Macromodeling of Differential Drivers

Macromodelling of differential drivers

I.S. Stievano, I.A. Maio and F.G. Canavero

Abstract: The development of macromodels of differential drivers for the prediction of the analogue operation of very high-speed digital communication links is explored. The macromodels are mathematical relations hiding the information on the internal structure of devices; they are estimated from port device responses and can be easily implemented in any circuit or analogue mixed-signal simulator as SPICE-like subcircuits or VHDL-AMS code descriptions. Accuracy and efficiency are assessed by applying the modelling procedure to actual devices.

1 Introduction

Low voltage differential signalling (LVDS) is to be established as the dominant standard for on-board and off-board high-performance data links [1–3]. It allows extremely high data rates, in the order of one Gbps, along with reduced electromagnetic interference effects and reduced power dissipation. In order to simulate the operation of LVDS links for the assessment of signal integrity (SI) and electromagnetic compatibility (EMC) problems, suitable behavioural models (or macromodels) of differential drivers and receivers are needed. The macromodels must be efficient and accurate enough to handle the complexity of actual simulation problems and to yield reliable predictions of sensitive effects such as crosstalk or radiation.

A common approach to the modelling of devices is via simplified equivalent circuit representations, in which the information on the internal structure of the device is used to devise a simplified equivalent circuit. The equivalent circuit is composed of various blocks, accounting for a specific static or dynamic effect. A well-known example is provided by the input/output buffer information specification (IBIS) [4], which has been established as a standard for the description of the ports of a digital integrated circuit (IC), leading to a large availability of device descriptions and commercial tools handling models based on IBIS. When properly designed and estimated, IBIS models for regular devices have been proven to be accurate enough for the simulation problems at hand. Recent advances on the modelling of differential driver models based on IBIS can be found in the work of Hegazy and Korany [5] and Muranyi [6]. However, the growing complexity of recent devices and their enhanced features such as pre-emphasis and specific control circuitry demands for more and more refinements of the basic equivalent circuits, and calls for the present version of IBIS to include external models. Such an extension is known as the IBIS multilingual extension, and allows the inclusion of models in different possible languages such as SPICE or VHDL-AMS.

Within this framework, this paper proposes a possible modelling alternative based on mathematical relations and circuit theory, with the aim of reproducing the electrical behaviour of device ports, without any use of physical insights and of equivalent circuit representations. The advantage of this approach relies in the flexibility of the equation descriptions with respect to the circuit representation. In particular, the parasitic effects and some of the exotic effects inherent to the nonlinearity of devices are difficult to capture if we have at our disposal only capacitors, inductors and resistors (even if nonlinear). On the contrary, equations allow us to better fit the complex behaviour of components. Besides, the proposed macromodels can be easily implemented in any simulation tool.

2 LVDS devices and their model structure

The output buffers of LVDS drivers operate via current steering techniques, as shown in Fig. 1. Two voltage-controlled current source devices are used to provide the current sent to and drawn from resistor R_r at receiver input terminals. When switches A are closed, i_r is positive, whereas when switches B are closed i_r is negative, and the voltage across receiver input terminals changes polarity. In actual applications, output buffers may contain matching resistors across the output terminals and control subcircuits to ensure proper output current and voltage values over possible process, supply voltage and temperature variations (e.g. see the work of Boni *et al.* [7], Young [8] and Gabara *et al.* [9] for possible implementations of control circuits).

In fixed logic state, the ideal LVDS output buffer of Fig. 1 can be considered as a three-terminal circuit element characterised by constitutive relations (which we call submodels) of the form

$$\begin{cases} i_1 = i_{1H}(v_1, v_2) \\ i_2 = i_{2H}(v_1, v_2) \end{cases} \quad \begin{cases} i_1 = i_{1L}(v_1, v_2) \\ i_2 = i_{2L}(v_1, v_2) \end{cases} \tag{1}$$

where H and L denote the HIGH and LOW logic state, respectively, and the output currents are allowed to be functions of both terminal voltages to take into account variants of the buffer basic scheme with internal resistor and control circuits. As an alternative, the above submodels can be expressed in terms of different variables obtained as linear combinations of port voltages v_1 and v_2. A typical set of alternative variables are the common mode voltage $v_c = (v_1 + v_2)/2$ and the differential voltage $v_d = (v_1 - v_2)$.

A complete macromodel describing state switching from steady-state operation can be obtained by combining the

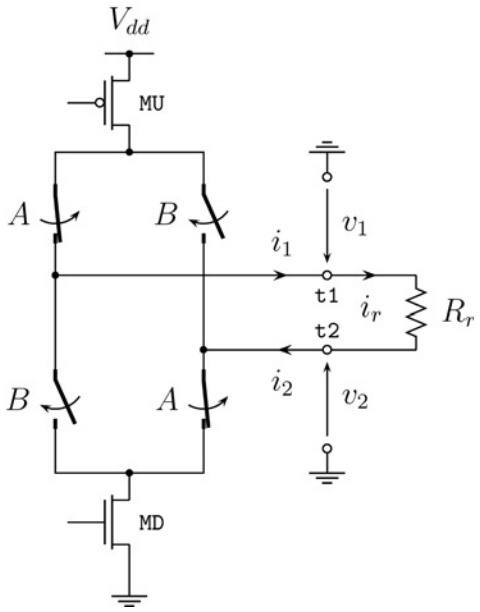

Fig. 1 *Generic structure of a LVDS driver and its relevant electric variables*

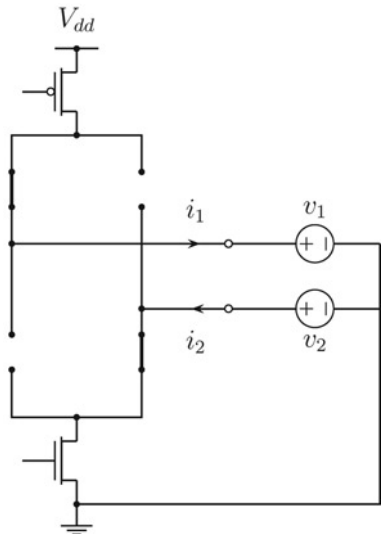

Fig. 2 *Common setup for both the estimation of the static characteristics and the dynamic behaviour of the LVDS device of Fig. 1 in the HIGH logic state*

submodels (1) by means of time-varying weighting coefficients, as already proposed in the work of Stievano *et al.* [10, 11] for the case of single-ended devices. Such a combined two-piece model writes

$$\begin{cases} i_1 = w_{1H}(t) i_{1H}(v_1, v_2) + w_{1L}(t) i_{1L}(v_1, v_2) \\ i_2 = w_{2H}(t) i_{2H}(v_1, v_2) + w_{2L}(t) i_{2L}(v_1, v_2) \end{cases} \tag{2}$$

where w_{nH} and w_{nL}, $n = 1, 2$, are the weighting coefficients accounting for logic state transitions. It is ought to remark that representation (2) approximates the external device behaviour including the information on state transitions without assumptions on the device internal structure.

The problem is then to devise suitable relations for the submodels of (1), to estimate their parameters and estimate the weighting coefficients of (2). A straightforward approach is to represent i_{nH} and i_{nL} by a sum of a static mapping and a (possibly nonlinear) relation taking into account dynamic effects, as discussed in the work of Stievano *et al.* [11]. The representation of $i_{nH}(v_1, v_2)$ is

$$\begin{cases} i_{1H}(v_1, v_2) = \hat{i}_{1H}(v_1, v_2) + \bar{i}_{1H}\left(v_1, v_2, \dfrac{\mathrm{d}}{\mathrm{d}t}\right) \\ i_{2H}(v_1, v_2) = \hat{i}_{2H}(v_1, v_2) + \bar{i}_{2H}\left(v_1, v_2, \dfrac{\mathrm{d}}{\mathrm{d}t}\right) \end{cases} \tag{3}$$

where $\hat{i}_{1H}$ and $\hat{i}_{2H}$ are the static characteristics of currents i_1 and i_2 for the driver forced in the fixed HIGH logic state and $\bar{i}_{1H}$ and $\bar{i}_{2H}$ are the dynamic submodels. Similar equations occur for $i_{nL}(v_1, v_2)$ of (2).

Model representations defined by parametric relations can be effectively used for the dynamic terms in (3). This choice comes from the well-established theory of system identification that uses parametric relations for approximating the nonlinear dynamic behaviour of almost any nonlinear dynamical system. A complete review of possible representations as well as the methods for estimating their parameters can be found in the work of Sjöberg *et al.* [12].

Under the simplified assumption of dealing with devices with dynamic behaviour dominated by linear capacitive effects, the dynamic terms in (3) can be simply replaced by linear parametric models involving the sole derivatives of the port voltages only, that is

$$\begin{cases} \bar{i}_{1H} = \alpha_{11H} \dfrac{\mathrm{d}v_1}{\mathrm{d}t} + \alpha_{12H} \dfrac{\mathrm{d}v_2}{\mathrm{d}t} \\ \bar{i}_{2H} = \alpha_{21H} \dfrac{\mathrm{d}v_1}{\mathrm{d}t} + \alpha_{22H} \dfrac{\mathrm{d}v_2}{\mathrm{d}t} \end{cases} \tag{4}$$

where the coefficients α are constant. Of course, the dynamic submodels $\bar{i}_{1L}$ and $\bar{i}_{2L}$ take the same structure as (4), with different constants.

Submodels representation defined by (3) and (4) approximates the port constitutive relations and includes both static and dynamic coupling effects between the terminal variables without any specific assumption on the internal structure of device. Besides, if needed, the linear capacitive relation (4) can be improved and replaced by an arbitrary nonlinear parametric model when this simplified assumption is not met [10, 12].

Both the static mapping and the dynamic part can be estimated from currents caused by suitable test sources connected to driver output terminals, as in Fig. 2. The static mappings can be readily derived by means of DC analyses. Of course, the terminal voltage swings applied by test sources should correspond to differential and common mode voltage variations within limits specified by the LVDS standard. The dynamic parts, instead, can be estimated from transient responses $i_1(t)$ and $i_2(t)$ (e.g. those caused by large fast variations of sources of the estimation setup of Fig. 2) by matching the submodel responses to the recorded ones. As an example, the estimation of the linear parameters of (4) is achieved by recording the current responses, while the driver is forced in the HIGH logic state. The recorded port voltage and current waveforms are sampled using a sampling period T on the order of $^1/_{10}$–$^1/_{20}$ of the port switching time, leading to the sequences $\{i_1(k), i_2(k), v_1(k), v_2(k)\}$, $k = 1, \ldots, N$. Such sequences are then used in (3) and (4) by recasting those equations in terms of a linear least squares problem as follows

$$\begin{bmatrix} i_1(1) - \hat{i}_{1H}(v_1(1), v_2(1)) \\ \cdots \\ i_1(N) - \hat{i}_{1H}(v_1(N), v_2(N)) \\ i_2(1) - \hat{i}_{2H}(v_1(1), v_2(1)) \\ \cdots \\ i_2(N) - \hat{i}_{2H}(v_1(N), v_2(N)) \end{bmatrix} = \begin{bmatrix} M & 0 \\ 0 & M \end{bmatrix} \begin{bmatrix} \alpha_{11H} \\ \alpha_{12H} \\ \alpha_{21H} \\ \alpha_{22H} \end{bmatrix} \tag{5}$$

where

$$M = \begin{bmatrix} \frac{dv_1}{dt}(1) & \frac{dv_2}{dt}(1) \\ \cdots & \cdots \\ \frac{dv_1}{dt}(N) & \frac{dv_2}{dt}(N) \end{bmatrix} \quad (6)$$

and the differential operator in (6) is replaced by a difference operator (e.g. $dx/dt(k) \simeq (1/T)[x(k) - x(k-1)]$).

Once the submodels i_{1H}, i_{1L} and i_{2H}, i_{2L} in (2) are completely defined and their parameters estimated, the weighting signals $w_{1H}(t), w_{1L}(t)$ and $w_{2H}(t), w_{2L}(t)$ must be computed by means of the procedure reported below. For the sake of simplicity, the following discussion is based on the first equation in (2), that is, on the macromodel for the current i_1 only. The same procedure is used for the macromodel for the current i_2.

(i) The device port voltage and current responses are collected while the driver is connected to one (or more than one) reference load and is forced to produce a single-up (bit stream '0111 ...') and a single-down (bit stream '1000 ...') state transitions. As an example, Fig. 3 shows the test setup for the generation of the port transient responses for the up transition event.

(ii) The elementary up, $w_{1H\uparrow}$ and $w_{1L\uparrow}$, and down, $w_{1H\downarrow}$ and $w_{1L\downarrow}$, weighting coefficients are computed via linear inversion of the model equation (2) from the device port responses recorded during the previous transition events. For the example setup of Fig. 3, the driver is directly connected to the input port of the transceiver that will be used in a real application of the device. For the sake of simplicity, only one load is considered and the assumption $w_{1L\uparrow} = (1 - w_{1H\uparrow})$ is exploited. Under the above conditions, (2) becomes

$$i_{1u}(t) \simeq w_{1H\uparrow}(t) i_{1H}\left(v_{1u}, v_{2u}, \frac{d}{dt}\right) + (1 - w_{1H\uparrow}(t)) i_{1L}\left(v_{1u}, v_{2u}, \frac{d}{dt}\right) \quad (7)$$

and the basic weighting coefficient $w_{1H\uparrow}$ is computed by direct inversion of (7).

(iii) The complete weighting coefficients w_{1H} and w_{1L} accounting for a specific logic activity of the driver are obtained by generating a sequence in time by juxtaposition of part of the elementary weighting coefficients of up and down transitions (e.g. see Fig. 4, where the basic coefficients are concatenated for the generation the bit stream '01001110 ...').

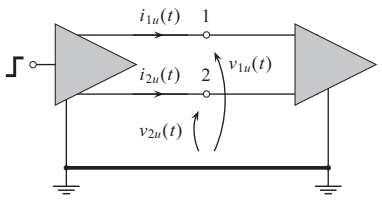

Fig. 3 *Test setup for the generation of the transient waveforms used for the computation of the basic weighting coefficients $w_{1H\uparrow}$, $w_{1L\downarrow}$ in (7)*

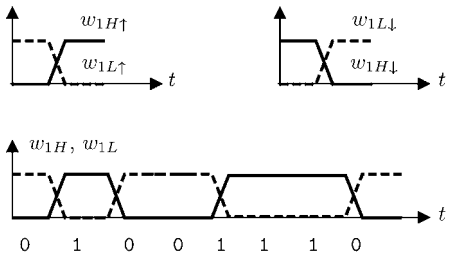

Fig. 4 *Example elementary weighting coefficients $w_{1H\uparrow}$, $w_{1L\uparrow}$ for the up state transition and $w_{1H\downarrow}$, $w_{1L\downarrow}$ for the down state transition (top panel)*

Weighting coefficients w_{1H} and w_{1L} obtained as juxtaposition of the elementary weighting coefficients for the generation of the bit pattern '01001110'

3 Macromodel implementations

The standard procedure to use models based on equations, (2)–(4) in circuit simulation environments, is to convert them into an equivalent circuit and implement it as a SPICE-like subcircuit. Such a conversion and implementation is a standard procedure, and is based on current-controlled sources for the static submodels of (3), and on C elements and possible controlled source components for the dynamic parts. However, it is worth noting that the basic keywords available in any SPICE-type simulator do not allow to easily implement the juxtaposition of the elementary weighting coefficients of up and down transitions as shown in Fig. 4. Owing to this, the possible sources that implement the weighting coefficients in (2) must be computed offline for a predetermined bit pattern. They must include all the evolution of the weighting signals for the predetermined bit stream, leading to large model sizes for a large number of bits.

On the other hand, the recent interest and availability of integrated analogue mixed-signal simulation tools drove the attention to other possible model descriptions via metalanguages such as Verilog-AMS or VHDL-AMS. Such tools allow the simulation of the analogue propagation paths between drivers and receivers, possibly including the interaction between the internal functional part of the IC and the analogue output ports of buffers driving the external interconnects. Besides, they greatly facilitate the juxtaposition of the elementary weighting sequences since they include commands for logical operation. A detailed description of the VHDL-AMS language is out of the scope of this paper and can be found in [13, 14].

It is worth noting that both the SPICE-based and VHDL-AMS implementations can be effectively used in most commercial tools for SI/EMC since they accept IBIS descriptions of ICs as a standard way for the inclusion of devices models into them, and the IBIS multilingual extension allows the freedom to provide external and enhanced device models in different formats.

4 Application examples

In this section, the proposed modelling approach is demonstrated on two different example devices defined by detailed transistor-level models, which are assumed as the reference models hereafter. Reference models are used to compute the responses needed for the estimation of macromodel parameters and for model validations. All the required responses are computed by means of HSPICE. Both the examples are addressed by the model representation (4)

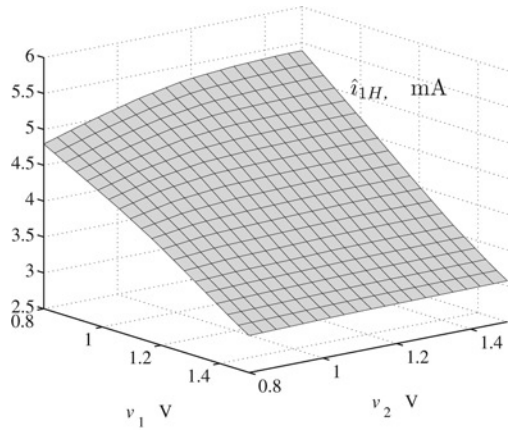

Fig. 5 *Static characteristic $\hat{i}_{1H}(v_1, v_2)$ for Example 1 driver forced in the HIGH logic state*

and the obtained models are implemented as SPICE-like subcircuits and as VHDL-AMS code descriptions. For the latter case, the Mentor Graphics ICX Ver. 3.2 simulation tool is used for the simulation of driver macromodels. The ICX tool allows the description of IC ports by means of the IBIS specification, with the inclusion of externally specified models, as outlined in Section 2. The examples considered in this study are a plain differential driver and an enhanced driver exploiting a control mechanism to reduce the fluctuations of the common mode voltage around a reference voltage. Some results on the application of the proposed methodology to complex devices with pre-emphasis can be found in the work of Stievano *et al.* [15].

4.1 Example 1

The first modelled device is the Fairchild FIN1001 ($V_{dd} = 3.3$. V) LVDS High Speed Differential Driver, whose HSPICE encrypted transistor-level model is available at www.fairchildsemi.com. This device behaves like a plain differential driver (see Fig. 1) without internal matching resistors or control mechanisms.

For the macromodel estimation, both the static and the dynamic parts of (4) are computed through the procedure discussed in the previous section. As an example, Fig. 5 shows the static characteristic $\hat{i}_{1H}(v_1, v_2)$. In order to facilitate the model implementation, the static characteristics $\hat{i}_{1H}$ and $\hat{i}_{2H}$, which are known as sets of sampled DC

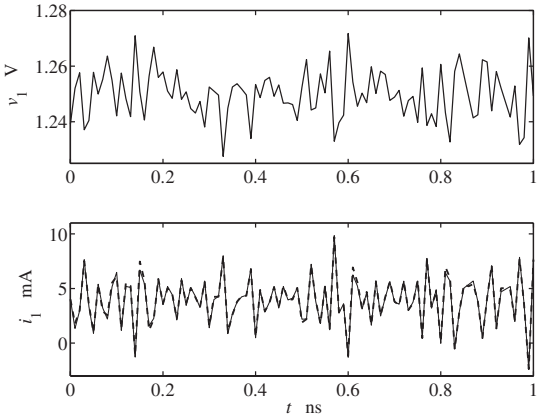

Fig. 6 *Gaussian source $v_1(t)$ applied to Example 1 device port terminal as in Fig. 2 for the estimation of the coefficients for the dynamic part of the model (top panel)*

Driver response $i_1(t)$
Solid line: reference; dashed line: macromodel

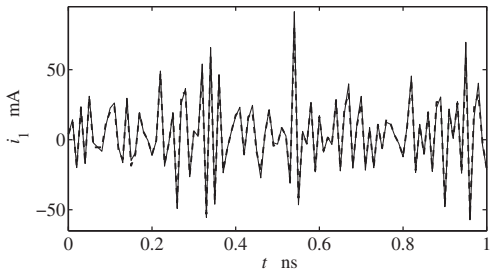

Fig. 7 *Device port response $i_1(t)$ computed when Example 1 driver is connected as in Fig. 2 where the voltage sources are independent Gaussian noise sources with mean value of 1.25 V and standard variation of 100 mV*

Solid line: reference; dashed line: macromodel

curves, are approximated by suitable analytical expressions (i.e. sigmoidal expansions in this case [12, 16, 17]). The dynamic parts are obtained from the port current responses i_1 and i_2 to independent Gaussian noise sources v_1 and v_2 connected as in Fig. 2 with mean value equal to the nominal common mode voltage (e.g. 1.25 V) and 10 mV amplitude standard deviation. As an example, Fig. 6 shows the voltage waveform of the Gaussian source $v_1(t)$ and the device port response $i_1(t)$ computed by the reference transistor-level model of the example device and by the estimated macromodel. The waveform of the source $v_2(t)$, not shown in Fig. 6, is just a different realisation of the same gaussian noisy signal used for v_1. The unknown coefficients defining the dynamic part of the model are obtained from the curves in Fig. 6 by solving the standard linear least squares problem (5). The estimated values of the unknown coefficients of (5) are $\{\alpha_{11H}, \alpha_{12H}, \alpha_{21H}, \alpha_{22H}\} = \{-1.775, -0.125, 0.125, 1.785\}$ pF and $\{\alpha_{11L}, \alpha_{12L}, \alpha_{21L}, \alpha_{22L}\} = \{-1.769, -0.109, 0.109, 1.729\}$ pF. In addition, the linearity of the dynamic part has been verified by applying noisy signals with amplitude on the order of the full voltage swing of 350 mV specified by the LVDS standard. Fig. 7 shows the comparison between the reference and the predicted responses of the current $i_1(t)$ computed for the same test case of Fig. 2 where the standard deviation of the two Gaussian sources has been increased to 100 mV. The weighting coefficients are computed as described in the previous section, by means of switching experiments while the device is connected to the input

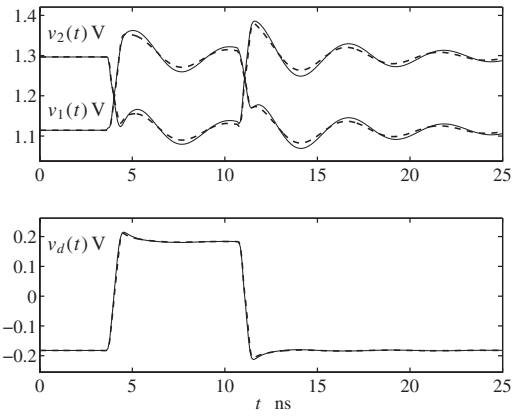

Fig. 8 *Output port voltages $v_1(t)$, $v_2(t)$ (top panel) and differential voltage $v_d(t)$ (bottom panel) computed for Example 1 driver connected to a 50 Ω differential resistor and producing a bit stream '010'*

Solid line: reference; dashed line: macromodel

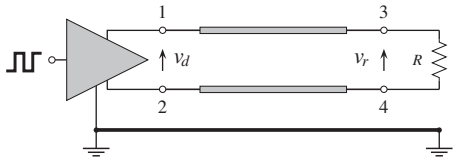

Fig. 9 *Application test circuit for the validation of Example 1 driver*

port of the transceiver that will be used in a real application of the device.

In order to validate the complete macromodel, two different simulation test cases are considered. The first test circuit is composed of the modelled device driving a 50 Ω differential resistor with a logic HIGH pulse. For this test case, Fig. 8 shows the reference and macromodel responses of the output terminal voltages $v_1(t)$, $v_2(t)$ and of the differential voltage $v_d(t)$.

The accuracy of the proposed macromodel has been quantified by computing the timing error and the maximum relative voltage error. The timing error is defined as the maximum delay between the reference and the macromodel differential voltage responses measured for the zero voltage crossing. For the test cases illustrated in Fig. 8, the maximum timing error is 15 ps. The maximum relative voltage error is computed as the maximum error between the reference and macromodel voltage responses divided by the nominal voltage swing of 350 mV. For the previous validation case, the maximum relative error turns out to be 5.4%.

The second test circuit consists of the more realistic case shown in Fig. 9 where the driver is connected to a coupled transmission line (common mode impedance $Z_c = 50\ \Omega$, differential mode impedance $Z_d = 100\ \Omega$, line length 0.15 m, $v_o = 2.5 \times 10^8$ m/s, $v_e = 2.6 \times 10^8$ m/s) loaded by an $R = 50\ \Omega$ differential resistor. The data pattern used for this study is a 2048 bit-long sequence with 2 ns bit time and jitter error uniformly distributed in the range [−100, 100] ps. For this test case, Fig. 10 shows the reference and macromodel responses of the far-end differential voltage $v_r(t)$, for a duration of 35 ns, picked at random along the simulation of the entire bit pattern. The macromodel response is obtained either by using the HSPICE and the VHDL-AMS implementations of the macromodel. Also in this realistic situation, a very good agreement between the reference and macromodel responses is achieved. In order to quantify the maximum errors in the predicted waveforms, the complete eye diagrams derived from both the reference and predicted waveform of $v_r(t)$ are compared, as shown in Fig. 11. Such a comparison is done by computing the eye apertures ΔT

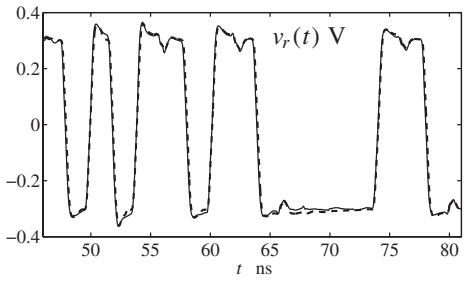

Fig. 10 *Differential voltage waveform $v_r(t)$ computed for Example1 driver producing a 2048 long bit stream*

The driver is connected to a distributed load as shown in Fig. 9 (see text for details)
Solid line: reference; dashed line: macromodel

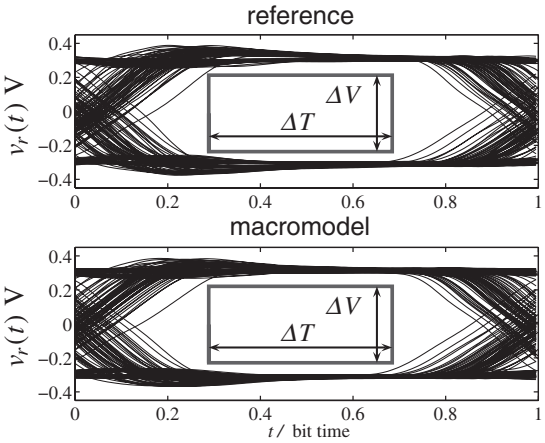

Fig. 11 *Eye diagram arising from the reference and predicted waveforms $v_r(t)$ of the test case of Fig. 9 and definition of the eye opening parameters ΔV and ΔT*

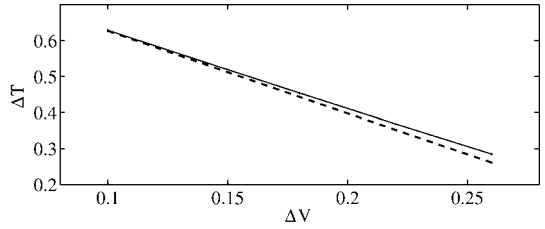

Fig. 12 *Eye opening parameter values (defined in Fig. 11) for the eye diagrams of waveforms $v_r(t)$ of Fig. 11*

Solid thin line: reference; dashed thick line: macromodels

and ΔV defined as in Fig. 11. Plots of ΔT against ΔV are shown in Fig. 12, where, for every value of ΔV, the difference of the corresponding ΔTs quantifies the error in the eye opening caused by the use of approximate waveforms. The above comparison highlights that the openings of eye diagrams obtained from simulations with macromodels are within 1–2% of openings from reference simulations for the entire 2048 bit long sequence.

Finally, macromodel efficiency is assessed by the speed-up factor of the different implementations of Example 1 driver macromodel with respect to the CPU time required by the simulation of the reference transistor-level model of the driver. Table 1 collects the figures of the efficiency comparison for the computation of eye diagram of Fig. 11, thus highlighting the efficiency of the proposed macromodel implemented in either SPICE or VHDL-AMS. Since we used different simulation environments running on different machines, for a fair comparison, every CPU time has been normalised to the CPU time for the simulation of a simple RC circuit in the same simulation setup. From the comparison carried out in this study, it is

Table 1: Speed-up factor introduced by different implementations of Example 1 driver macromodel w.r.t. CPU time for simulation of the reference transistor-level model of the driver for computation of the eye diagrams of Fig. 11 (see text for more details)

Model	Implementation	Simulator	Speed-up
Reference	SPICE	HSPICE	1
Macromodel	SPICE	PSPICE	10
Macromodel	VHDL-AMS	ICX Mentor	7

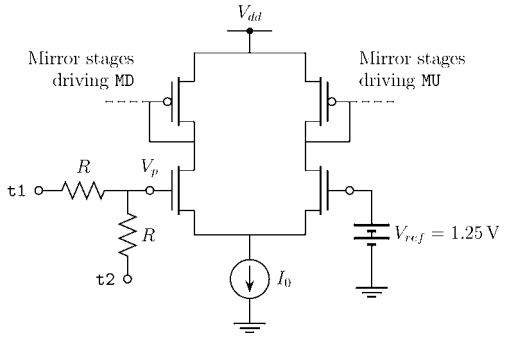

Fig. 13 *Control circuit for the Example driver 2*

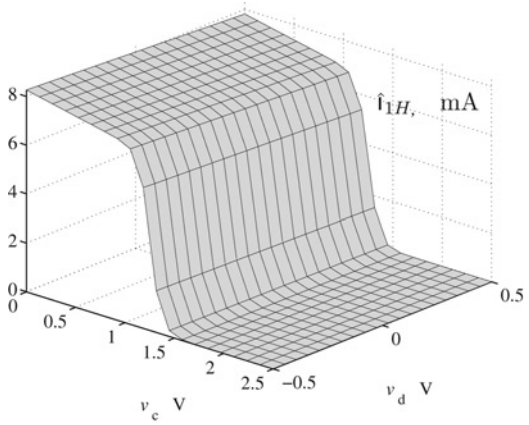

Fig. 14 *Static characteristic $\hat{\imath}_{1H}(v_d, v_c)$ for Example 2 driver forced in the HIGH logic state*

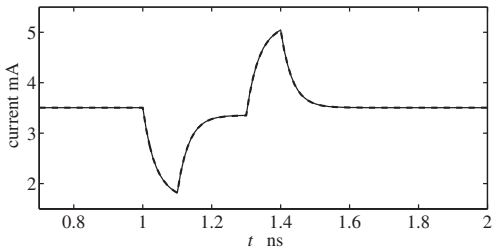

Fig. 15 *Load current computed for the test circuit of Example 2 (see text)*

Solid line: reference; dashed line macromodel.

worth noting that the VHDL-AMS implementation of the macromodel is much more efficient than the reference model but it turns out to be slightly less efficient than the alternate SPICE implementation. We believe this is due to the differences in the internal structure of simulators that use different solvers and algorithms for the transient analysis of circuits with nonlinear dynamic elements. Besides, in this comparison, the optimisation of the implementation of model equations for achieving the best results in terms of model efficiency has not been addressed. In spite of this, speed-up introduced by the VHDL-AMS implementation of the macromodel is very good and VHDL-AMS still offers the freedom of handling, within the same environment, both discrete-time and continuous-time events and allows the direct description of macromodel equations without any circuit interpretation required by a SPICE-type simulator.

4.2 Example 2

The second modelled device is an idealised version of the differential driver proposed in the work of Boni *et al.* [7], which exploits a control mechanism to reduce the fluctuations of the common mode voltage v_c around a reference voltage (e.g. 1.25 V). Here, the mechanism is implemented by the differential amplifier and current mirrors of Fig. 13, regulating the drain currents of MU and MD of Fig. 1. The probe voltage V_p is obtained by a high resistance ($R = 100$ kΩ) voltage divider connected to the output terminals of Fig. 1. In this paper, both the output stage of Fig. 1 and the control circuit of Fig. 13 are implemented in HSPICE and used as the reference model for Example 2.

Fig. 14 shows the static characteristic $\hat{\imath}_{1H}(v_d, v_c)$ for this device. According to the purpose of the control circuit, the variations of this characteristic against v_c are dominant, and, since $v_c = (v_1 + v_2)/2$, the usual simplification $\hat{\imath}_{1H}(v_1, v_2) = \hat{\imath}_{1H}(v_1)$ does not hold. The coefficients of the linear part are estimated as in Example 1, and their values are $\{\alpha_{11H}, \alpha_{12H}, \alpha_{21H}, \alpha_{22H}\} = \{-0.626, -0.063, 0.063, 0.626, 0.063\}$ pF.

The obtained model is tested by a simulation problem devised to highlight the differences introduced by the control mechanism and to assess the accuracy of the proposed model even for devices with enhanced features. The test circuit consists of the example driver forced in HIGH state and connected to a differential load composed of a 100 Ω resistor in series with an independent voltage source. The voltage source produces a pulse with 0.5 V amplitude and 100 ps transitions. The load current waveforms predicted by using the reference and the estimated models in such a test circuit by means of HSPICE are shown in Fig. 15. The good agreement of the curves confirms the ability of model (4) to describe differential drivers with control mechanism and highlights the importance of taking into account the dependence of the modelled currents on both output voltages.

5 Conclusions

This paper proposes a systematic procedure for the behavioural macromodelling of differential drivers for the assessment of SI and EMC effects in Gbps digital communication links. The proposed macromodels are mathematical relations reproducing the port behaviour of devices, thus hiding their internal structure and protecting the intellectual property of producers. They can be easily implemented in any circuit and mixed-signal simulation environment as SPICE-like subcircuits or via direct metalanguage code descriptions like VHDL-AMS. Besides, they can be included as external models within the widely adopted IBIS descriptions of digital integrated circuits. The modelling procedure is applied to the characterisation of actual devices, leading to macromodels that have been proven to be accurate and efficient enough for the simulation problem at hand.

6 Acknowledgments

The authors would like to thank Mentor Graphics for providing the ICX simulation tool and Sergio Perazza (Mentor Italy) for the ICX application test of Example 1. This work was supported in part by the Italian Ministry of University (MIUR) under a Program for the Development of Research of National Interest (PRIN Grant #2004093025).

7 References

1 Bloomingdale, C., and Hendrickson, G.: 'LVDS data outputs for high-speed analog-to-digital converters'. Analog Devices Technical Paper, 2001

2 'Electrical characteristics of low voltage differential signaling (LVDS) interface circuits'. ANSI/TIA/EIA-644, March 1996

3 'IEEE Standard for Low Voltage Differential Signals (LVDS) for Scalable Coherent Interface (SCI)'. IEEE std. 1596.3-1996, 1996

4 I/O Buffer Information Specification (IBIS) Ver. 4.1, January 2004. Available online at: http://www.eigroup.org/ibis/ibis.htm

5 Hegazy, H., and Korany, M.: 'IBIS modeling of LVDS buffers'. Mentor Graphics Technical Paper, November 2002

6 Muranyi, A.: 'Introduction to IBIS models and IBIS model making', Folsom, CA, 3–4 November 2003 (Intel course available at http://www.eda.org/pub/ibis/training/)

7 Boni, A., Pierazzi, A., and Vecchi, D.: 'LVDS I/O interface for Gb/s-per-pin operation in 0.35 μm CMOS', *IEEE J. Solid-State Circuits*, 2001, **36**, (4), pp. 706–711

8 Young, B.: 'An SOI CMOS LVDS driver and receiver pair'. Proc. 2001 Symp. VLSI Circuits, Kyoto, Japan, 14–16 June 2001, pp. 153–154

9 Gabara, T., Fischer, W., Werner, W., Siegel, S., Kothandaraman, M., Metz, P., and Gradl, D.: 'LVDS I/O buffers with a controlled reference circuit'. Proc. 10th Ann. IEEE Int. ASIC Conf. Exhibit, Portland, OR, USA, 7–10 September 1997, pp. 311–315

10 Stievano, I.S., Maio, I.A., and Canavero, F.G.: 'Parametric macromodels of digital I/O ports', *IEEE Trans. Adv. Packag.*, 2002, **25**, (2), pp. 255–264

11 Stievano, I.S., Canavero, F.G., and Maio, I.A.: 'On the behavioral modeling of integrated circuit output buffers'. Proc. 12th IEEE Topical Meeting on Electrical Perform Electron Packaging, EPEP, Princeton, NJ, 27–29 October 2003, pp. 281–284

12 Sjöberg, J., *et al*.: 'Nonlinear black-box modeling in system identification: a unified overview', *Automatica*, 1995, **31**, (12), pp. 1691–1724

13 'IEEE Standard VHDL Analog and Mixed-Signal Extensions'. IEEE std. 1076.1-1999, 18 March 1999

14 Christen, E., and Bakalar, K.: 'VHDL-AMS—a hardware description language for analog and mixed-signal applications', *IEEE Trans. Circuits Syst. II*, 1999, **46**, (10), pp. 1263–1271

15 Stievano, I.S., Maio, I.A., Canavero, F.G., and Siviero, C.: 'Behavioral macromodels of differential drivers with pre-emphasis'. Proc. 9th IEEE Workshop on Signal Propagation on Interconnects, Garmisch-Partenkirchen, Germany, 10–13 May 2005, pp. 129–132

16 Hagan, M.T., and Menhaj, M.: 'Training feedforward networks with the Marquardt algorithm', *IEEE Trans. Neural Netw.*, 1994, **5**, (6), pp. 989–993

17 Nguyen, D., and Widrow, B.: 'Improving the learning speed of 2-layer neural networks by choosing initial values of the adaptive weights'. Proc. Int. Joint Conf. on Neural Netw. (IJCNN), San Diego, CA, USA, 17–21 June 1990, pp. 21–26