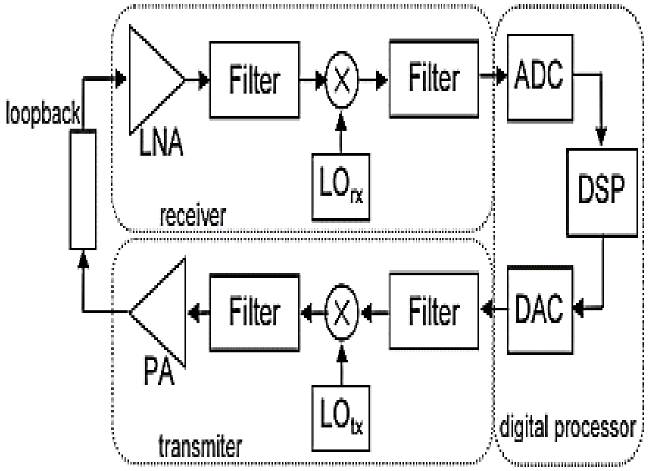


Figure 3 External Loopback

BIST without error injection flow chart is as shown in figure 4.

Here test starts with reset sequence of the design which resets all register values to default values, followed by PLL configuration of Transmitter and Receiver block which makes sure clocks of both block are set to common frequncy. Further BIST flow is started by setting same BIST mode for both Transmitter and receiver BIST. To exercise BIST module the design should be in external loopback as shown in figure 4. Then enable BIST and wait for rx_bist_sync signal which is asserted after receiver BIST synchronies with received PRBS sequence that means BIST module is working as expected. If rx_bist_sync not asserted then exit with failure message. Wait for 10 micro seconds after assertion of rx_bist_sync, then check for rx_bist_status signal which indicates error detection, if asserted exit with failure message else exit with pass message.

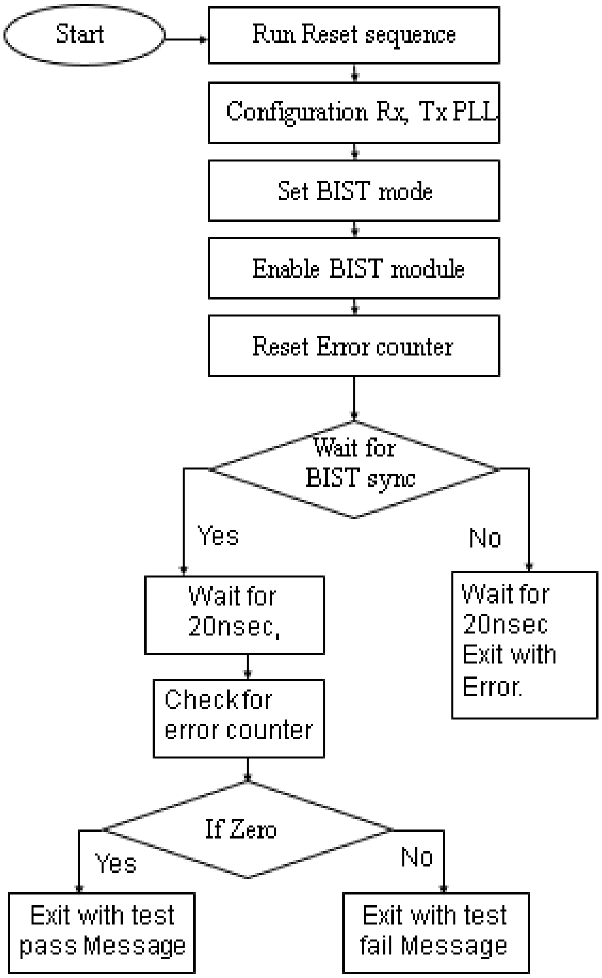


Figure 4 BIST flow without error injection

BIST with error injection flow chart is shown is figure 5. Here above BIST flow that is BIST flow without error injection is followed then finite number of errors are injected and checked at receiver BIST module for error detection. If number of errors detected is same as that of error injected, then test is passed. Transmitter BIST module supports force injection with which finite number of errors are injected into pattern generated by PRBS generator. Receiver BIST module is responsible for error detection and it include error counter to recode number errors detected by BIST module. After error detection receiver BIST module will assert rx_bist_status, in increment error counter. Error counter value is used to check that error injected and error detected values are same.

Common failures observed in both BIST flows are due to faults in setting BIST modes, PLL configuration mismatch or loopback path faults. Error might occur due to any of these reasons. Failure in BIST flow with error injection might be due to fault in error counter as well.

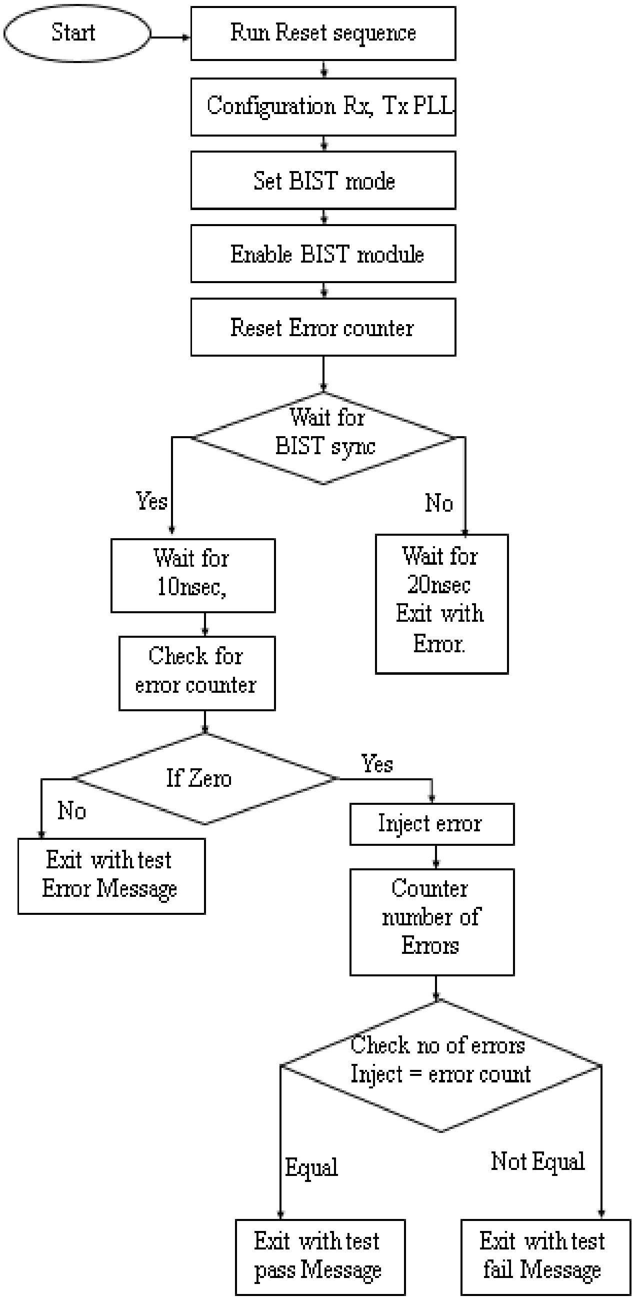


Figure 5 BIST flow with error injection

IV. Results

In this section waveforms for BIST flow without error injection and with error injection is presented. Verification plan and regression results are discussed.

Development of verification plan is first step of verification. Verification of BIST module is said to be complete after verifying it in all scenarios. Verification plan includes tests which exercise BIST module in all modes with both BIST flow which are discussed in previous section. Vplanner tool is used to generate verification plan.

NCSIM simulator is used to run for simulation and analysis. Screenshot in figure 6, shows waveform for BIST flow without error injection. In this waveform rx_bist_status is observed to be low and rx_bist_errcnt is zero throughout which indicates no error has detected. It is also observed that rx_bist_sync asserts after tx_m, tx_p, rx_m and rx_p signals start data transmission. Assertion of rx_bist_sync represents receiver BIST module is synchronized with receiver data sequence. Same BIST mode is set to transmitter and receiver BIST module by register write through Advanced Peripheral Bus (APB) interface. BIST status signals are monitored by register read. That is rx_bist_sync, rx_bist_status and rx_bist_errcnt are read by APB interface. In figure 6 assertion of rx_bist_sync is highlighted by circle.

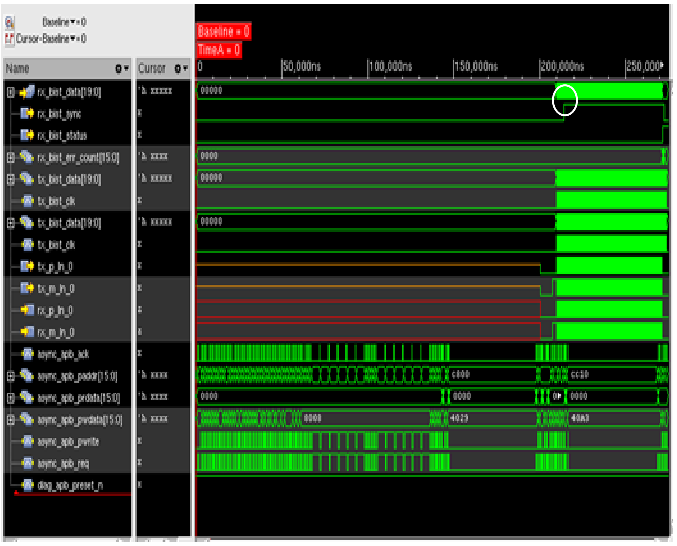


Figure 6 BIST without error injection

Waveform for BIST with error injection is shown in figure 7. Assertion of rx_bist_status represents error detection and rx_bist_errcnt represent number of errors detected. Waveform seems similar to that of BIST flow without error injection only difference is rx_bist_status toggled and rx_bisy_errcnt incremented. Here transmitter BIST module generates PRBS sequence which is feedback to receiver by external loopback. Transmitter BIST module features force error injection using which finite number of error are introduced in PRBS sequence. Receiver detects this errors and rx_bist_status is asserted and error counter is incremented every time receiver BIST module detects error. Error counter is read by particular register read and reset by register write.

In figure 7 rx_bist_status and rx_bist_errcnt are highlighted by circle. In this particular test case 3 errors are injected and 3 errors detected. Test is completed successfully. Failure test cases are analyzed for faults and reported to design team.

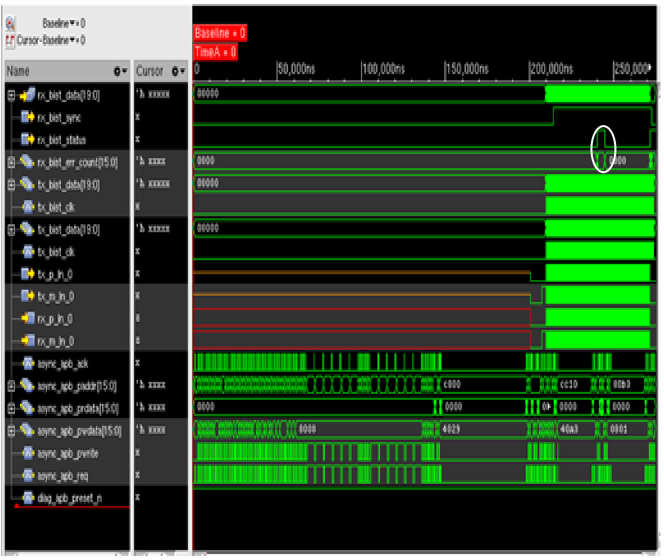
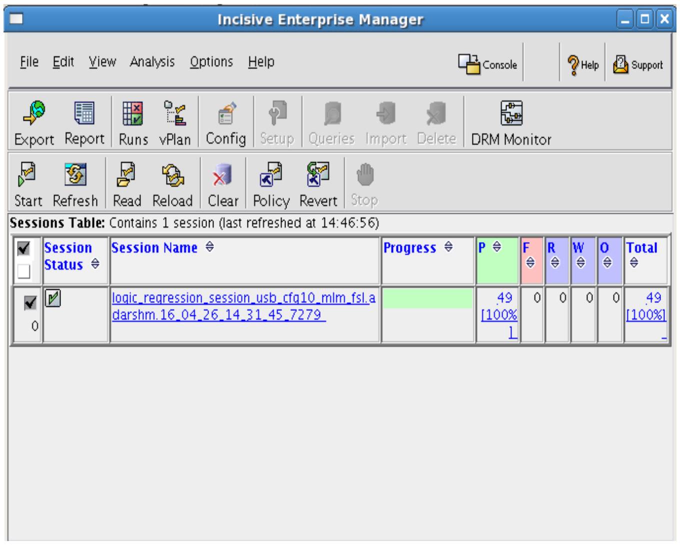


Figure 7 BIST with error injection

These BIST verification technique are implemented for all BIST modes. Test cases are carried out for each BIST mode for both BIST flow. Screenshot shown in figure 8 represents Regression run. All test cases are passed in regression run with total of 49 test cases. Ten test case are written for BIST module verification and run for multiple seeds in regression. Vmanager tool is used to run regression.

Verification document is developed as per verification plan. Which includes coverage items of each test case. Vmanager is used to develop verification document. All test cases are mapped to respective coverage item for all BIST mode. After completion of regression verification document is read and functional coverage analysis is done.

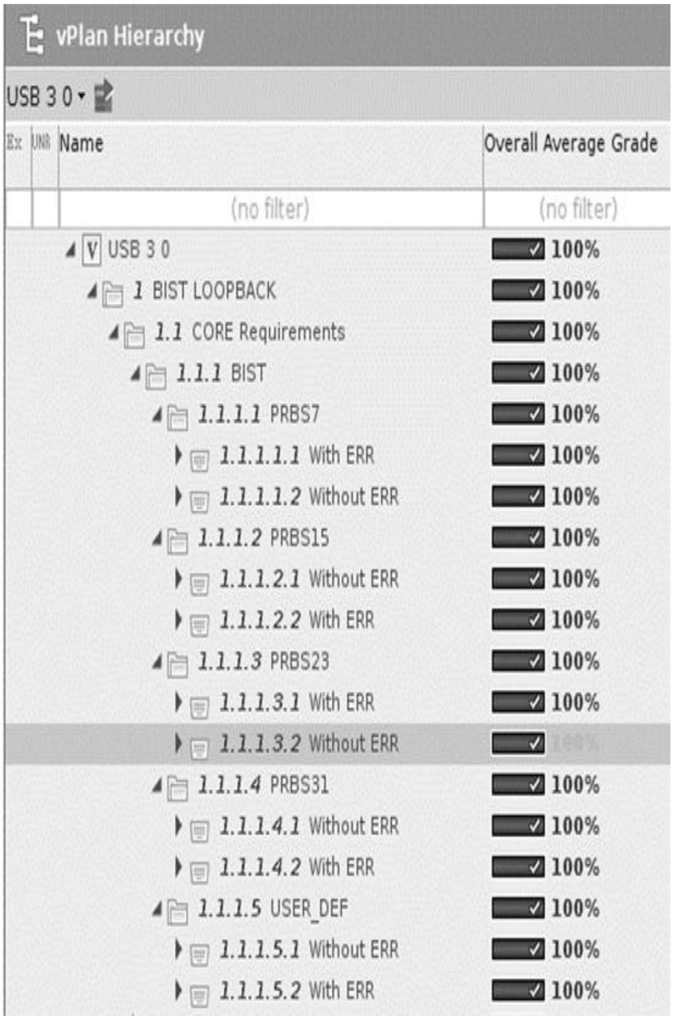


Sessions Table: Contains 1 session (last refreshed at 14:46:56)

☑ □	Session Status ⇕	Session Name ⇕	Progress ⇕	P ⇕	F ⇕	R ⇕	W ⇕	O ⇕	Total ⇕
☑ 0	☑	logic_regression_session_usb_cfg10_mlm_fsl.adarshm.16_04_26_14_31_45_7279	🟩	49 [100%]	0	0	0	0	49 [100%]

Figure 8 Regression Results

Figure 9 shows functional coverage of verification of BIST module. Here 100% functional coverage is achieved. Here BIST includes five groups for every BIST mode. And all group include two targets one BIST with error injection and another without error injection. And all targets are met.



vPlan Hierarchy

USB 3 0 ▾

Ex	UNR	Name	Overall Average Grade
		(no filter)	(no filter)
		◢ V USB 3 0	✓ 100%
		◢ 1 BIST LOOPBACK	✓ 100%
		◢ 1.1 CORE Requirements	✓ 100%
		◢ 1.1.1 BIST	✓ 100%
		◢ 1.1.1.1 PRBS7	✓ 100%
		▸ 1.1.1.1.1 With ERR	✓ 100%
		▸ 1.1.1.1.2 Without ERR	✓ 100%
		◢ 1.1.1.2 PRBS15	✓ 100%
		▸ 1.1.1.2.1 Without ERR	✓ 100%
		▸ 1.1.1.2.2 With ERR	✓ 100%
		◢ 1.1.1.3 PRBS23	✓ 100%
		▸ 1.1.1.3.1 With ERR	✓ 100%
		▸ 1.1.1.3.2 Without ERR	✓ 100%
		◢ 1.1.1.4 PRBS31	✓ 100%
		▸ 1.1.1.4.1 Without ERR	✓ 100%
		▸ 1.1.1.4.2 With ERR	✓ 100%
		◢ 1.1.1.5 USER_DEF	✓ 100%
		▸ 1.1.1.5.1 Without ERR	✓ 100%
		▸ 1.1.1.5.2 With ERR	✓ 100%

Figure 9 Functional coverage analysis

CONCLUSION

Verification plan for BIST module of USB PHY IP design is done successfully. Which include verification strategies for BIST module and loopbacks. Test cases are written according to the verification plan. To exercise BIST module two test flows are developed one without error injection and another with error injection. Functional coverage analysis is done on regression run using Vmanager by loading verification document and 100% functional coverage is achieved.

REFERENCES

i. *Kiran George and Chien-In Henry Chen, "Logic Built-In Self-Test for Core-Based Designs on System-on-a-Chip", IEEE Conference, Pages 1503 - 1508, 2008.*

ii. *Zhang Wen, HaiQi Liu, Qiang Li, "BIST-BASED DIGITAL DESIGN FOR IEEE 1394B PHY", IEEE Conference, Pages 1 - 3, 2014.*

iii. *Jerzy Dabrowski, "Loopback BiST for RF Front-Ends in Digital Transceivers", IEEE Conference, Pages 143 - 146, 2003.*

iv. *Jerzy Dabrowski, Javier Gonzalez Bayon "Mixed Loopback BiST for RF Digital Transceivers", IEEE Conference, Pages 220 - 228, 2004.*

v. *Jerzy Dabrowski, "Built-in Loopback Test for IC RF Transceivers", IEEE Journals & Magazines, Pages 933 – 946, 2010.*

vi. *Jeanne Trinko Mechler, Raymond Bulaga, Jon Garlett, "In-system Failure Investigation on 0.18um High Speed Serial Link ASIC using Logic Built-in Self-Test", IEEE Conference, Pages 613 - 616, 2003.*

vii. *Alexander Irion, Gundolf Kiefer, Harald Vranken, Hans-Joachim Wunderlich, "Circuit partitioning for efficient logic BIST synthesis", IEEE Conference, Pages 86 - 91, 2001.*

viii. *G. Hetherington, T. Fryars, N. Tamarapalli, M. Kassab, A. Hassan, J. Rajski, "Logic BIST for large industrial designs: real issues and case studies", IEEE Conference, Pages 358 - 367, 1999.*

ix. *Juin-Ming Lu, Cheng-Wen Wu, "Cost and benefit models for logic and memory BIST", IEEE Conference, Pages 710 - 714, 2000.*

x. *Wu-Tung Cheng, Manish Sharma, Thomas Rinderknecht, Liyang Lai and Chris Hill, "Signature based diagnosis for logic BIST", IEEE Conference, Pages 1 - 9, 2006.*