Verilog Modeling of Transmission Line for USB 2.0 High-Speed PHY Interface

Ki-Hwan Seong, Ji-Hoon Lim, Byungsub Kim, Jae-Yoon Sim, and Hong-June Park

***Abstract*—A Verilog model is proposed for transmission lines to perform the all-Verilog simulation of high-speed chip-to-chip interface system, which reduces the simulation time by around 770 times compared to the mixed-mode simulation. The single-pulse response of transmission line in SPICE model is converted into that in Verilog model by converting the full-scale analog signal into an 11-bit digital code after uniform time sampling. The receiver waveform of transmission line is calculated by adding or subtracting the single-pulse response in Verilog model depending on the transmitting digital code values with appropriate time delay. The application of this work to a USB 2.0 high-speed PHY interface reduces the simulation time to less than three minutes with error less than 5% while the mixed-mode simulation takes more than two days for the same circuit.**

***Index Terms*—Verilog, transmission line, USB 2.0 high-speed PHY, mixed-mode simulation**

I. INTRODUCTION

The data rate of chip-to-chip interface is increased over several hundreds of Mbps or several Gbps recently, because of the ever-increasing requirement of data-bandwidth of electronic equipments. For the high-speed chip-to-chip interface, a transmission line is used to connect chips [1]. The circuit simulation is required to send out the designed chip for fabrication. For the circuit simulation the chip-to-chip interface system, the chips are described mostly in Verilog netlist and the transmission line is described in SPICE netlist. Therefore, a mixed-mode simulator such as SPECTRE is used for the simulation of chip-to-chip interface. It takes an excessively long time to perform the mixed-mode simulation of the circuit, which contains both Verilog and SPICE netlist, For example, it takes around two days to finish the mixed-mode simulation of a USB 2.0 high-speed interface for the time interval of 3us with a recent desktop PC. For the simulation of chip-to-chip interface, the MATLAB model [2] or the Verilog-AMS model [3, 4] were used. In the MATLAB model, the simulation time is much shorter than the mixed-mode simulator such as SPECTRE, since both transmission line and chip are modeled in MATLAB. However, because the chip is modeled usually in Verilog, it is inconvenient to convert the Verilog chip model to the MATLAB chip model. In the Verilog-AMS model, the Verilog chip model is used unchanged. However, the simulation time of the Verilog-AMS model is almost the same as the mixed-mode simulator such as SPECTRE, since the transmission line is described by the Verilog-A language in the Verilog-AMS model [5, 6].

In this work, to reduce the simulation time while keeping the Verilog chip model, the SPICE netlist of transmission line is converted into a Verilog netlist. This work takes less than three minutes to perform the above-mentioned simulation of the USB 2.0 high-speed interface. Section II shows the procedure to convert the SPICE netlist of transmission line into a Verilog netlist. Section III presents the simulation results of this work for the USB 2.0 high-speed chip-to-chip interface. Section

Manuscript received Mar. 26, 2014; accepted Jun. 10, 2014
of Electronic and Electrical Engineering, Pohang University of Science and Technology (POSTECH), Gyeongbuk, Korea
E-mail : hjpark@postech.ac.kr

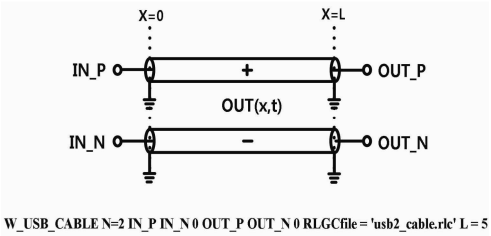


Fig. 1. A lossy differential transmission line.

IV concludes this work.

II. VERILOG OF TRANSMISSION LINE

1. SPICE Model of Transmission Line

A differential transmission line is mostly used for the serial-link chip-to-chip interface, such as USB, SATA, PCIe, MIPI, HDMI and Display Port.

The transmission line has a low-pass characteristic because of the dielectric loss and the skin effect loss. That is, the higher-frequency component is more severely attenuated by transmission line than the lower frequency component.

This phenomenon is modeled by a lossy transmission line model (W model) in SPICE. Fig. 1 shows the SPICE netlist of a differential lossy transmission line, where N=2 refers to the differential line. The 'usb2_cable.rlc' file contains the model parameter values of a lossy transmission line, which are represented by six 2-by-2 matrices shown in Fig. 2. Table 1 shows the values of the lossy transmission line parameters of Fig. 2 for a USB 2.0 cable.

If a differential input voltage of {+0.5exp(jwt), -0.5exp(jwt)} is applied at the differential input nodes at x=0 (IN_P, IN_N) of Fig. 1, the differential voltage OUT(x, t) at a position x and a time t can be represented by

$$OUT(x,t) = e^{jw(t-L\sqrt{L_O C_O})} e^{-\alpha x} \tag{1a}$$

$$\alpha = \frac{R_{skin}}{2\sqrt{L_O C_O}} + \frac{G_{dielectric}}{2\sqrt{L_O C_O}} \tag{1b}$$

where no reflection is assumed because of the RX termination.

The attenuation constant alpha is determined by the skin effect parameter R_{skin} and the dielectric loss

$$L_O = \begin{bmatrix} L_s & L_m \\ L_m & L_s \end{bmatrix} \qquad R_O = \begin{bmatrix} R_{os} & 0 \\ 0 & R_{os} \end{bmatrix}$$

$$C_O = \begin{bmatrix} C_s & C_m \\ C_m & C_s \end{bmatrix} \qquad R_S = \begin{bmatrix} R_{ss} & 0 \\ 0 & R_{ss} \end{bmatrix}$$

$$G_O = \begin{bmatrix} G_{os} & G_{om} \\ G_{om} & G_{os} \end{bmatrix} \qquad G_d = \begin{bmatrix} G_{ds} & G_{dm} \\ G_{dm} & G_{ds} \end{bmatrix}$$

Fig. 2. Model parameter of a lossy differential transmission line.

parameter $G_{dielectric.}$ R_{skin} and $G_{dielectric}$ are represented by the lossy transmission line model parameters(R_o, R_s, G_o, G_d), which are included in the 'usb2_cable.rlc' file.

$$R_{skin} = R_O + R_S\sqrt{f} \tag{2a}$$

$$G_{dielecctric} = G_O + G_d f \tag{2b}$$

2. Conversion of a Single-pulse Response from SPICE to Verilog

In this work, a SPICE single-pulse response PR(t) is calculated by using SPECTRE and then the SPICE single-pulse response is converted into a Verilog single-pulse response D_PR(t). D_PR(t) is used to get the received waveform at RX for any digital data transmitted at TX. To get the SPICE single-pulse response PR(t), the SPECTRE [7] simulation is performed for the SPICE model part enclosed by the dashed line shown in Fig. 3(a), which includes the chip pin parasitic and the transmission lines. The TX driver is assumed to be an ideal CML driver, and the RX front-end consists of an ideal differential comparator. For this, a single current pulse is applied to IINP(t) of Fig. 3(a) while IINN(t) remains at zero, as shown in Fig. 4. Then, the RX input waveform VRX(t) from SPECTRE simulation is converted into a SPICE single-pulse response PR(t) by eliminating the transmission line delay (time of flight tF), as shown in Fig. 5. T is an input data period. That is,

$$PR(t) = V_{RX}(t + t_F) \tag{3}$$

While the non-zero response starts from t = tF in

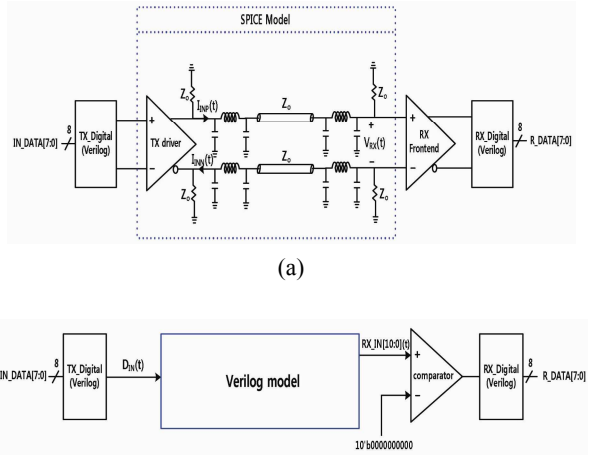


Fig. 3. (a) SPICE model, (b) Verilog model of transmission line.

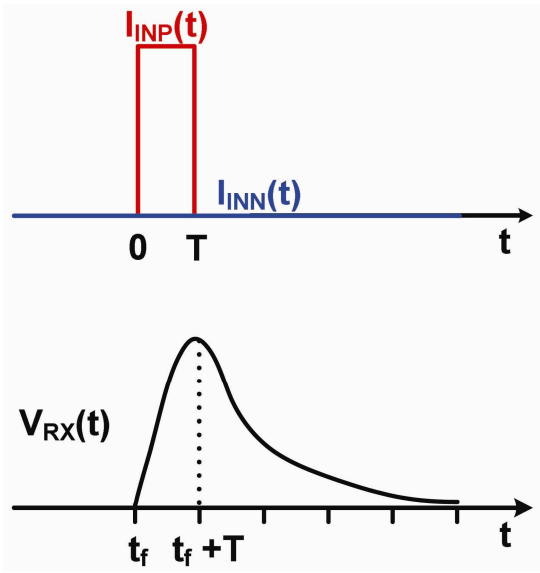


Fig. 4. Input and output waveforms of the SPICE model (Fig. 3(a)).

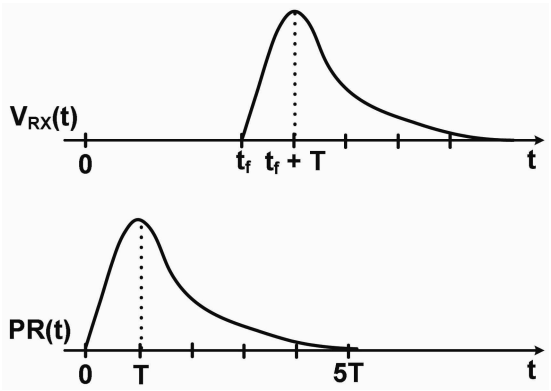


Fig. 5. Derivation of a SPICE single-pulse response PR(t) from $V_{RX}(t)$.

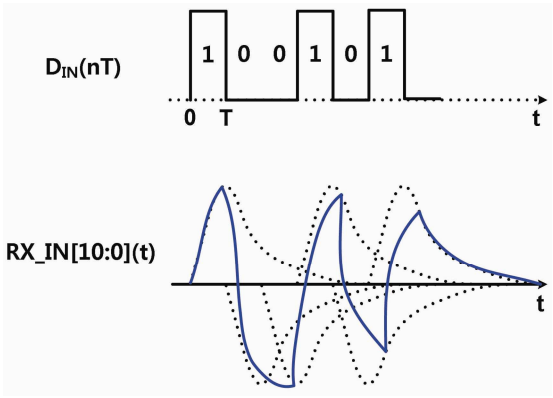


Fig. 6. Superposition of a Verilog single-pulse response (D_PR(t)) to find the receiver input waveform RX_IN[10:0](t) for the TX digital output ($D_{IN}(nT)$).

VRX(t), it starts from t = 0 in PR(t). This SPICE single-pulse response PR(t) is sampled in uniform time steps and the sampled values are stored as the Verilog single-pulse response D_PR(t) for a few data periods.

3. Verilog Model of Transmission Line

The SPICE model part of Fig. 3(a) is converted into a Verilog model, as shown in Fig. 3(b). The SPICE model part includes the transmission line, the CML driver, and the parasitic components. Also, the analog differential comparator of Fig. 3(a) is converted into the digital comparator of Fig. 3(b). In this way, all the components in the chip-to-chip interface are described entirely in Verilog. The Verilog single-pulse response D_PR(t) is used to get the receiver input waveform RX_IN(t) at RX for any digital data DIN(nT) transmitted at TX. At every data period of t=nT, either a positive or negative single-pulse response (+ D_PR(t) or – D_PR(t)) is delayed by nT and added to RX_IN(t) for the TX digital data DIN(nT) of 1 or 0, respectively. This can be expressed as

$$RX\_IN[10:0](t) = \sum_{n=0}^{\infty} PR(t-nT)\{2 * DIN(ntT) - 0.5\} \tag{4}$$

This superposition of single-pulse response can be performed since the transmission line is a linear-time-invariant(LTI) system. The Verilog single-pulse response D_PR(t) corresponds to an impulse response of LTI system, as can be seen in Eq. (4). Eq. (4) is a convolution function. Fig. 6 shows how to find the RX_IN(t)

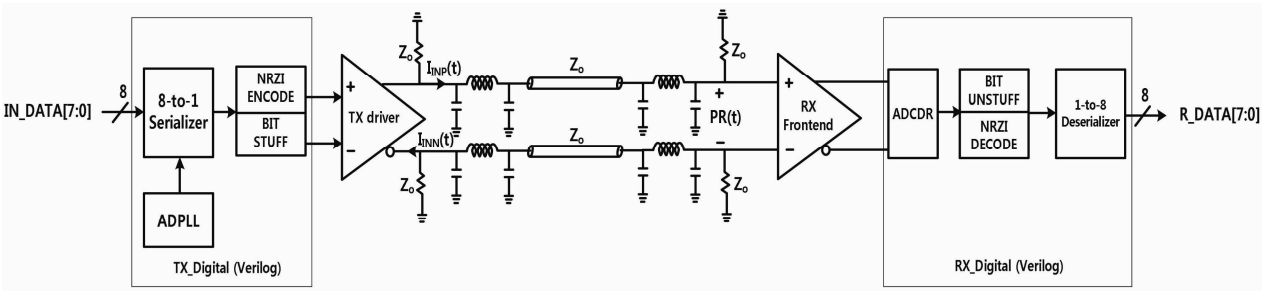
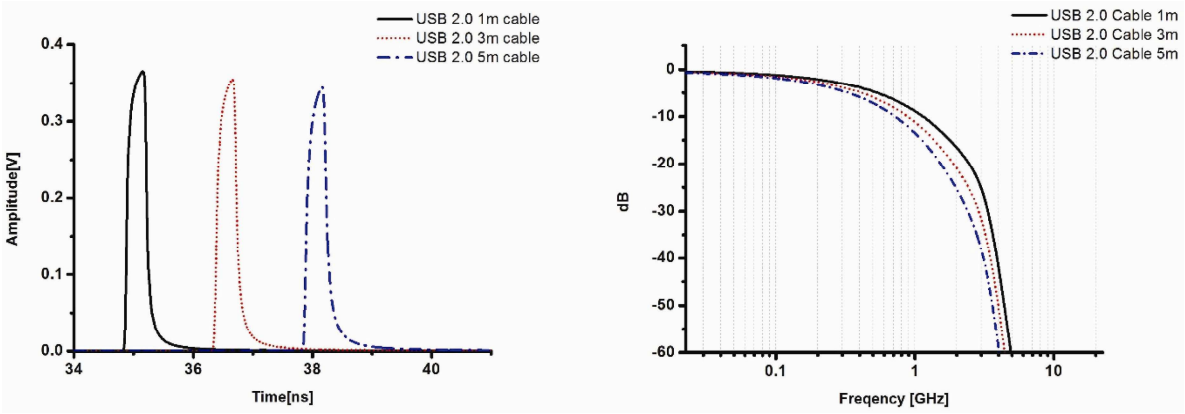


Fig. 7. Chip-to-chip interface through a USB 2.0 high-speed PHY [9].



(a) (b)

Fig. 8. A transient pulse response (a) and a frequency magnitude response, (b) of USB 2.0 cable.

waveform by the linear superposition for the TX output data of '100101'.

III. Application to USB 2.0 High-speed PHY Simulation

1. Verilog Model of USB 2.0 Cable

The proposed Verilog model of transmission line is applied to the USB 2.0 high-speed PHY simulation. Two USB 2.0 high-speed PHY chips (TX_Digital and RX_Digital) are connected through a USB cable, as shown in Fig. 7. Both TX_Digital and RX_Digital blocks are described entirely in Verilog. The transmission line part is also described in Verilog by using the method proposed in this work.

To find the SPICE single-pulse response PR(t) of a USB 2.0 cable by using SPECTRE, a lossy transmission line is used. The parameters of lossy transmission line model (W model), which are included in the RLGC file, are shown in Table 1. A π-shaped pin parasitic model with a 3nH inductor and two 1.5pF capacitors is also included in the USB cable model. For three USB 2.0 cables with different lengths(1m, 3m, 5m), a transient SPECTRE simulation is performed to find PR(t). A single current pulse of 18mA is applied to $I_{INP}(t)$ of transmitter (Fig. 3(a)) during one data period of 500Mbps. The analog PR(t) waveforms for the three USB cables are shown in Fig. 8(a), where t_{F1}, t_{F2}, andt_{F3} are the times of flight of each USB cable. Also, the frequency responses of USB cables are presented in Fig. 8(b). The digital D_PR(t) waveform is stored in a file named 'Single_Pulse_Response.txt' in a 11-bit resolution, starting from t = 0until t = 5T in ten ps steps. T is a data period(2ns) of 500Mbps. Only five data periods are stored for D_PR(t) since the ISI component of a single-pulse response is negligible after five data periods, as shown in Fig. 8(a).

The Verilog code to calculate the 11-bit receiver input waveform RX_IN[10:0](t) by using (4) is presented in Fig. 9. Only five preceding data values including the current one are used for the RX_IN computation. The RX_IN[10:0](t) output from the Verilog code is shifted

```
module BEH_CHANNEL

input                RESET;
input                CLK100G;     // Sampling Clock (100GHz)
input                CLK500;      // 500MHz Clock
input                DIN;         // 500 Mbps Data


output signed [10:0] RX_IN;       // Channel output

reg            [ 9:0] PulseResponse[0:999];
reg            [ 9:0] Pre_Cursor, Main_Data, Post_Cursor1, Post_Cursor2, Post_Cursor3;

initial
begin
 $readmemb( " Single_Pulse_Response.txt " , PulseResponse);
end

always @(posedge CLK100G or posedge RESET)
 if(RESET)           t <= 0;
 else
 begin
  if( t == 199 )     t <= 0;
  else               t <= t + 1;
 end


always @(posedge CLK100G or posedge RESET)
 if(RESET)
 begin
   Pre_Cursor     <= 0;
   Main_Data      <= 0;
   Post_Cursor1   <= 0;
   Post_Cursor2   <= 0;
   Post_Cursor3   <= 0;
 end

 else
 begin
  case (t)
    0 : Pre_Cursor <= Pulse[0];    Main_Data <= Pulse[200]; Post_Cursor1 <= Pulse[400]; Post_Cursor2 <= Pulse[600]; Post_Cursor3 <= Pulse[800];
    1 : Pre_Cursor <= Pulse[1];    Main_Data <= Pulse[201]; Post_Cursor1 <= Pulse[401]; Post_Cursor2 <= Pulse[600]; Post_Cursor3 <= Pulse[801];




199 : Pre_Cursor <= Pulse[199]; Main_Data <= Pulse[399]; Post_Cursor1 <= Pulse[599]; Post_Cursor2 <= Pulse[799]; Post_Cursor3 <= Pulse[999];


always @(posedge CLK500 or posedge RESET)
 if(RESET)           DIN            <= 0;
 else
 begin
                     DATA_POST3 <= DATA_POST2;
                     DATA_POST2 <= DATA_POST1;
                     DATA_POST1 <= DATA_MAIN;
                     DATA_MAIN  <= DATA_PRE;
                     DATA_PRE   <= DIN;
 end


assign  RX_PRE1    = (DATA_PRE1)   ? ( 0 + Pre_Cursor    ) : ( 0 – Pre_Cursor     );
assign  RX_MAIN    = (DATA_MAIN)   ? ( 0 + Main_Data     ) : ( 0 – Main_Data      );
assign  RX_POST1   = (DATA_POST1) ? ( 0 + Post_Cursor1 ) : ( 0 – Post_Cursor1 );
assign  RX_POST2   = (DATA_POST2) ? ( 0 + Post_Cursor2 ) : ( 0 – Post_Cursor2 );
assign  RX_POST3   = (DATA_POST3) ? ( 0 + Post_Cursor3 ) : ( 0 – Post_Cursor3 );


assign  RX_IN = RX_PRE1 + RX_MAIN + RX_POST1 + RX_POST2 + RX_POST3;


endmodule
```

Fig. 9. Verilog model for USB 2.0 Cable.

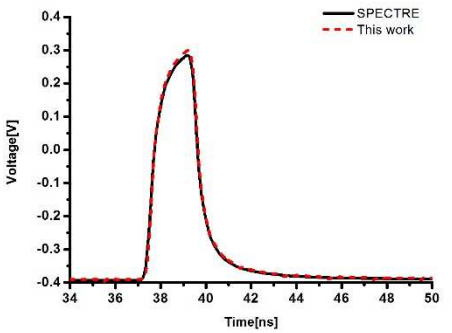
Table 1. W model parameters of USB 2.0 cable

Parameters	L_s[H/m]	L_m[H/m]	C_s[F/m]	C_m[F/m]
Values	2.03×10^{-7}	1.68×10^{-8}	9.96×10^{-11}	-5.18×10^{-12}
Parameters	R_{os}[Ω/m]	R_{ss}[Ω/m]	G_{ds}[$Ω^{-1}$/m]	G_{dm}[$Ω^{-1}$/m]
Values	2.81×10^{-1}	2.53×10^{-4}	2.08×10^{-11}	-8.81×10^{-13}

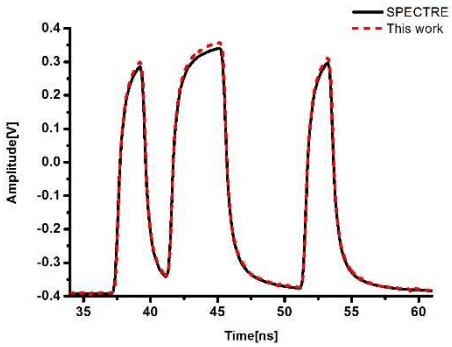
by t_F in time axis for the final result.

2. Comparison of this work with SPECTRE simulation

To verify the validity of this work, a single-pulse response of receiver input waveform is compared between this work and SPECTRE. Excellent agreements



(a)



(b)

Fig. 10. Comparison between this work and SPECTRE with 5m USB 2.0 cable (a) single-pulse response, (b) response to '1011000100' input.

can be observed with the maximum relative error of 4.05% as shown in Fig. 10(a).

To simulate the real environment, the input data(IN_DATA[7:0]) stream of Fig. 7 is extracted by applying a USB protocol analyzer to a real USB 2.0 high-speed interface between a memory stick and PC, starting from the plug-in time until the memory stick device is recognized by PC after the device configuration stage is completed [8]. It takes around 3us in a real USB 2.0 high-speed interface. The extracted IN_DATA[7:0] stream for the entire 3us interval is applied both to the Verilog model(Fig. 3(a)) of this work and SPECTRE (Fig. 3(a)). The data rate of 500Mbps with a 5m USB 2.0 cable is used in this simulation. The eye patterns from this simulation are compared in Fig. 11, which shows excellent agreements. The receiver output data(OUT_DATA[7:0] of Fig. 7) are obtained from Verilog simulation of this work, and are presented in Fig. 12 for the very initial part of the 3us simulation. Exact matches can be found between the TX input data(IN_DATA[7:0]) and the RX output data. The simulation statistics are

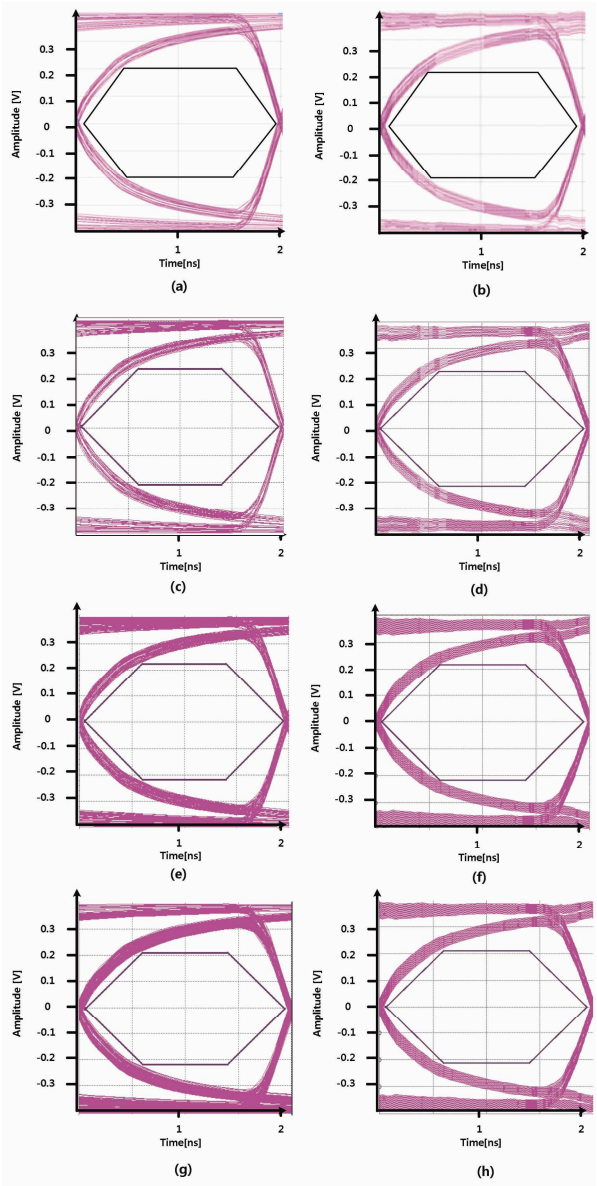


Fig. 11. Comparison of eye patterns (a) SPECTRE, (b) this work of USB 2.0 Descriptor, (c) SPECTRE, (d) this work of 2^7-1 test pattern, (e) SPECTRE, (f) this work of 2^{15}-1 test pattern, (g) SPECTRE, (h) this work of 2^{31}-1 test pattern.

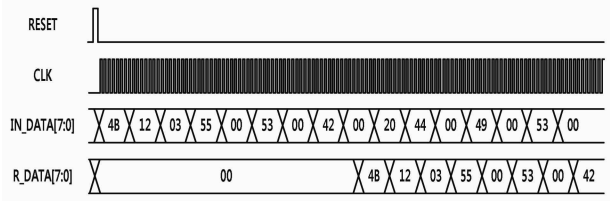


Fig. 12. Received digital output code (OUT_DATA[7:0]) from simulation of this work.

compared in Tables 2 and 3. This work is faster than SPECTRE by around 770 times.

Table 2. Comparison of simulation statistics between this work and SPECTRE

	SPECTRE	This work
TX, RX netlist	Verilog	Verilog
TL netlist	SPICE	Verilog
Total simulation time[sec]	126,000	163
#Time point	30,000	30,000

Table 3. Comparison of simulation statistics between this work and SPECTRE of USB 2.0 Descriptor, PRBS 2^7-1, 2^{15}-1 and 2^{31}-1 Test pattern

Channel	Test Pattern	Voltage Opening		Time Opening	
		SPETRE	This work	SPETRE	This work
USB 2.0 5m Cable	USB 2.0 Descriptor	66.5%	66.5%	97%	97%
	2^7-1 PRBS	68.75%	68.73%	97.5%	97.5%
	2^{15}-1 PRBS	67.5%	67.1%	97.5%	97.2%
	2^{31}-1 PRBS	66.5%	66.5%	97%	97%

IV. CONCLUSIONS

The SPICE netlist of transmission line is converted into a Verilog model in the chip-to-chip USB 2.0 high-speed PHY interface to reduce the simulation time by around 770 times compared to the SPECTRE mixed-mode simulation. It takes less than 3 minutes to simulate the entire period(~3us) of device configuration stage of USB 2.0 high-speed interface, while it takes more than 2 days with SPECTRE. The USB 2.0 interface circuit used in this work consists of a TX digital part, a USB 2.0 cable and a RX digital part.

Both the TX and the RX digital parts are modeled in Verilog for both this work and SPECTRE. The USB 2.0 cable is modeled in Verilog for this work and in SPICE netlist for SPECTRE. A single-pulse response of five data periods is calculated by SPECTRE and stored in a file. The stored single-pulse response is linearly added or subtracted with appropriate delays to calculate the receiver input waveform depending on the TX data values. The receiver input waveform is represented in a 11-bit digital code. Excellent agreements are observed between this work and SPECTRE in waveforms, eye patterns and code values.

ACKNOWLEDGMENTS

This research was supported by ITRC(NIPA-2013-H0301-13-1007), IDEC.

REFERENCES

[1] M. Choi et al. "A Channel Model of Scaled RC-dominant Wires for High-Speed Wireline Transceiver Design", *Journal of Semiconductor Technology and Science(JSTS)*, vol. 13, no. 5, pp. 482- 491, Oct. 2013.

[2] M. Van Ierssel, H. Yamaguchi and A. Sheikholeslami, "Event-driven modeling of CDR jitter induced by power supply noise, finite decision-circuit bandwidth and channel ISI", *IEEE Trans. Circuits Syst. I*, vol. 55, no. 5, pp. 1306 – 1315, May. 2008.

[3] K. S. Kundert, "Predicting the phase noise and jitter of PLL-based frequency synthesizers", in *Phase-Locking in High-Performance Systems : From Devices to Architectures*. IEEE Press, 2003.

[4] Sabrina Liao et al., "A Verilog Piecewise-Linear Analog Behavior Model for Mixed-Signal Validation", *IEEE Custom Integrated Circuits Conference (CICC)*, Sep. 2013.

[5] M. Park et al., "Fast and Accurate Event-Driven Simulation of Mixed-Signal Systems with Data Supplementation", *IEEE Custom Integrated Circuits Conference (CICC)*, Sep. 2011.

[6] Mohammad Ashraf et al. "A FULL FUNCTION VERILOG PLL LOGIC MODEL", *IEEE Custom Integrated Circuits Conference (CICC)*, Sep. 1997.

[7] "Spectre Circuit Simulator User Guide", Product Version 5.0, July 2002

[8] "USB2.0 Transceiver Macrocell Interface Specification, Revision 1.05," Intel, Hillsboro, OR, Mar. 29, 2001.

[9] S. Park et al., "A Single-Data-Bit Blind Oversampling Data-Recovery Circuit With an Add-Drop FIFO for USB2.0 High-Speed Interface," *IEEE Trans. Circuits Syst. II*, vol. 55, no. 2, pp. 156 – 160, Feb. 2008.

Ki-Hwan Seong was born in Dae-gu, Korea, on 1985. He received the B.S. degree in the Department of Electronic and Electrical Engineering from Kyung-Pook National University, Korea, in 2009 and M.S. degree in Electronic Engineering from Pohang University of Science and Technology (POSTECH), Korea, in 2011, respectively. He is currently pursuing the Ph.D. degree in the Department of Electronic and Electrical Engineering from Pohang University of Science and Technology (POSTECH), Korea. His interests include high-speed interface circuits (USB2.0/3.0/3.1) and all-digital synthesizable circuits.

Ji-Hoon Lim was born in Seoul, Korea, on 1989. He received the B.S. degree in the Department of Electronic and Electrical Engineering from Pohang University of Science and Technology (POSTECH), Korea, in 2011. He is currently pursuing the M.S. and Ph.D. degrees in the Department of Electronic and Electrical Engineering from Pohang University of Science and Technology (POSTECH), Korea. His interests include data converters, clock and data recovery, high-speed interface circuits and ultra-low-voltage analog circuits.

Byungsub Kim received the B.S. degree in Electronic and Electrical Engineering (EEE) from Pohang University of Science and Technology (POSTECH), Pohang, Korea, in 2000, and the M.S. (2004) and Ph.D. (2010) degrees in Electrical Engineering and Computer Science (EECS) from Massachusetts Institute of Technology (MIT), Cambridge, USA. From 2010 to 2011, he worked as an analog design engineer at Intel Corporation, Hillsboro, OR, USA. In 2012, he joined the faculty of the department of Electronic and Electrical Engineering at POSTECH, where he is currently working as an assistant professor. He received several honorable awards. In 2011, Dr. Kim

received MIT EECS Jin-Au Kong Outstanding Doctoral Thesis Honorable Mentions, and IEEE 2009 Journal of Solid-State Circuits Best Paper Award. In 2009, he received Analog Device Inc. Outstanding Student Designer Award from MIT, and was also a co-recipient of the Beatrice Winner Award for Editorial Excellence at the 2009 IEEE Internal Solid-State Circuits Conference.

Jae-Yoon Sim received the B.S., M.S., and Ph.D. degrees in Electronic and Electrical Engineering from Pohang University of Science and Technology (POSTECH), Korea, in 1993, 1995, and 1999, respectively. From 1999 to 2005, he worked as a senior engineer at Samsung Electronics, Korea. From 2003 to 2005, he was a post-doctoral researcher with the University of Southern California, Los Angeles. From 2011 to 2012, he was a visiting scholar with the University of Michigan, Ann Arbor. In 2005, he joined POSTECH, where he is currently an Associate Professor. He has served in the Technical Program Committees of the International Solid-State Circuits Conference (ISSCC), Symposium on VLSI Circuits, and Asian Solid- State Circuits Conference. He is a co-recipient of the Takuo Sugano Award at ISSCC 2001. His research interests include high-speed serial/parallel links, PLLs, data converters and power module for plasma generation.

Hong-June Park received the B.S. degree from the Department of Electronic Engineering, Seoul National University, Seoul, Korea, in 1979, the M.S. degree from the Korea Advanced Institute of Science and Technology, Taejon, in 1981, and the Ph.D. degree from the Department of Electrical Engineering and Computer Sciences, University of California, Berkeley, in 1989. He was a CAD engineer with ETRI, Korea, from 1981 to 1984 and a Senior Engineer in the TCAD Department of INTEL from 1989 to 1991. In 1991, he joined the Faculty of Electronic and Electrical Engineering, Pohang University of Science and Technology (POSTECH), Gyeongbuk, Korea, where he is currently Professor. His research interests include CMOS analog circuit design such as high-speed interface circuits, ROIC of touch sensors and analog/digital beamformer circuits for ultrasound medical imaging. Prof. Park is a senior member of IEEE and a member of IEEK. He served as the Editor-in-Chief of Journal of Semiconductor Technology and Science, an SCIE journal (http://www.jsts.org) from 2009 to 2012, also as the Vice President of IEEK in 2012 and as the technical program committee member of ISSCC, SOVC and A-SSCC for several years.