2017 DESIGN AND VERIFICATION™ DVCON CONFERENCE AND EXHIBITION EUROPE

State-Space "Switching" Model of DC-DC Converters in SystemVerilog.

Elvis Shera, Methodology, Dialog Semiconductor, Nabern, Germany

elvis.shera@diasemi.com

***Abstract*—today systems are complex and they also need to come to the market quickly. This means that we need to have the right tools to describe the systems, from the feasibility phase down to the architectural exploration, modelling and towards the implementation. This is still a difficult area as it requires different kinds of models for the different phases and isolated simulation to prove the concept. The details are usually left to the implementation phase for different models but they are becoming important to be known very early. In this paper we are considering a modelling approach which is a good fit for the early system buildup, architectural exploration and it can be reused downstream towards the implementation because additional details will not have significant influence on simulation performance. Moreover, an enhanced model can still be pushed upwards to the system level for additional observation. SystemVerilog is both a design and modelling language, which more and more is being used for the modelling of analog circuits and at the system level. The advantage is that of being able to simulate all together within same simulation environment: RTL, SystemVerilog models, schematic models, or other models described in languages such as Verilog-A/MS. In particular the state-space approach is being discussed here, not in its typical averaged style, but in new nuances. i.e in a "switching" context which represents somehow the novelty part and which fits especially those applications such as switching converters which are difficult to simulate into details both at a block level and system level. This paper highlights why the model remains fast in simulation whilst still capable of showing details and being accurate and also why such an approach is a best fit for an event driven simulator.**

Keywords—architectural exploration, what-if analysis, realn number modelling, systemverilog, mixed-signal-simulation, event-driven simulator, dc-dc switching converters, state space modelling, user defined net.

I. INTRODUCTION

State space representation is a mathematical model of a physical system described as a set of input, output and state variables, related by first-order differential equations. State space models are well known in the literature and electronic circuits are a class of physical systems which can be described by such models. The predominant approach is the so called "averaged" model. Averaged models are a good fit for the spice-like simulators as the final implemented model contains one set of equations, containing the averaged behavior of the system over more configurations. Avoiding any switching is very critical in getting performance on any analog spice-like simulator, not only because of slow simulation but also because of increased possibility of convergence issues.

Averaged models though present some limitations as they do not show what really happens within the cycle. Examples are power dissipation and action taken from the controller due to the current on the inductor becoming negative. The average model needs to make assumptions on the load and line being met over a certain time interval. Assumptions which are not always met as we can clearly have the current going negative within the cycle but have a positive average value. Examples like this already pose a limitation in the adoption of the averaged model in a full development of the controller logic. Limitations which, as we are going to see in this paper, can be easily overcome due to the adoption of the "switching" state space model.

.

II. APPLICATION CASE – BOOST CONVERTER

A boost converter is a switching voltage regulator capable of regulating at the output a higher voltage than the input one. Figure 1, shows the output stage of a boost converter connected to the input port, typically a battery, and the output, the load.

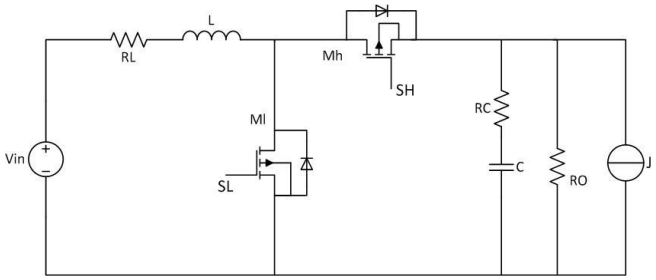

Figure 1. Boost Converter, Power Stage

In this case R_o and J can be seen as load with R_o being used to model loss of charge in the output capacitor. R_L, R_C represents here the parasitic resistor for the coil and the capacitor. The boost function then finally is decided from the way the gates of the transistors are being controlled. There are different schemes, depending on the architecture of choice but this is not in the scope of this paper. Since this is a highly nonlinear circuit we should consider linearizing this before getting into the state space model. Figure 2, shows a possible way of linearizing the previous circuit.

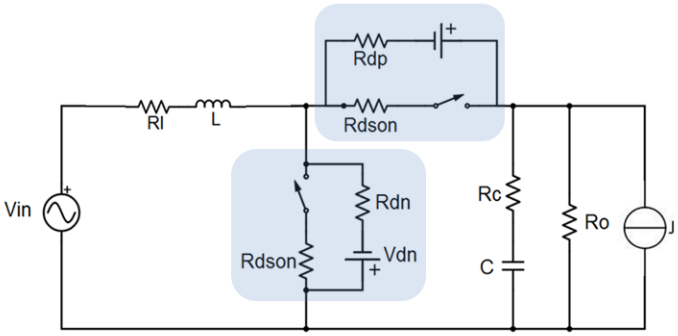

Figure 2. Boost Converter, Linearizing the Power Stage

In the linearization step, we have replaced the transistors with the relative R_{dson} resistors and an ideal switch. This is justified as such transistors are driven as close as possible to the ideal switch in order to avoid losses during the switching time. The body diodes have been replaced with a resistor R_d and a DC voltage source, V_d, in series with an ideal switch. Depending on the model, it is important to model the body effect diodes so that there is a way for the current in the coil to flow once the transistors are turned off. Parasitic elements need also to be considered if we are interested in efficiency calculations.

III. A QUICK OVERVIEW OF THE STATE SPACE MODEL

The state space basically says that given the state variables and the input at the time t_N it is possible to define the state of the system at time t_{N+1}, and therefore continue the process over any time interval. Unlike the frequency domain approach, the use of the state-space representation is not limited to systems with linear components and zero initial conditions. The internal state variables are the smallest possible subset of system

variables that can represent the entire state of the system at any given time. In electric circuits, the number of state variables is often, though not always, the same as the number of energy storage elements in the circuit such as capacitors and inductors and the state variables defined must be linearly independent, i.e., no state variable can be written as a linear combination of the other state variables or the system will not be able to be solved. The general state space representation is given from Equation 1.

$$\begin{aligned}\dot{\mathbf{x}}(t) &= A(t)\mathbf{x}(t) + B(t)\mathbf{u}(t)\\ \mathbf{y}(t) &= C(t)\mathbf{x}(t) + D(t)\mathbf{u}(t)\end{aligned} \qquad \text{Eq.1}$$

In case of linear time invariant system the matrices A, B, C, D are not time dependent therefore the representation can take the form of Equation 2.

$$\begin{aligned}\dot{\mathbf{x}}(t) &= A\mathbf{x}(t) + B\mathbf{u}(t)\\ \mathbf{y}(t) &= C\mathbf{x}(t) + D\mathbf{u}(t)\end{aligned} \qquad \text{Eq.2}$$

Where:

$$\dot{\mathbf{x}}(t) := \frac{\mathrm{d}}{\mathrm{d}\,t}\mathbf{x}(t).$$

As a generic example, Figure 3, shows an RLC circuit for which the state space model will be calculated as per Equation 2.

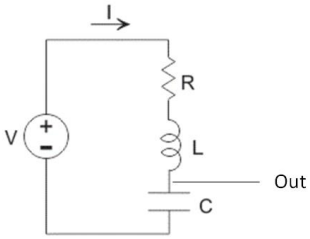

Figure 3. Example Circuit (RLC) for the state-Space model

Such a circuit gives rise to the below differential equation:

$$\begin{aligned}\dot{I}_L &= \frac{dI_L}{dt} = \frac{1}{L}V - \frac{R}{L}I_L + \frac{1}{L}V_C\\ \dot{V}_C &= \frac{dV_C}{dt} = \frac{1}{C}I_L\end{aligned} \qquad \text{Eq.3}$$

Which written in matrix form, looks like:

$$\begin{bmatrix}\dot{I}_L\\ \dot{V}_C\end{bmatrix} = \begin{bmatrix}\frac{-R}{L} & \frac{1}{L}\\ \frac{1}{C} & 0\end{bmatrix}\begin{bmatrix}I_L\\ V_C\end{bmatrix} + \begin{bmatrix}\frac{1}{L}\\ 0\end{bmatrix}V \qquad \text{Eq.4}$$

We have to remember that we can get a state space representation by following the transfer function method, but there are more possible equivalent representations. The ideal representation is what allows us to set initial conditions on the state variables where state variables are the current in the inductors and the voltages on the capacitors.

IV. BOOST CONVERTER STATE SPACE MODEL

To go from the linear circuit of Figure 2, to the state space model we should first remove the ideal switches. This can be done by analyzing each switch configuration and the circuit it will yield. Figure 4, shows more possible configurations where each configuration generates its own system of differential equation and therefore state space model. An important point here is that each state space model contains the same state variables. That together with the fact that matrices A, B, C and D are constants, means that we can switch from one model to the other in simulation without any change to the state variables themselves.

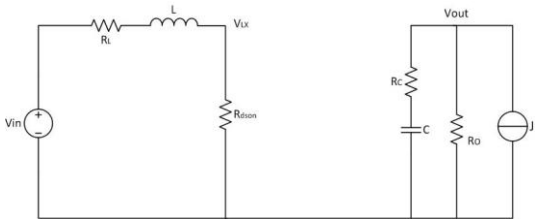

Figure 4a. Inductor charging phase

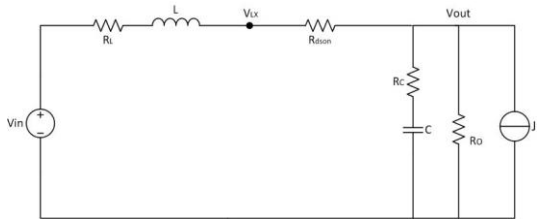

Figure 4b. Inductor discharge phase

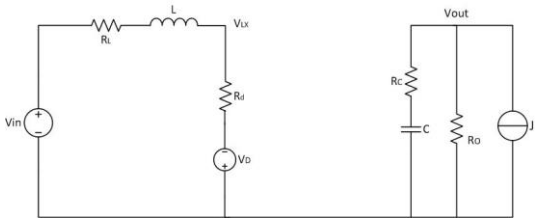

Figure 4c. Low side diode in conduction.

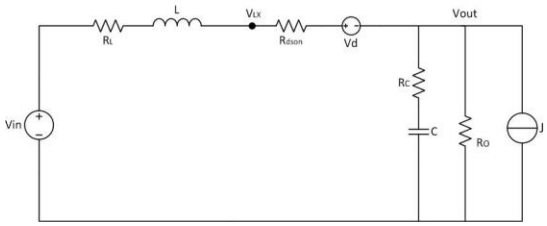

Figure 4d. High side diode in conduction.

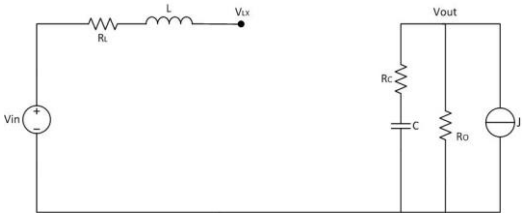

Figure 4e. Discontinuous mode of operation, no current in the coil.

Figure 4. Boost circuit configurations. Each configuration will generate an own set of equation therefore an own state space model.

Given that we have now Aa, Ab, Ac, Ad, Ae representing the A matrices of the equations of the above circuits and the same for the B, C and D matrices: the average model will be a model which has one A_{avg} matrix which is the average of the Aa, Ab, Ac, Ad, Ae matrices, weighted by the time the circuit will stay in the related configuration. Same can be done for B_{avg}, C_{avg} and D_{avg}. It is very easy to understand the average if we would consider only the 2 circuits in figure 4a and 4b reflected also from the coil current waveform in Figure 5.

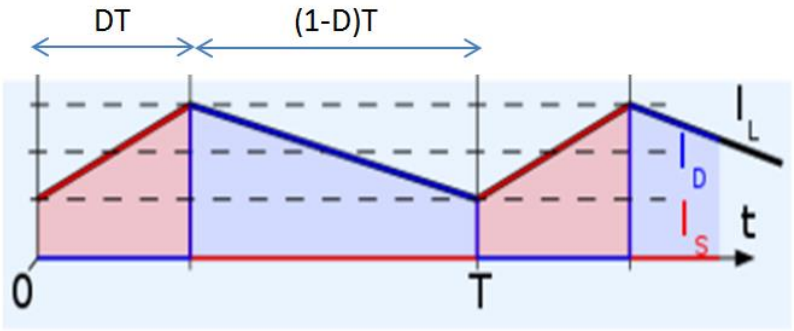

Figure 5. Current in a boost converter. Continuous conduction mode (CCM).

The equations of Figure 4a are valid only for DT time where T is the period and D is the duty-cycle. The boost will stay then in the configuration 4b for (1-D)T time. An average matrix will then be formed from:

$$A_{avg} = DA_a + (1-D)A_b$$
$$B_{avg} = DB_a + (1-D)B_b \qquad \text{Eq.4}$$

While averaging the two main active configurations is easy, averaging all of them is not that immediately obvious. There are modes in which the converter might stay in undefined time, as an example pulse skipping, so averaging is not trivial. As a general rule, we could say that an averaging model requires the knowledge of the duty-cycle information upfront, therefore all times this is unclear it will be difficult to average. Moreover, the duty-cycle and t_{ON}, t_{OFF} times are real numbers while in the reality the switches are controlled from digital signals toggling in a PWM fashion.

After such considerations, instead of trying to average, we could simply decide to use each of the equations in its own time slot/configuration. In such a way we can have as many configurations as we like and increase the use case of the model to all possible architectures. In addition we get for free the switching behavior and everything that is happening inside. The switching model it will be closer to the implementation of the controller of the DC-DC converter due to the PWM controller.

V. SystemVerilog Code

Figure 6, shows a SystemVerilog code snippet which implements the state space modelling as discussed above. An averaged model would be ok with a single set of equations. For a switching model, each circuit needs to have its own set of equations and the logic which is controlling the switching will be required.

```
bit clk; initial forever #(T*1s) clk = !clk;
var real a00, a01, a10, a11;
var real b00, b01, b10, b11;

always @(clk) begin
        // calculate the first derivatives
        dx0 = a00*x0 + a01*x1 + b00*J + b01*Vin;
        dx1 = a10*x0 + a11*x1 + b10*J + b11*Vin;
        // calculate the next state variables
        x0 = x0 + dx0*T;
        x1 = x1 + dx1*T;
end
```

Figure 6. Template example of SystemVerilog code for implementing the sate space model

VI. SIMULATION RESULTS

This section is presenting simulation results. The implemented model simulates in closed loop so an error amplifier and other component parts of the controller are part of the simulation although they are not the focus of this paper. Figure 7.a-g shows several simulation results for different modes of operation and a comparison when possible with the averaged state space model. In some of the figures details on the accuracy of the results are also highlighted.

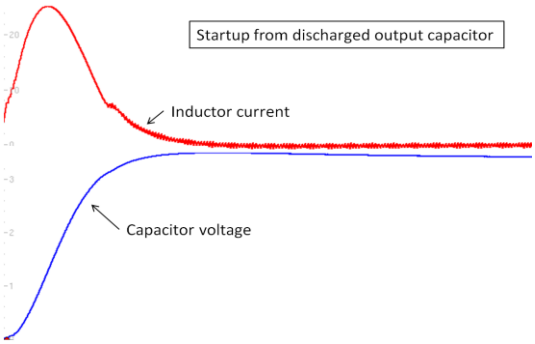

Figure 7a. Startup with no load current.

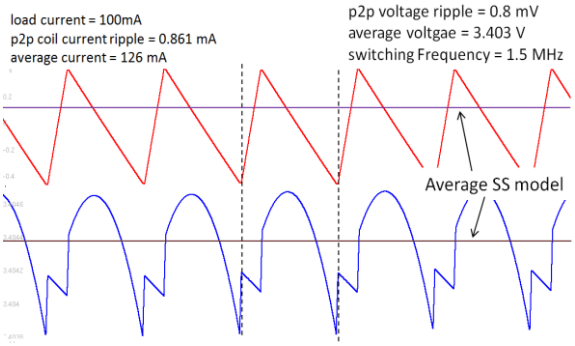

Figure 7b. CCM Steady state. Vin = 2.7V.

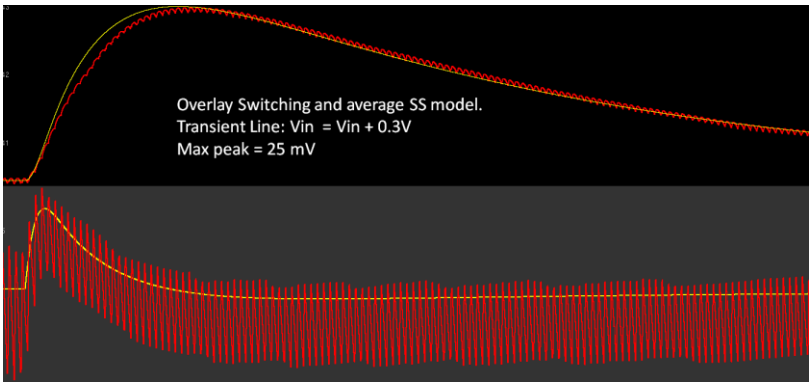

Figure 7c. Transient Line simulation and comparison between SS averaged and SS switching in CCM mode.

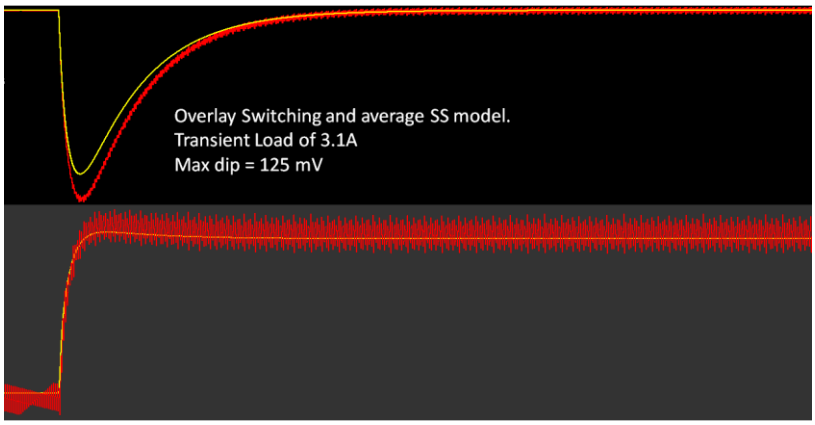

Figure 7d. Transient Load simulation and comparison between SS averaged and SS switching in CCM mode.

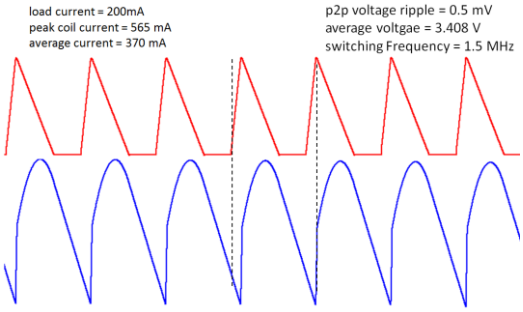

Figure 7e. DCM Mode. Steady state

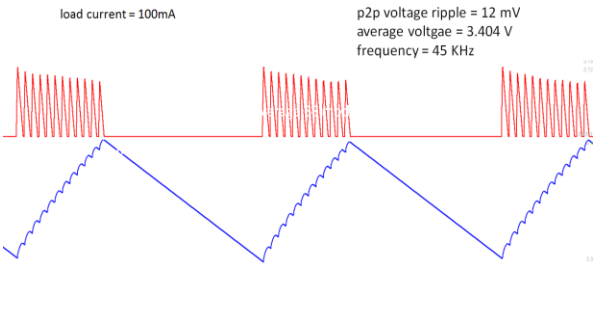

Figure 7f. Pulse Skip Mode of Operation. Asynchronous behavior.

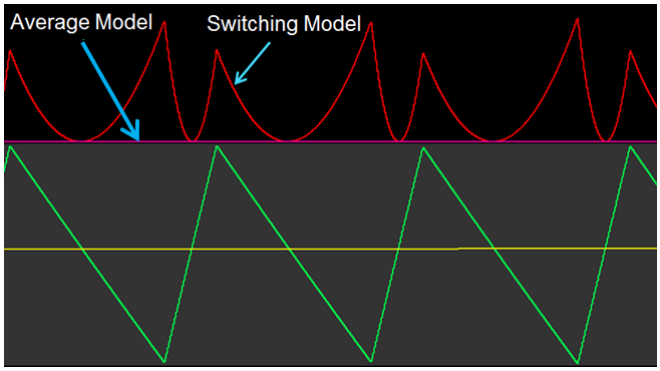

Figure 7g. Power dissipation in the inductor series resistor. Averaged Model will hide the losses happening within the cycle while with a switching model, calculation of the losses, in all modes of operation can be done in an easy manner. The overall efficiency can then be dynamically calculated and made available as all the other waveforms.

Figure 7. Simulation Results for Different Modes of Operation and comparison with the averaged Model

VII. Simulation Speed

The state space modelling approach is one of the fastest models to simulate. It is not affected from convergence issues and it only consists on algebraic operations such as multiplication and addition. The test used for measurements consists of several line and load transients in different configurations together with transients due to changes on the point of reference. Also body diode effects, DCM and pulse skip have been included in the tests. The calculation step used is 1ns. The wall clock time showed that it required 27s for 8.8ms of simulation; which given the details and accuracy is a very respectable result and makes the model a good fit for system level design and also for usage in typical digital regression and mixed-signal simulation. Compering with a schematic simulation and with all the rest of the controller (FSM, error amplifier, ramp generator, etc…) we have observed a gain factor of 8000 in the simulation time (two and half day for the corresponding schematic simulations). Note that the simulated model also contains a description of the controller with voltage and current loop which are out of the scope of this paper.

VIII. DISADVANTAGE

The main disadvantage of the presented method is that after spending some time in getting the equations right, the topology of the circuit might change which means that the equations need to be rewritten. In addition, the number of equation and complexity will depend from the numbers of component in the circuit. One way to mitigate this is via smart partitioning. The output stage of a DC-DC converter is quite stable, so in the presented example, once these equations are written and the model is done, it could be reused in many other scenarios.

IX. CONCLUSION

In this paper a semi-novel approach to state space models in an event driven simulator has been described. Simulation waveforms have been given for different test conditions and benchmark data concerning simulation speed have been presented. The model is a good fit for system level design and it can be reused during the implementation phase. Spice type simulator is slow due to the convergence issues that are found with the extremely non-linear devices (switches). Event driven simulator is used to "act" like a time step simulation. The fact that it is an even drive simulator is mainly meaningful in that the digital part of the circuit does not need to be simulated except for when an event occurs. That saves a lot of time. Finally efficiency calculations have been performed and it has been highlighted how the losses happening within the cycle are not considered from an average model in an accurate manner. The switching model enables more accurate calculation of the efficiency whilst simulating and makes it available dynamically in the different modes for which the DC-DC converter operates.

ACKNOWLEDGMENT

I would like to thank John Kesterson, Frederic Choquet and Giovanni Tarroboiro for their expert advice on the presented modelling approach, for reviewing the simulation results and this paper. Without their collaboration and encouragement this work would not have been possible.

REFERENCES

[1] Seth R. Sanders, J. Mark Noworolski, Xiaojun Z. Liu, George C.Verghese, Generalized Averaging Method for PowerConversion Circuits. IEEE Power electronic Specialists Conf. San Antonio, June 1990.

[2] R.D. Middlebrook and S. Cuk, "A General Unified Approach to Modeling Switching Power Converter Stages,"IEEE PESC Record, 1976, pp. 18-34.

[3] N. Mohan, T. M. Undeland and W. P. Robbins, "Power Electronics, Converters, Applications, and Design, " John Wiley & Sons, 2003.

[4] Christopher P. Basso, "Linear Circuit Transfer Functions", John Wiley & Sons. 2016

[5] V. Vorperian, Fast analytical techniques for Electrical and Electronics Circuits, "Cambridge University Press, ISBN 0-521-62442-8, 2004.

[6] H. Jin, "Behavior-mode simulation of power electronic circuits," IEEE Trans. on Power Electronics, pp.443-452, Mar. 1997.

[7] E. Dijk, et al., "PWM-switch modeling of DC-DC converters," IEEE Trans. on Power Electronics, pp.659-665, Jun. 1995.

[8] R. B. Staszewski, et al., "Event-driven simulation and modeling of phase noise of an RF oscillator," IEEE Trans. on Circuits and Systems I, pp.723-733, Apr. 2005.

[9] J.-E. Jang, et al., "True event-driven simulation of analog/mixedsignal behaviors in SystemVerilog: a decision-feedback equalizing (DFE) receiver example," IEEE Custom Integrated Circuits Conf. (CICC), pp.1–4, Sep. 2012.

[10] H.-P. Le, et al., "Design techniques for fully integrated switchedcapacitor DC-DC converters," IEEE J. of Solid-State Circuits, pp.2120-2131, Sep. 2011.

[11] M. Ierssel, et al., "Event-driven modeling of CDR jitter induced by power-supply noise, finite decision-circuit bandwidth, and channel ISI," IEEE Trans. on Circuits and Systems I, pp.1306-1315, Jun. 2008.