An Event-Driven Simulation Methodology for Integrated Switching Power Supplies in SystemVerilog

Ji Eun Jang, Myeongjae Park, and Jaeha Kim

Department of Electrical Engineering and Computer Science, Inter-university Semiconductor Research Center

Seoul National University, Seoul, Korea

jieunjang@mics.snu.ac.kr, mjpark@mics.snu.ac.kr, jaeha@snu.ac.kr

ABSTRACT

Emerging power-supply-on-chip applications such as on-chip DC-DC conversion, energy harvesting, and LED drivers use switching regulator ICs integrated with digital controllers. Although the resulting mixed-signal systems call for efficient system-level behavioral simulation, this remains difficult due to the fast switching and slow transients of the regulator and the high complexity of the controller. This paper presents a truly event-driven approach for modeling and simulating such integrated power systems entirely in SystemVerilog. By modeling various switching regulator topologies as switched linear networks whose responses can be expressed as a sum of complex exponentials, $ct^{m-1}e^{-at}u(t)$, the accurate voltage/current waveforms can be captured by updating the coefficients, c, at each input or switching event. The model is applied to two examples, a power factor corrector and switched-capacitor DC-DC converter, and the results demonstrate that the proposed simulator can achieve 20~100× improvements in speed while maintaining SPICE-level accuracy in evaluating power efficiency, steady-state ripples, and power factor.

Categories and Subject Descriptors

EDA6.3 [Analog Design and Simulation] Analog, mixed-signal, and RF simulation.

General Terms

Algorithms, Design, Verification.

Keywords

Event-driven simulation, Behavioral modeling, Switching-mode power supplies, SystemVerilog.

1. INTRODUCTION

The design and verification of state-of-the-art switching-mode power supplies demands advanced methodologies for mixed-signal IC systems. This is because a large portion of these switching regulators is now being realized in ICs to minimize their power, cost, and size when used in various emerging applications such as energy harvesting systems [1], LED drivers [2], and on-chip power management systems [3]. These so-called

power-supplies-on-chip [4] have brought two notable changes to the previous discrete-component-based switching regulator designs: first, complex converter topologies such as time-interleaved architectures are frequently used because the cost is no longer set by the component count, and second, many controllers are digitally implemented to support multiple operating modes and maintain high efficiency in different conditions. Hence, the resulting switching-mode power supplies typically comprise a complex mixed-signal system with an analog power converting stage (e.g. buck or boost) and a digital controller. This paper presents an efficient, event-driven behavioral simulator suited for these integrated switching power supplies.

For accurate performance estimation and complete functional verification of switching-mode power supplies, a behavioral simulator should be fast at handling large digital circuits while accurately simulating analog signals. However, the existing simulation tools do not meet these requirements. For instance, SPICE, a versatile circuit simulator capable of nonlinear transient simulation, is not the best solution for simulating integrated switching power supplies, as it cannot achieve the required speed and accuracy [5]. The problem is that switching supplies are characterized by high-frequency switching activities that demand fine-grained simulation for accuracy, but also by slow transients that require long simulation times. The large transistor counts in the digital components further aggravate the situation.

To address this problem, several approximate models, such as average models and sampled-data models, have been proposed [6,7]. The average models capture only the slow-moving transients of the system dynamics and neglect the fast-moving transients due to switching. Similarly, the sampled-data models capture only the cycle-to-cycle behavior of the system by dealing only with the samples at a specific time instant in each switching period. A device-level average model is another alternative [8], with which it is possible to compose an average model of an arbitrary converter topology and simulate it directly in SPICE. The pulse-width-modulated switch model is such an example [9]. While these approximate models can significantly improve the simulation time by simplifying the system to a non-switching, time-invariant one, a critical shortcoming of these models is that they cannot model the behavior or estimate the performance related to the high-frequency switching activity of the regulator, such as the conversion efficiency and power factor. For instance, although the magnitude of the output voltage ripples is one of the key factors influencing the power efficiency, it is not well modeled by the average or sampled-data models.

This paper presents a fast, event-driven simulation methodology to accurately simulate the high-frequency switching behavior of integrated switching power supplies. Event-driven simulation is a preferred solution because the simulation cost only increases with the system activities and not with the system size. However, until

recently, there were no good methods for simulating analog signals in an event-driven way. For instance, an event-driven oscillator model [10] and an analog filter model based on a look-up table [11] have been reported, but these techniques share the common limitation that they can be applied only to specific types of systems or inputs.

The event-driven approach presented in this paper builds on the method described in [12], whose key idea is to express an analog waveform as a linear combination of basis functions. The approach was demonstrated in a system containing linear time-invariant (LTI) analog components, whose output signals can always be expressed as a series of complex exponentials. With this signal representation, it was shown that the coefficients in the output series need to be updated only when those in the input series change, thus enabling a truly event-driven simulation. In addition, a series of complex exponentials is very expressive in representing various analog signals encountered in circuit applications, including steps, ramps, sinusoids, exponentials, and all of their combinations. Nonetheless, the approach in [12] cannot be directly applied to the modeling of switching-mode power supplies as it assumes that each analog block has a fixed linear transfer function over time.

This paper extends the event-driven simulation method described in [12] to switched linear systems, which include the majority of the switching-mode power supply topologies. This is achieved by allowing the system to change its input-to-output transfer function, while preserving previously-stored state values such as the currents through inductors or voltages across capacitors. To enable this, the states of the energy-storing elements are made explicit in the system formulation so that their initial values can be supplied as additional inputs to the system. As in [12], the proposed simulation method is implemented on a single logic simulation platform of SystemVerilog so that both the analog and digital components can be simulated in an event-driven fashion.

The rest of this paper is organized as follows. Section 2 shows that a general switching-mode power supply can be modeled as a switched linear system, using a boost converter as a working example. Section 3 describes the proposed event-driven simulation method for switched linear systems and Section 4 discusses its implementation in SystemVerilog. Finally, in Section 5, the proposed method is demonstrated with two power converter examples, a power factor correction (PFC) boost converter [13] and a time-interleaved switched-capacitor DC-DC converter [14].

2. SWITCHED LINEAR SYSTEM MODEL FOR SWITCHING POWER SUPPLIES

A common characteristic of switching regulators is that the circuit configuration changes depending on the switching phases. For example, a boost converter has two operation phases alternated by a switch connection as shown in Fig. 1(a). With the switch in position 1 (phase 1), the right-hand side of the inductor is connected to the ground, resulting in the network shown in Fig. 1(b). With the switch in position 2 (phase 2), the inductor is connected to the output, leading to the circuit shown in Fig. 1(c).

In this work, we model such switching regulators as a switched linear system rather than as an average linear system or sampled-data system. Even though the input/output relationship changes at each switching instant, the circuit can be modeled as a linear time-invariant system within each operation phase. For example, the relationship between the input $v_{IN}(t)$ and output $v_{OUT}(t)$ of the
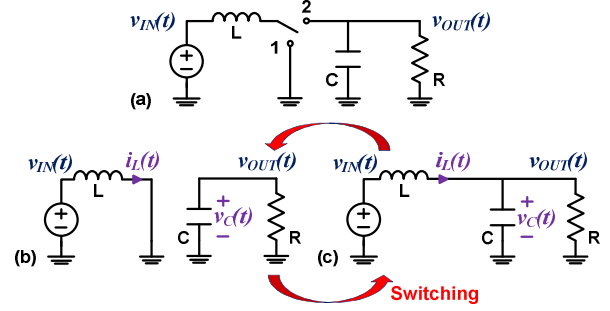
Figure 1. (a) A boost converter circuit and its linear system model in (b) switching phase 1 and (c) switching phase 2.

boost converter in phases 1 and 2 can be modeled as a set of differential equations listed below:

For phase 1:

$$\begin{cases} \frac{dv_C(t)}{dt} & = -\frac{1}{RC}v_C(t) \\ \frac{di_L(t)}{dt} & = \frac{1}{L}v_{IN}(t) \\ v_{out}(t) & = v_C(t) \end{cases}$$

For phase 2:

$$\begin{cases} \frac{dv_C(t)}{dt} & = \frac{1}{C}i_L(t) - \frac{1}{RC}v_C(t) \\ \frac{di_L(t)}{dt} & = \frac{1}{L}v_{IN}(t) - \frac{1}{L}v_C(t). \\ v_{OUT}(t) & = v_C(t) \end{cases} \tag{1}$$

This set of differential equations can be converted to a Laplace s-domain equivalent, using the Laplace transformation formula for a function derivative in Eq. (2), resulting in Eq. (3). This Laplace transformation is used to apply the event-driven simulation method in [12], which is explained in more detail in the next section. Note that the initial conditions of the internal state variables v_C and i_L are made explicit in the s-domain equations. In Eq. (3), the capital letters denote s-domain signals while the italic letters denote their initial conditions in the time domain.

$$\mathcal{L}\left\{f^n(t)\right\} = s^n \mathcal{L}\left\{f(t)\right\} - \sum_{k=1}^{n} s^{k-1} f^{n-k}(0) \tag{2}$$

For phase 1:

$$\begin{cases} V_C(s) & = \frac{RC}{sRC+1}v_C(0) \\ I_L(s) & = \frac{1}{sL}V_{OUT}(s) + \frac{1}{s}i_L(0) \\ V_{OUT}(s) & = V_C(s) \end{cases}$$

For phase 2:

$$\begin{cases} V_C(s) & = \frac{V_{IN}(s)+sLC\cdot v_C(0)+L\cdot i_L(0)}{s^2LC+sL/C+1} \\ I_L(s) & = \frac{(sC+1/R)V_{IN}(s)-C\cdot v_C(0)+(sLC+L/R)i_L(0)}{s^2LC+sL/C+1} \\ V_{OUT}(s) & = V_C(s) \end{cases} \tag{3}$$

As this simple boost converter example shows, the main difference between this switched linear system model and a linear time-invariant system model is that the former requires additional terms related to the initial conditions of the energy-storing elements in the circuit. In Eq. (3), for instance, $v_C(0)$ and $i_L(0)$ represent the initial voltage across the capacitor and the initial current through the inductor, respectively. These terms in Eq. (3) model how the stored energy in the circuit affects the output responses after $t = 0$. Each time a switch occurs, the final state (of

phase 1 or phase 2) becomes the initial state of the next phase (phase 2 or phase 1, respectively), thus preserving the energy stored within the system.

3. TRUE EVENT-DRIVEN SIMULATION OF SWITCHED LINEAR SYSTEMS

This section describes how the above switched linear system model can be simulated in a truly event-driven fashion. The main difficulty in performing event-driven simulations for analog circuits stems from the fact that their output response can continuously change even when there is no change in the input. If the output signal is represented by a set of data points, a one-time update (equivalent to one data point) cannot capture this continuous waveform. The method proposed in [12] addressed this problem by representing an analog signal with a linear combination of basis functions, $c \cdot t^{m-1} e^{-at} \cdot u(t)$, and updating the coefficients, c, when the input changes. For instance, considering a first-order filter with a step input, $c_{in}u(t)$, its output is known to be in the analytical form of $c_{out}u(t) - c_{out}e^{-\omega_p t}u(t)$, where ω_p is its pole frequency (see Fig. 2). When the input coefficient, c_{in}, changes, the corresponding change in the output only affects the output coefficient, c_{out}. Therefore, it is sufficient to update the output only once at the time of the input change.

Note that this method also computes the output waveform analytically in the s-domain without involving time integration, provided that both the input signal and the system transfer function are expressed in the s-domain representation after the Laplace transformation. Again, for the first-order filter example, the output signal of the filter in the s-domain is nothing but a product of a step function, $1/s$, and a filter transfer function, $\omega_p/(\omega_p+s)$, which can be decomposed to the sum of a step and an exponential functions. The two main advantages of this method are that fast simulation is possible because the output is explicitly calculated without solving an ordinary differential equation, and its accuracy is virtually infinite because the signal is represented as a function, not as a data point.

This work extends the method described in [12] to switched linear systems by adopting a signal representation with a series of complex exponentials and the s-domain output computation method. There are two main differences between our method and that in [12]. First, in addition to the input events, the switching event can also trigger an output event. Second, while the output of a time-invariant system can be determined only by an input and its input-to-output transfer function, a switching system additionally requires the information on the initial states and its relationship (transfer function) to the system dynamics, as explained in Section 2.
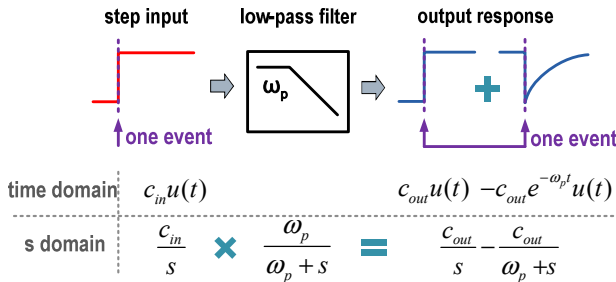

Figure 2. A simple analog filter example to demonstrate the event-driven simulation method for linear time-invariant (LTI) systems proposed in [12].
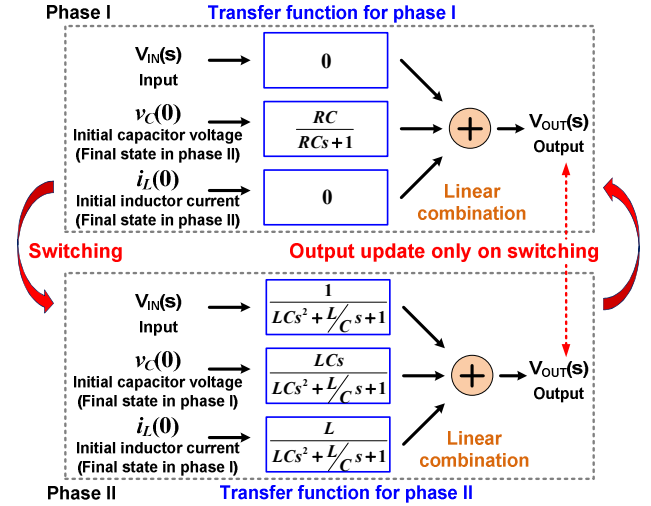

Figure 3. The proposed method applied to the boost converter example in Section 2.

Based on these differences, our proposed event-driven simulation methodology can be summarized as follows: 1) model a switched linear system with multiple transfer functions, each of which describes the circuit network during each switching phase, 2) when the switching occurs, sample the initial states of the reactive elements, and 3) compute the output by summing the effects from both the input signal and the initial states of the reactive elements. Fig. 3 illustrates the presented method applied to the boost converter example described in Section 2. For each phase, the three transfer functions define the relationship between the input (V_{IN}), the initial capacitor voltage $(v_C(0))$, the initial inductor current $(i_L(0))$, and the output (V_{OUT}). Every time the circuit switches between phases, the initial capacitor voltage and the inductor current are sampled and their transfer functions are redefined according to Eq. (3). Then, just as in the LTI system described in Fig. 2, the output voltage induced by each of the input and initial conditions can be evaluated simply by multiplying each one by its corresponding transfer function. The zero transfer gains for V_{IN} and $i_L(0)$ in phase 1 imply that the output is not related to the input and the inductor current, as they are disconnected from the output during this phase. When the circuit switches to the next phase, however, V_{IN} and the energy stored in the inductor start to increase V_{OUT} again. Finally, the output is updated as a linear combination of these three components.

4. IMPLEMENTATION IN SYSTEMVERILOG

This section outlines the implementation of the proposed event-driven simulation methodology in SystemVerilog. SystemVerilog was chosen as our simulation platform as it can serve as a true event-driven simulation engine that seamlessly integrates analog and digital models. A further advantage of SystemVerilog over other HDL standards is that it offers a composite data type, *struct,* and hence the set of multiple parameters can be exchanged between the block modules as though they are a single bundled signal. We adopt the digital signal data type defined in [15] ("XBIT") and the analog signal data type defined in [12] ("XREAL"). The digital data type named XBIT has two member variables: *value,* and *t_offset. Value* is a digital data bit and *t_offset* is a real-valued variable that indicates the actual time instant of the event. The analog data type named XREAL comprises three variables: *param_set, t_offset,* and *flag.*

```
module boost_converter(
   input XBIT switching,
   input XREAL in,
   output XREAL out);

   // transfer functions in phase 1
   chandle TF_in_out_ph1;
   chandle TF_vc0_out_ph1;
   chandle TF_il0_out_ph1;
   // transfer functions in phase 2
   chandle TF_in_out_ph2;
   chandle TF_vc0_out_ph2;
   chandle TF_il0_out_ph2;

   always @(switching) begin
      vc0=sample(vc);   // sampling initial states
      il0=sample(il);

      if (phase1)        // evaluating outputs
         out.param_set = multiply(in,TF_in_out_pout_ph1)
                      + multiply(vc0,TF_vc0_out_ph1)
                      + multiply(il0,TF_il0_out_ph1);
      If (phase2)
         out.param_set= multiply(in,TF_in_out_ph2)
                      + multiply(vc0,TF_vc0_out_ph2)
                      + multiply(il0,TF_il0_out_ph2);

      out.t_offset = switching.t_offset;
      -> out.flag;        // output triggering
   end
endmodule
```

Figure 4. Pseudo-code in SystemVerilog for the boost converter example in Section 2.

Param_set is a set of coefficients for expressing the analog waveforms, and *flag* is an event variable that indicates whether the event change has happened for the signal in the current time step, which is generally used within *always* statements.

An outline of a boost converter model in SystemVerilog is given in Fig. 4. The input and output signals are defined as XREAL, while the switching input signal is XBIT. Six transfer functions, three for each of the two switching phases, are given as a C-pointer handle pointing to a linked list in C that stores information on the poles, zeros, and gain of the system. The *always* statement within the module is triggered when the circuit switches between phases. The initial states of the capacitors and inductors are then sampled and the *param_set* of output XREAL signals is updated according to the current input and sampled initial conditions. The *multiply()* function is a DPI function written in C that performs s-domain multiplications of XREAL signals and transfer functions. As the output update is aligned in time with a switching event, the *t_offset* of the output has the same value as the switching. Once the *param_set* and *t_offset* outputs are updated, the event variable *out.flag* is triggered, thus notifying subsequent blocks of the change event.

5. EXPERIMENTAL RESULTS

5.1 POWER FACTOR CORRECTION (PFC) BOOST CONVERTER

The speed and accuracy of the event-driven simulation method are demonstrated using the example of a power factor correction circuit (PFC) composed of a bridge-diode rectifier and a boost converter (Fig. 5). The power factor is one of the key performance metrics of AC-DC power converters required by many regulatory standards. It is defined as in Eq. (4), which expresses the ratio of
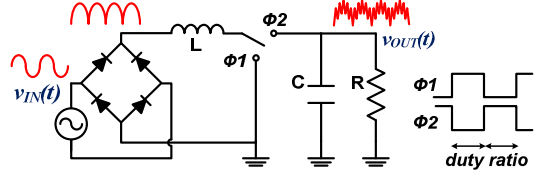
Figure 5. A power factor correction boost converter.

the real power flowing to the load and the apparent power in the circuit:

$$power\ factor = \frac{average\ power}{(rms\ voltage)(rms\ current)}. \qquad (4)$$

For a high power factor, the circuit should basically behave as a pure resistive load. A boost converter is a widely used topology for power factor correction circuits because the switched inductor at the input conducts a current that is proportional to the input voltage with very low harmonics [13].

One difficulty in simulating such an AC-DC power converter is that there is a big gap between the input AC frequency and the switching frequency of the boost converter. For instance, in most applications, the input source is 50~60-Hz 110~220V AC power, while the switching frequency is typically in the 100kHz~1MHz range. The average behavior analysis mentioned in the introduction is not suitable here because accurate measurement of the power factor requires detailed information on the input voltage/current waveforms, such as ripples. The required simulation time is long, typically several tens of milliseconds, to simulate a few cycles of the 60-Hz AC input.

Fig. 6 illustrates the accuracy of the waveforms simulated by the proposed event-driven simulation method. Fig. 6(a) is the simulated output voltage of the boost converter, $v_{OUT}(t)$, for one 60-Hz input cycle. The zoom-in waveforms in Figs. 6(b) and (c), simulated with the proposed method and HSPICE, respectively, demonstrate that they are well matched, and illustrate the switching ripples of the converter. It is noteworthy that HSPICE requires many data points to express the switching ripples (marked by the blue dots in Fig. 6(c)) while our event-driven method generates only two events per switching cycle, as indicated by the arrows in Fig. 6(b). The power factor can be measured from the simulated input current waveform, which has a similar level of accuracy to SPICE (not shown). The comparison between the simulated power factors as a function of switching frequency and duty cycle in Fig. 7 confirms that the proposed method indeed achieves the same level of accuracy as SPICE.

The proposed event-driven simulator demonstrates significant improvements in speed compared with HSPICE, yet retains the equivalent accuracy. On a Linux machine with an AMD Phenom II X4 945 processor, the total execution time to simulate a 0.1-second period with a 100-ns time step is 8.2 seconds. Under the same conditions, the HSPICE simulation takes 920.5 seconds, which is 110× slower.

The execution time of the proposed method varies weakly with the time step. For instance, when the time step is reduced from 100-ns to 10-ps (1/10,000X reduction), the execution time of the proposed method increases by only 15% (from 8.2 to 9.4 seconds), while that of HSPICE increases by 15000%. The reason the execution time hardly varies is that the number of switching events within the 0.1-second period remains the same regardless of the time step, which confirms that the proposed simulation
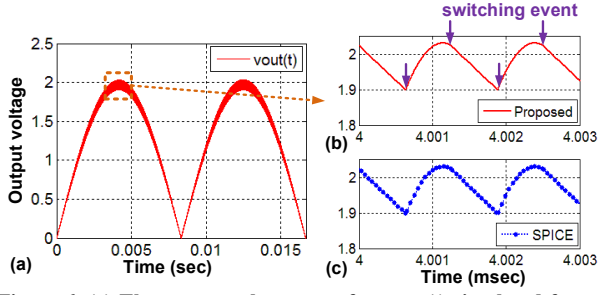
Figure 6. (a) The output voltage waveform $v_{out}(t)$ simulated for one 60-Hz input cycle, (b) 5000× zoom-in view of $v_{out}(t)$, (c) $v_{out}(t)$ simulated by HSPICE.
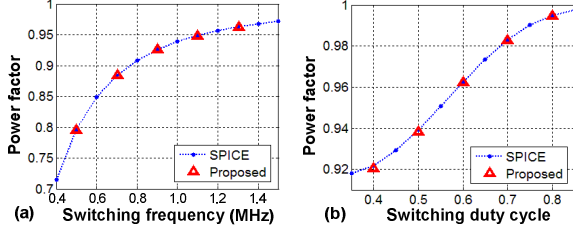

Figure 7. The comparison of the simulated power factors vs. (a) frequency and (b) switching duty cycle.
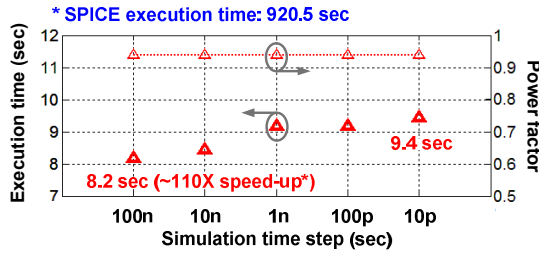

Figure 8. Execution time and simulated power factor vs. time step.

indeed operates in a purely event-driven fashion. This remarkable speed-up does not incur any penalty in accuracy, as the power factor measured using the proposed method is virtually constant, independent of the time step (Fig. 8).

5.2 TIME-INTERLEAVED SWITCHED-CAPACITOR DC-DC CONVERTER

For the second example, we choose the multi-phase time-interleaved switched-capacitor (TI-SC) DC-DC converter described and analyzed in [14]. The switched-capacitor DC-DC converter topology is becoming a common choice for power-supplies on chips, as the IC technology is more amenable to integrating high-density capacitors than low-loss inductors. One difficulty in simulating a TI-SC DC-DC converter is that the number of switching activities increases with the number of time-interleaving phases.

Our TI-SC converter example is composed of N interleaved 2:1 step-down converter units, as shown in Fig. 9(a). The total capacitance is divided into a set of small units and the switching is controlled by a set of N equally spaced clocks (Φ1, …, ΦN). Fig. 9(b) illustrates the basic operation of a 4-phase TI-SC converter with the output waveforms. The output voltage ripple is inherent in a switched-capacitor converter, and generally decreases as the switching frequency increases. Time-interleaving is an alternate way of reducing the ripples without increasing the switching frequency.

The TI-SC converter in Fig. 9 can be modeled as in Fig. 10, where the unit capacitors, C_{fly}, switch their configurations between a series and parallel connections depending on the controlling clock phase (Φ). The model includes the on-resistance of the switches, R_{sw}, and parasitic top- and bottom-plate capacitances, C_{par}, to account for the conduction and switching losses, respectively. To simplify the model, the top- and bottom-plate capacitances are combined into a single capacitor because they experience approximately the same voltage swings in steady states [14]. The s-domain transfer function of each phase can be derived from Eq. (5), where $v_{cap}[i]$ denotes the voltage across the i-th unit capacitor.

$$V_{out}(s) = \frac{\dfrac{s}{2R_{sw}C_{par}}V_{in}(s) + \dfrac{N_S}{4R_{sw}^2 N_S C}v_{out}(0) + \dfrac{s}{2R_{sw}C_{par}N_S}v_{cap}(0)}{\left\{ s^2 + s\left(\dfrac{R_L N_S C_{par} + 2R_{sw}C + NCR_L}{2R_{sw}CR_L N_S C_{par}} \right) + \dfrac{1}{2R_{sw}CR_L N_S C_{par}} \right\}}$$

where $\displaystyle v_{cap}(0) = \sum_{parallel} v_{cap}[i](0) - \sum_{series} v_{cap}[i](0)$

$$V_{cap}[i](s) = \begin{cases} \text{for series capacitor:} \\ \dfrac{1}{2R_{sw}Cs+1}V_{in}(s) + \dfrac{1}{2R_{sw}Cs+1}V_{out}(s) + \dfrac{2R_{sw}C}{2R_{sw}Cs+1}v_{cap}[i](0) \\ \text{for parallel capcitor:} \\ -\dfrac{1}{2R_{sw}Cs+1}V_{out}(s) + \dfrac{2R_{sw}C}{2R_{sw}Cs+1}v_{cap}[i](0) \end{cases}$$

(5)

Fig. 11(a) plots the power efficiency of the TI-SC converter as a function of the number of time-interleaving phases, N, and compares the results produced by HSPICE and the proposed method. When the total amount of charge delivered to the load is fixed with N, the output voltage ripple initially decreases by a factor of N, as the amount of charge delivered per clock transition
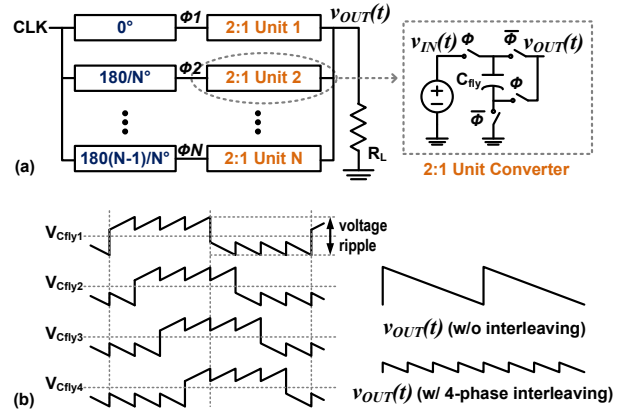

Figure 9. (a) N time-interleaved 2:1 step-down switched-capacitor DC-DC converter, (b) the waveforms of its internal capacitor voltages and final output voltage when N=4 [14].
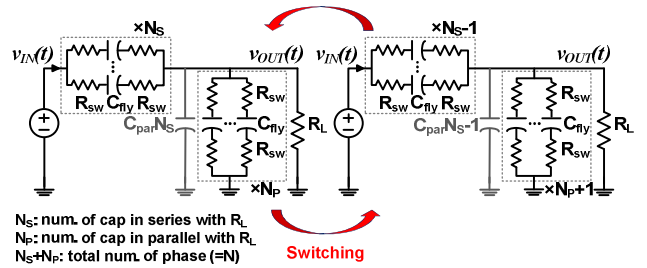

N_S: num. of cap in series with R_L
N_P: num. of cap in parallel with R_L
N_S+N_P: total num. of phase (=N)

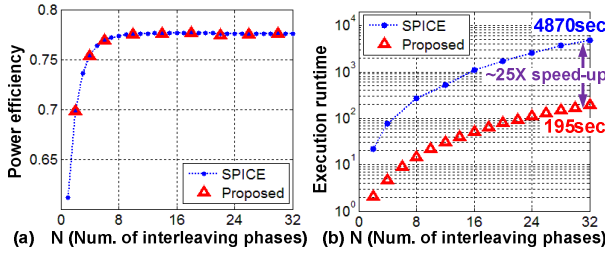Figure 10. Switched linear circuit model of an N time-interleaved, 2:1 step-down TI-SC converter.

Figure 11. (a) The simulated power efficiency of the TI-SC converter and (b) execution time vs. the number of time-interleaving phases (N).
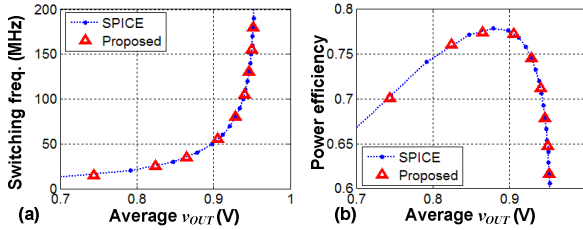

Figure 12. The simulated (a) switching frequency and (b) power efficiency of the TI-SC converter vs. the average output voltage.

is smaller. Therefore, the better power efficiency can be achieved with a higher N. However, increasing N above a certain value produces diminishing returns because the other losses, such as the conduction loss of the switches and the switching loss of the parasitic capacitors, increase. Fig. 11 illustrates this tendency, and the simulation results of the proposed method match well with the SPICE simulation results.

The improvement in speed with the proposed method is moderate compared with the boost converter case, as SPICE is better at simulating switched capacitors than inductors. When simulated on a Linux machine with an AMD Phenom II X4 945 processor, the proposed method shows a ~20X overall speed improvement compared with the HSPICE simulation (see Fig. 11(b)).

The proposed method predicts the well-known dependencies of the switching frequency and power efficiency for the output voltage v_{OUT}, described in [14]. Fig. 12 verifies the accuracy of the obtained results against those of HSPICE. Fig. 12(a) plots the switching frequency vs. the average v_{OUT} and Fig. 12(b) plots the power efficiency vs. the average v_{OUT} when v_{IN} is 2V and N=16. The v_{OUT} dependency on the switching frequency is similar to the IR-drop phenomenon in linear regulators, in which the output voltage drops when the load current is higher than the current that the TI-SC converter can nominally supply. As a result, slower switching leads to a lower average v_{OUT} and lower power efficiencies (Fig. 12). Nonetheless, an excessively high switching frequency is also undesirable as the power efficiency can be degraded due to the loss in the switching capacitors.

6. CONCLUSIONS

This paper presents an event-driven simulation methodology for switching regulators. A common characteristic of various switching regulators such as buck, boost, and switched-capacitor converters is that they can all be modeled as switched linear systems. By extending the previously published event-driven

methodology for LTI systems, the proposed method realizes a truly event-driven simulation of these switched linear systems, in which the output response is updated only once with each input change or switching event. Two power converter examples simulated in SystemVerilog demonstrate that the proposed method achieves SPICE-level accuracy with 20~100× faster simulation speeds. The experimental results also show that the simulation accuracy and execution time are not influenced by the simulation time step, thus confirming that the proposed method achieves a truly event-driven simulation.

7. ACKNOWLEDGMENTS

8. REFERENCES

[1] G. K. Ottman, et al., "Adaptive piezoelectric energy harvesting circuit for wireless remote power supply," *IEEE Trans. on Power Electronics*, pp.669-676, May 2002.

[2] H. Broeck, et al., "Power driver topologies and control schemes for LEDs," *IEEE Applied Power Electronics Conf. (APEC)*, pp.1319-1325, Mar. 2007.

[3] G. Patounakis, et al., "A fully integrated on-chip DC-DC conversion and power management system," *IEEE J. of Solid-State Circuits*, pp.443-451, Mar. 2004.

[4] S. S. Kudva, et al., "Fully integrated on-chip DC-DC converter with a 450x output range," *IEEE J. of Solid-State Circuits*, pp.1940-1951, Aug. 2011.

[5] T. G. Wilson, Jr., "Life after the schematic: the impact of circuit operation on the physical realization of electronic power supplies," *Proc. of the IEEE*, pp.325-334, Apr.1988.

[6] D. Maksimovic, et al., "Modeling and simulation of power electronic converters," *Proc. of the IEEE*, pp.898-912, Jun. 2001.

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

[8] G.W. Wester, et al., "Low-frequency characterization of switched dc-dc converters," *IEEE Trans. on Aerospace and Electronic Systems*, pp.376-385, Mar. 1973.

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

[10] 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.

[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.

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

[13] B. Singh, et al., "A review of single-phase improved power quality AC-DC converters," *IEEE Trans. on Industrial Electronics*, pp.962-981, May 2003.

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

[15] M.-J. Park, et al., "Fast and accurate event-driven simulation of mixed-signal systems with data supplementation," *IEEE Custom Integrated Circuits Conf. (CICC)*, pp.1-4, Sep. 2011.

SUPPLEMENTAL MATERIAL

This supplemental section provides additional experimental results to further demonstrate the efficiency and accuracy of the proposed method.

S.1. DISCONTINUOUS CURRENT MODE (DCM) BUCK CONVERTER EXAMPLE

The discontinuous current mode (DCM) buck converter illustrated in Fig. 13 is chosen as an additional power converter example. Under low load conditions, the power converter needs to operate in a DCM to achieve high efficiency. In this case, the average current flow may be low compared with the current ripple magnitude, and a reverse current flow may exist during phase 2, as shown in Fig. 14(a). Because this reverse current lowers the converter's power efficiency, a DCM converter disconnects switch 2 ($\Phi2$) to block the reverse current (phase 3 in Fig. 14(b)). However, as it is difficult to determine a precise period in phase 2, any discrepancy from the ideal turn-on period will result in performance degradation. For instance, if the switch is turned off later than the zero-crossing instant of the inductor current, the converter will suffer from reverse current. If the switch is disconnected too early, then a large negative voltage spike will appear at the v_X node and turn on the parasitic diode of the MOS pull-down switch (source/drain-to-body pn-junction), resulting in a loss of power due to the diode voltage drop and conduction loss. Most of the DCM power converters use digital control loops to determine the precise time for turning off this switch, and therefore an accurate co-simulation with digital circuitry is key to simulating its performance.

Fig. 15 demonstrates the aforementioned operating phases of a DCM buck converter. The converter can be modeled as a switched linear system with four phases depending on the configuration of the two switches. During the first phase (Fig. 15(a)), the load is connected to the input and then switches to ground during the second phase (Fig. 15(b)). When the circuit switches into the third phase, the parasitic diode either kicks in or is disconnected, depending on the v_X node voltage. If the v_X node voltage is lower than the ground due to a diode voltage turning on at the switching instant, then the parasitic diode turns on and charges the v_X node (Fig. 15(c)). In this phase, the diode is modeled as an ideal voltage source and a current-limit resistor.
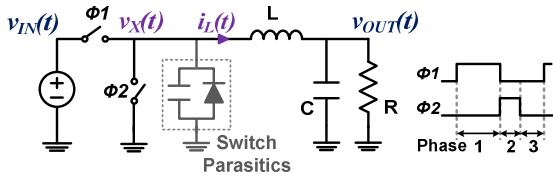

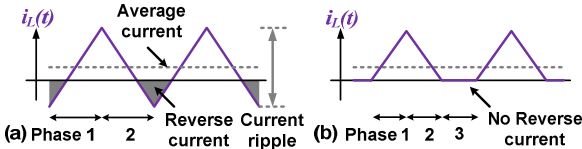Figure 13. A discontinuous current mode buck converter and its switching signals.

Figure 14. (a) A continuous current mode (CCM), and (b) a discontinuous current mode (DCM) for a low-power condition.
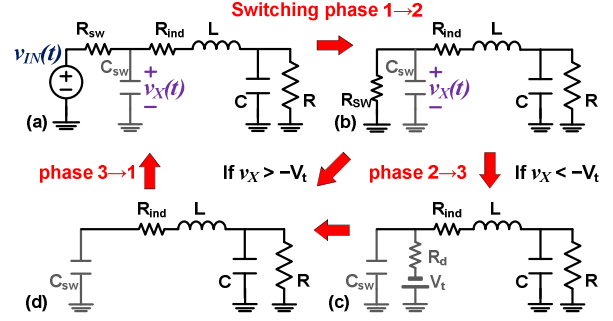

Figure 15. A DCM buck converter: (a) phase 1, (b) phase 2, (c) phase 3 with parasitic diode on, and (d) phase 3 with parasitic diode off.

Once the voltage across the diode becomes smaller than the turn-on voltage, the diode is disconnected, leaving only the parasitic capacitor connected to the inductor (Fig. 15(d)).

Fig. 16 shows the simulated inductor current waveforms for three cases in which switch 2 is disconnected 1) too early, 2) precisely, and 3) too late. The proposed method produces the same waveforms as HSPICE, but again this signal waveform is expressed with far fewer events, which are indicated by arrows in Fig. 16. Based on the inductor current and output voltage, the power efficiency is calculated as shown in Fig. 17. When the period in phase 2 is too short (case 1 in Fig. 16), the power efficiency drops due to the conduction loss of the parasitic diode. With a longer period in phase 2 (case 3 in Fig. 16), the converter efficiency is degraded by the inductor reverse current. The simulation execution time is 82.4 sec for 1-msec simulation time, while SPICE takes 3053.7 sec (~37× slower) under the same conditions.
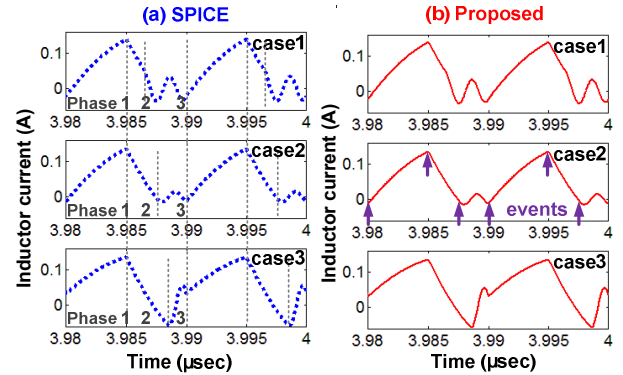

Figure 16. Inductor current waveforms simulated with (a) HSPICE and (b) the proposed method (case1: switch-off too early, case2: switch-off precisely, case3: switch-off too late).
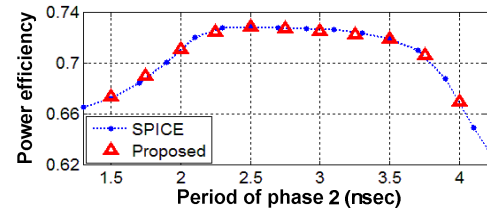

Figure 17. Simulated power efficiency vs. period of phase 2.