An Effective Digital Logic Verification Methodology of High Speed Interface IP Using a Configurable AFE Behavioral and Channel Model

Kiyoon Shim, Seungsik Eom, Beomseok Kang, Jaechul Park, Kyubeum Lee, Youngjin Chung, Hyo Gyuem Rhew
Samsung Electronics, 1-1, Samsungjeonja-ro, Hwaseong-si, Gyeonggi-do 18488, Korea
ky12.shim@samsung.com, ss1.eom@samsung.com, bs96.kang@samsung.com, jejepark@samsung.com, kyubeum.lee@samsung.com, cool.jung@samsung.com, hyo.rhew@samsung.com

Abstract – In this work, a verification methodology for the digital logic of the mixed signal high speed interface (HSI) IP is proposed. In the mixed signal HSI IP, since the digital logic is responsible for important functions such as equalizer coefficients estimation and offset calibration (OC) to achieve target bit-error-rate (BER), accurate and effective verification of digital logic requires the characteristics of analog front-end (AFE) components and channel to be included. Previous works relied on behavioral models of the analog components for the digital logic verification. However, lack of configurability of the AFE and channel characteristics leads to repeated extraction of behavioral models for each AFE and channel condition. An AFE behavioral and channel (ABC) model that provides diverse AFE and channel conditions is proposed and a verification methodology based on the ABC model is also demonstrated. The proposed verification methodology effectively generates signals affected by channel loss and AFE through the ABC model and the signals are provided to the digital logic. The model contains the equalization coefficients and offset code values corresponding to each AFE and channel case, improving the reliability of the verification results. Moreover, the proposed methodology simplifies the integration of IP and ABC model. The proposed model and verification methodology allow for the straightforward debugging of digital logic and significant simulation time reduction. We applied the proposed methodology to PCI express IP to verify the digital logic functions such as OC, decision feedback equalizer (DFE), and eye opening monitor (EOM) logic. The results show that 70.3% verification time reduction to achieve 100% of code coverage compared to the existing behavioral model based verification methodology.

I. INTRODUCTION

The increase in the speed of data transmission and channel loss has resulted in digital logics that estimate and compensate for inter-symbol interference (ISI) becoming more significant in high-speed interface (HSI) IP [1]. Additionally, as the process voltage temperature (PVT) variation increases, the offset caused by the mismatch of analog front-end (AFE) device characteristics degrades the bit error rate (BER) [2], necessitating digital logics that calibrate it. Therefore, the functional verification of the logics that compensate ISI and offset arising from various characteristics of AFE and channel is crucial, and various analog behavioral models [3-5] and verification methodologies [6-8] have been developed to achieve this.

While existing behavioral models offer advantages in integrated verification of mixed signal design, they present a significant challenge in digital logic verification since they need repeated extraction of behavioral models reflecting each AFE and channel characteristics and it is challenging to estimate exact magnitude of the ISI and offset immediately.

This paper proposes a verification methodology that integrates configurable AFE Behavioral and Channel (ABC) model composed of the some AFE components and channel that relevant to digital logic, using the data that passed through the ABC model to verify relevant digital logics. The ABC model generates AFE offset and ISI, calculated based on the AFE and channel characteristic parameters which enabled digital logics to experience various ISI and AFE offset reflected data. Furthermore, it enabled straightforward debugging in verification since we have knowledge of both ISI and AFE offset from the ABC model and expected output from digital logics. This methodology achieved the mentioned improvements through simple integration and significantly reduced simulation time, thereby enabling fast verification and coverage closure of relevant digital logics.

The rest of the paper is organized as follows. Section II introduces the ABC model and illustrates the internal structure of ABC model with a pseudo code example. UVM-based testbench architecture that integrated the ABC model, comparing it with the conventional one is demonstrates in Section III. Section IV shows application of proposed methodology details on digital logics verification sequence and the experimental results. Finally, Section V summarizes the proposed methodology and concludes with a discussion on application of the proposed methodology.

II. STRUCTURE OF ABC MODEL USED IN PROPOSED METHODOLOGY

The ABC model is a simplified behavioral model that generates Rx data that reflect the characteristics of AFE and channel, replacing the analog behavioral model. In mixed signal verification, an analog behavioral model requires extensive computation to reflect its complex nature, accounting for a significant portion of the entire simulation time [9-11]. Rather than reflecting the characteristics of an actual analog circuit accurately, the ABC model focuses on modeling built in offset of the AFE components and ISI for the Rx digital logic verification. It implemented the AFE and channel with simplified formula of the AFE offset and Finite Impulse Response (FIR) filter in sytemverilog real number modeling, achieving significant time reduction in generating AFE offset and ISI. Moreover, the ABC model takes an offset of the AFE components and channel tap coefficients as configurable parameters, while conventional behavioral model need repetitive extraction of model for various AFE and channel characteristics. Structure of the ABC model is shown in Figure 1. The ABC model has controllability over parameters such as offset of the AFE components and tap coefficients of channel and CTLE, and all the parameters could be randomized in UVM-based testbench without any limitation of its value. The magnitude of the AFE offset and ISI in the Rx data that passed through the ABC model can be mathematically calculated according to the formula of the ABC model and parameters that set on testbench. Thereby the expected output of digital logics when they get the Rx data passed through the ABC model could be known immediately, which enables fast debugging on digital logic verification.

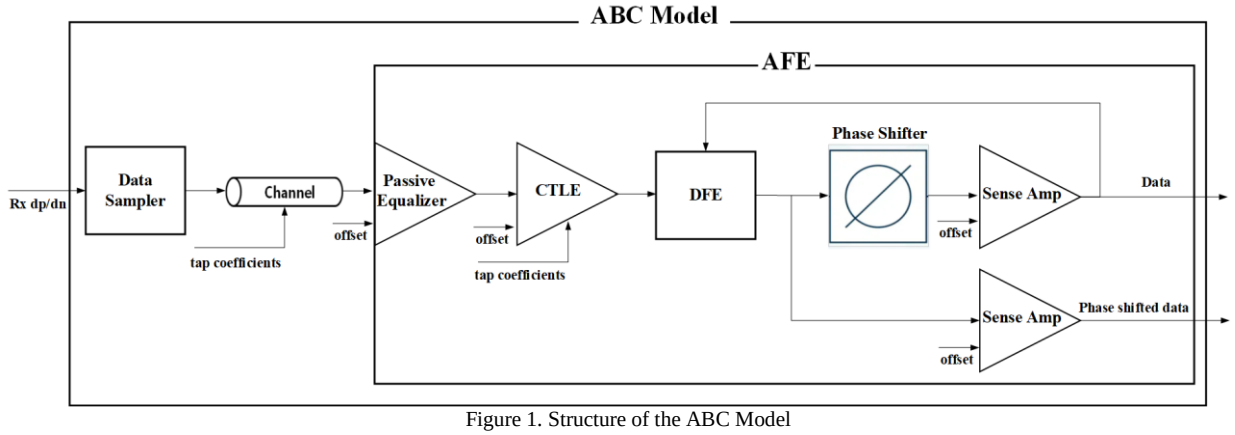


Figure 1. Structure of the ABC Model

The ABC model consists of data sampler, channel, phase shifter and several AFE components. Data sampler samples the data received from serial interface and converts it to real value. The channel is implemented as a FIR filter, which has 10 tap coefficients. The channel convolved the input data with the channel tap coefficients that set as the parameter of the ABC model as shown in Figure 2.

```
for(i = 0; i < ch_tap; i++) begin
   channel_passed_data = tap_coefff[ch_tap - i - 1] * rx_real_data[i] + channel_passed_data;
end
```

Figure 2. Pseudo Code of Channel Model

The AFE includes Passive Equalizer (PEQ), Continuous Time Linear Equalizer (CTLE), Decision Feedback Equalizer (DFE), Phase Shifter and Sense Amplifiers. In the PEQ, CTLE and Sense Amplifiers, built in offset due to the mismatch in transistor characteristics is modeled. Offset is applied to the received data when it passes through the each AFE component as shown in Figure 3.

$$AFE\ passed\ data = input\ data + AFE\ offset - calibrated\ offset$$

Figure 3. Formula of Built in Offset at each AFE Components

CTLE has a high pass filter to compensate Rx data passed through the channel, which is implemented as a 10 tap FIR filter as channel.

DFE subtracts the product of the feedback data and the tap coefficient values found in the sign-sign least-mean-square (SSLMS) logic from the input and processes it to the output.

Phase Shifter assumes that the data between discrete non-oversampled Rx data is linear and creates interpolated data using at the EOM logic. Phase shifter divides 1 Unit Interval (UI) into 'MAX_PHASE' and shifts the phase bidirectionally, requiring both previous data and subsequent data as shown in Figure 4.

```
if (phase_code < 2/MAX_PHASE) begin
   for(i=1;i<width+1;i++) begin
      phase_shift[i] = dfe_passed[i] + (dfe_passed[i+1] - dfe_passed[i])*(phase_code) / MAX_PHASE;
   end
end else begin
   for(i=1;i<width+1;i++) begin
      phase_shift[i] = dfe_passed[i-1] + (dfe_passed[i] - dfe_passed[i-1])*(phase_code)/MAX_PHASE;
   end
end
```

Figure 4. Pseudo Code of Phase Shifter

2 Sense amplifiers receive data from the DFE and phase shifter and make data decision considering the AFE offset, by comparing it with the reference data level.

III. UVM TESTBENCH ARCHITECTURE USED IN PROPOSED METHODOLOGY

Figure 5 shows the UVM-based testbench architecture used in the proposed methodology and the testbench architecture used in the conventional methodology for mixed signal IP verification taken reference from PCIe IP. Figure 5(a) and Figure 5(b) depict the testbench architecture used in the conventional methodology, while Figure 5(c) shows the one used in the proposed methodology.

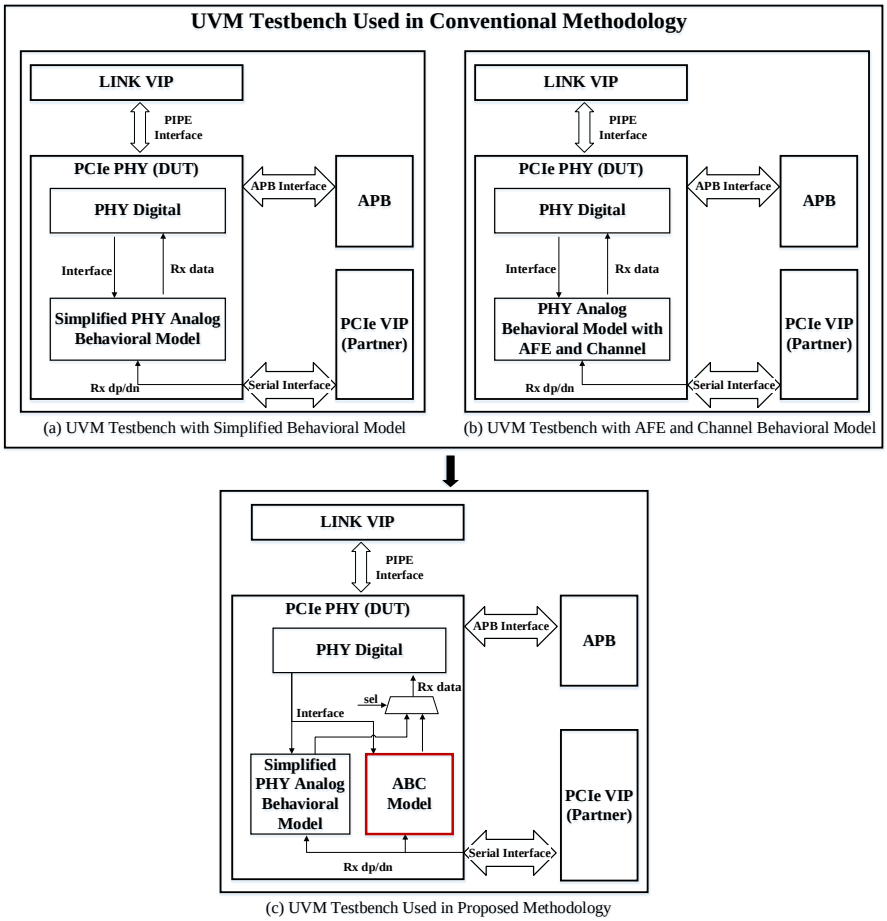


Figure 5. Block Diagrams of UVM Testbench Architecture of PCIe PHY

Analog behavioral model used in Figure 5(a) is a simplified behavioral model that does not include the AFE built in offset and ISI characteristics. It achieves fast simulation but since the characteristics of AFE and channel are lost during the simplification, it processes the Rx data without AFE and channel characteristics, which are necessary for verifying the Rx digital logic.

Analog behavioral model used in Figure 5(b) reflects the characteristics of AFE and channel, thereby processing the data that reflects the AFE offset and ISI. It is effective to verify the real analog circuit. However, it needs repetitive extraction of model to verify it on various AFE and channel characteristics. Additionally, the generation of ISI and the continuous processing of data reflecting it result in a significant simulation time load, making it unsuitable for functional verification where the characteristics of a real analog circuit are not necessary.

In the proposed methodology, the ABC model is multiplexed with a simplified analog behavioral model as shown in Figure 5(c). In this architecture, all test cases except those requiring AFE characteristics and ISI reflections use Rx data from the simplified behavioral model. Test cases that need the Rx data to reflect AFE offset and ISI will select the ABC model branch of the multiplexer. By randomizing the parameter of the ABC model in UVM-based testbench, it enables the creation of various environments of AFE and channel without the repetitive extraction of an analog model in Rx digital logic verification. Moreover, since the AFE offset and ISI are calculated immediately by referring to the parameter set on the testbench, the expected output of the corresponding digital logic can also be known immediately. It ensures fast debugging and high accuracy in verification. Additionally, Rx digital logic using Rx data that reflects AFE offset and ISI can be verified in significantly reduced simulation time with the ABC model.

Table 1 summarizes the characteristics of 3 UVM-based testbench architectures shown in Figure 5. The comparison of average simulation time was conducted with 300 runs of same protocol scenarios which are basic data path, speed change, and power management scenarios including transmission of data packet.

Table 1. Characteristics of UVM Testbench Architecture in Figure 5

	Figure 5(a) : Simplified PHY Behavioral model	**Figure 5(b) : PHY Behavioral model with AFE and channel**	**Figure 5(c) : Simplified PHY Behavioral model + ABC model**
Average simulation time	29.23 mins	77.18 mins	30.79 mins
Rx data reflects the AFE characteristics and ISI	X	O	O
How to implement new AFE and channel characteristics	Not supported	Extract new AFE and channel model	Change parameters of ABC model
How to know magnitude of AFE offset and ISI in behavioral model	Not supported	Additional calculation after extraction	In real time referring parameter of ABC model

The simulation time with Figure 5(b) increased by 164% comparing to Figure 5(a). This demonstrates that the repetitive computation and processing of AFE and channel characteristics in an analog model account for substantial portion in mixed signal design. Characteristics of Figure 5(c) shows that the proposed methodology enabled configuration of AFE and channel, as well as an immediate calculation of its magnitude within a 40% reduced simulation time compared to Figure 5(b), and simulation time increased by 5.3% compared to Figure 5(a) which indicates that load from integrating the ABC model is relatively minor.

General verification flow of Rx digital logics with a UVM-based tesbench used in proposed methodology is shown in Figure 6.

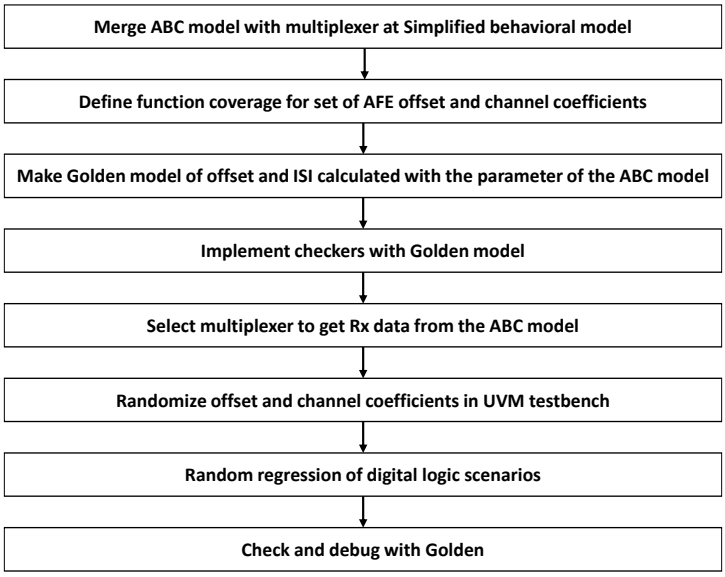


Figure 6. Rx Digital Logics Verification Flow using ABC Model

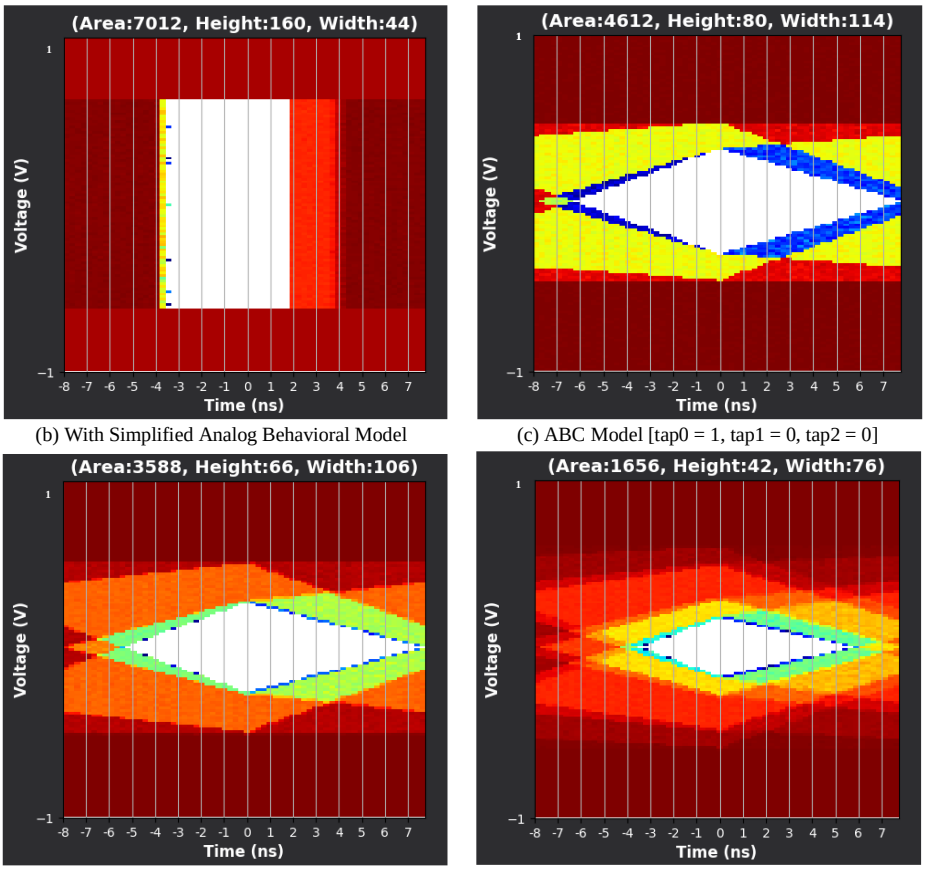
After merging the ABC model into a simplified analog behavioral model, we defined function coverage for the range of AFE offset in the AFE components and channel tap coefficients. Since the ABC model processes the Rx data without the constraint of reflecting the real analog circuit, out of range offset and channel coefficients that were unable to be covered in real analog circuits have been defined for coverage bins as well. Checkers are implemented based on the golden model, calculated based on the parameters of the ABC model. Randomization of parameters in the ABC model is performed in a UVM-based testbench, including ranges that the conventional behavioral model was unable to generate, and proceeds with regression. For error cases, straightforward debugging is enabled based on the golden model with a significantly reduced simulation time compared to conventional analog model.

IV. APPLICATION OF PROPOSED METHODOLOGY AT PCIE IP

In this work, the proposed verification methodology was applied to the verification of 16.0 GT/s PCIe IP. Rx digital logics such as EOM, Offset Calibration (OC), DFE and logics that measure margin of the Rx and Figure of Merit (FOM) were verified using the Rx data sent from the partner Verification IP (VIP) at the UVM testbench shown in Figure 5(c). Application of the ABC model in EOM, OC and DFE logics and their sequences are elaborated below. Figure 7(a) shows the verification sequence of EOM logic. Figure 7(b) is the EYE diagram plotted using EOM logic with a simplified analog behavioral model, representing the voltage level without ISI, and the shape of the EYE is rectangular since the phase is not shifted. In the UVM testbench used in the proposed methodology, EYE diagrams can be plotted under various channel environments by randomizing the tap coefficients in the ABC model as shown in Figure 7(c)~(e). tap0 refers to the main cursor, while tap1 and tap2 refer to the 1st and 2nd post cursor values respectively. Since the Rx data from the ABC model reflected the ISI, the size of the EYE decreased and the shape of the EYE is not rectangular since the phase of the Rx data is shifted in phase shifter. In the proposed methodology, the data level of Rx data that passes through the channel and phase shifter can be determined by referring to the channel tap coefficients. Thereby, after dividing the time domain of 1 UI into '*MAX_TIME*' and the voltage domain into '*MAX_VOLTAGE*', we verified the EOM logic by comparing the EYE diagram plotted by the EOM logic with the calculated error points across entire *'MAX_TIME'*'MAX_VOLTAGE'* points.

Sequence	Verification sequence of EOM logic
1	Select multiplexer to get Rx data from the ABC model
2	Randomize channel tap coefficients
3	Divide time domain of 1 UI into '*MAX_TIME'*, voltage domain into *'MAX_VOLTAGE'*
4	Calculate expected error points based on channel tap coefficients
5	Plot EYE using EOM logic with ABC model passed Rx data
6	Compare error points in EYE diagram with expected error points

(a) Verification Sequence of EOM Logic



(b) With Simplified Analog Behavioral Model

(c) ABC Model [tap0 = 1, tap1 = 0, tap2 = 0]

(d) ABC Model [tap0 = 1, tap1 = 0.1, tap2 = 0]

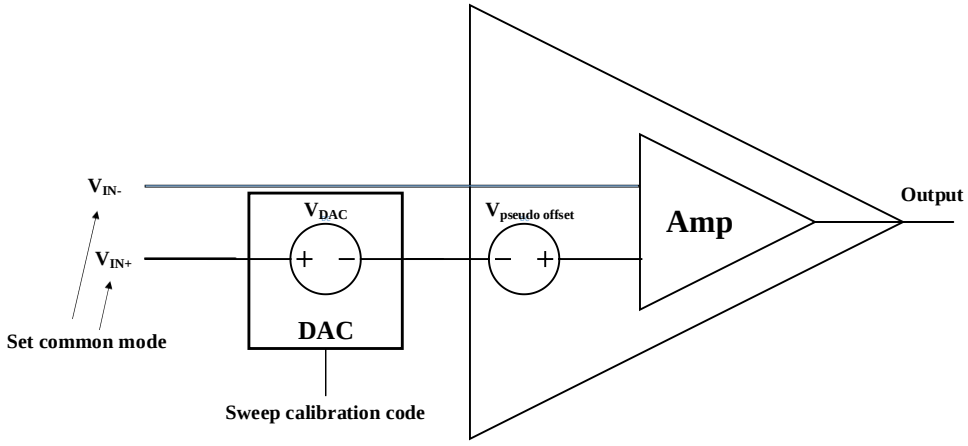
(e) ABC Model [tap0 = 1, tap1 = 0.2, tap2 = 0]

Figure 7. EOM Verification Sequence and Eye Diagrams with Simplified Behavioral Model and ABC Model

Figure 8(a) shows the verification sequence of the OC logic and Figure 8(b) shows the process of calibrating the AFE offset. The ABC model allowed for the modeling of the AFE offset which is challenging to represent it with the actual circuits and unable to calibrate by digital logic. By applying this out of range offset, it enabled the detection of bugs in functions related to calibration failures such as calibration retry, which cannot be detected with an in range offset in the behavioral model that reflects the characteristics of the actual circuit.

Sequence	Verification sequence of OC logic
1	Select multiplexer to get Rx data from the ABC model
2	Randomize pseudo offset in the AFE components
3	Set differential input as common mode
4	Sweep calibration code of OC logic
5	Check the crossing point where the sign of the Amp output changes
6	Check that crossing point is matching with forced pseudo offset

(a) Verification Sequence of OC Logic



Output = Gain * {{V_{In+} - V_{In-}) + ($V_{pseudo\ offset}$ - V_{DAC})

(b) Process of OC

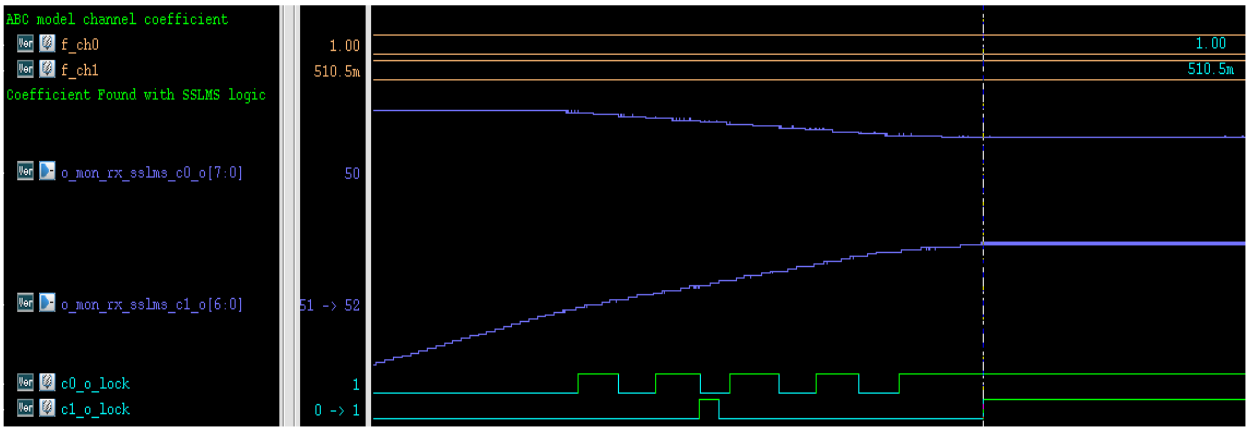
Figure 8. Verification of OC Logic

Figure 9(a) shows the verification sequence of the DFE SSLMS logic. The unit of channel tap coefficient is voltage and for the SSLMS logic, code C0 is converted to 20mV per code while C1 is converted to 10mV per code. By using the SSLMS algorithm, the tap coefficients converge to values that match the channel tap coefficients of the ABC model, as shown in Figure 9(b).

The verification covered various configurations of channel including the configuration which was unable to generate with the conventional behavioral model. The verification covered the cases of over equalization and ISI that cannot be covered by DFE, thereby the coefficients are not converge with the SSLMS algorithm. By verifying the DFE at these unprecedented channel, it enabled the detection of bugs in functions related to adaptation failures such as adaptation timeout which was not detected with the conventional behavioral model.

Sequence	Verification sequence of DFE SSLMS logic
1	Select multiplexer to get Rx data from the ABC model
2	Randomize channel tap coefficients
3	Run DFE and wait SSLMS algorithm to find the tap coefficients
4	Convert converged DFE coefficients into voltage
5	Check DFE coefficients match with tap coefficients of the ABC model

(a) Verification Sequence of DFE SSLMS Logic



(b) Finding Tap Coefficients with the SSLMS Logic

Figure 9. Verification of DFE SSLMS with the ABC Model

The experimental result that shows verification period in Rx digital logics is shown in Table 2.

Table 2. The Result of Verification Period Reduction in Rx Digital Logics

	Verification Period (Hour)		
	Figure 5(b) : PHY Behavioral model with AFE and channel	**Figure 5(c) : Simplified PHY Behavioral model + ABC model**	**Reduction Rate**
Average Simulation time of digital logic scenarios	6.53	2.01	69.2%
Time to achieve 100% code coverage of digital logics	54.26	16.13	70.3%
Time to achieve 100% function coverage of AFE and channel	106.75	33.87	68.3%

A comparison was performed on the simulation time and verification period of Rx digital logics, which required the AFE characteristics and ISI reflected Rx data, between the proposed methodology and the conventional methodology. The proposed methodology used UVM testbench shown in Figure 5(c), while the conventional methodology employed the UVM testbench shown in Figure 5(b), using Xmodel [12] for the analog behavioral model. They were compared using the same regression suite, which included scenarios for EOM, OC, DFE, FOM and logic designed to find the margin of Rx. The coverage bins of AFE characteristic and channel that could not be extracted from the actual circuit were excluded from function coverage in Figure 5(b). As shown in Table 2, the average simulation time for Rx digital logic was reduced by 69.2%. This was calculated based on the average simulation time in regression that achieved 100% code coverage.

V. CONCLUSION

This paper presented a verification methodology for mixed signal IP to both improve verification coverage and reduce overall Turn-around-Time in the verification of digital logic. The proposed methodology introduces a model capable of representing various AFE and off chip channel characteristics and achieved fast DV closure and improvement in digital logic coverage by integrating it on UVM-based testbench. By applying the proposed methodology, the characteristics of AFE and channel can be controlled within a UVM-based testbench and offset of the AFE components and ISI can be calculated immediately. The time required to achieve 100% code coverage for Rx digital logic was reduced by 70.3% using the proposed methodology. Considering the time saved in configuring various AFE and channel environments and reduced debugging time, additional enhancements in the verification period are expected. The conventional methodology could be effective when verification with an AFE and a channel that reflects the real analog circuit is necessary. Otherwise, the proposed methodology can be applied to mixed signal verification, enabling the early bug detection and verification closure of Rx digital logic in the early stage of the project.

REFERENCES

[1] Cristofoli, Andrea, et al, "Efficient statistical simulation of intersymbol interference and jitter in high-speed serial interfaces." IEEE Transactions on Components, Packaging and Manufacturing Technology 4.3 (2013): 480-489.

[2] Bull, David, et al, "A power-efficient 32 bit ARM processor using timing-error detection and correction for transient-error tolerance and adaptation to PVT variation." IEEE Journal of Solid-State Circuits 46.1 (2010): 18-31.

[3] Chuck McClish, “Take AIM! Introducing the Analog Information Model”, DVCON US, 2023.

[4] Mariam Maurice, “Functional Verification of Analog Devices modeled using SV-RNM” DVCON US, 2023.

[5] Doboli, Alex, Ranga Vemuri, "Behavioral modeling for high-level synthesis of analog and mixed-signal systems from VHDL-AMS." IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems 22.11 (2003): 1504-1520.

[6] Charles Dančak, “A UVM SystemVerilog Testbench for Analog/Mixed-Signal Verification: A Digitally-Programmable Analog Filter Example”, DVCON US, 2023.

[7] J. B. David, H. Chang, “UVM Testbench Automation for AMS Designs”, DVCON US, 2023.

[8] Jun Yan, Josh Baylor, “Automated Modeling Testbench Methodology Tested with four Types of PLL Models”, DVCON US, 2023.

[9] Cormac O’Sullivan, Donnacha O’Riordan, “Mixed Signal Design & Verification for Complex SoCs”, 2021 Renesas Electronics.

[10] Simone Fontanesi, “A comparison of methodologies to simulate mixed-signal IC”, DVCON EUROPE, 2021.

[11] Enrico Castaldo, "A Comprehensive Analog–Mixed Signal (AMS) Simulations Environment" 2024. [Online]. Available: https://www.researchgate.net/publication/384177408_A_Comprehensive_Analog-Mixed_Signal_AMS_Simulations_Environment.

[12] Scientific Analog Inc, "xmodel: Empower SystemVerilog with Event-Driven Analog Models," 2023. [Online]. Available: https://www.scianalog.com/xmodel/. [Accessed 13 Nov 2023].