Suppression of Chattering in the Real-Time Simulation of the Power Converter

Chen Liu, *Member, IEEE*, Hao Bai, *Member, IEEE*, Rui Ma, *Member, IEEE*, Yaoqiang Wang, *Senior Member, IEEE*, and Fei Gao, *Senior Member, IEEE*

***Abstract*—The achievement of the time-step below 500-ns in the field-programmable-gate-arrays (FPGAs)-based real-time simulation is of importance for the power converters having high switching frequencies. However, most of the existing focus on addressing the power converter modeling under the continuous conduction mode (CCM). Nevertheless, toward practical applications, the modeling of discontinuous conduction mode (DCM) with a light-load is of a greater challenge due to the "chattering" around zero point. In order to solve the chattering problem under the light-load, this article proposes a zero-regulation (ZR) method for FPGA-based real-time simulation. The proposed ZR method can not only represent CCM and DCM with a unified formula but also solve the chattering problem and improve accuracy and stability. In addition, results using different switch models are also given to demonstrate the feasibility and the generality of the proposed method. Finally, a case study of the series load resonant converter is presented. Simulation results are validated against a reference model at a 100-ns time step and a 10-kHz switching frequency.**

***Index Terms*—Chattering, FPGAs, power electronic system modeling, real-time simulation.**

I. Introduction

HARDWARE-IN-THE-LOOP simulation (HiLs) can provide an effective form of testing and validation for the power electronic system. Field-programmable-gate-arrays (FPGAs) in HiLs have permitted the simulation step below 500 ns [1]. Such low time-step is of importance for the power converter targeting switching frequencies in the range of 10–200 kHz range. However, the modeling of power converter on FPGA is a challenging task due to the time-varying topology caused by the ON/OFF-state change of the switching devices [2], [3].

In literature, there are three basic types of switching device model: the R_{on}/R_{off} model [4], [5], the associated discrete circuit (ADC) model [6], [7], and the ideal switch function (ISF) model [8], [9]. Although the switch models are different, the judgment of the switch status is the same. If the switch device is the external signal-controlled type, ON/OFF-status can be computed directly according to gate signals. However, updating a diode's status in the natural commutation process has to depend on the sign of the current or voltage at the switch terminals. Under the DCM, this will cause problems when switches pass from ON state to OFF state and the current is close to zero. Chattering (high-frequency oscillation around the zero-crossing point) is prone to happen. To illustrate this phenomenon, the four-quarter converter (4-QC) with a light-load is shown in Fig. 1 as an illustrative example.

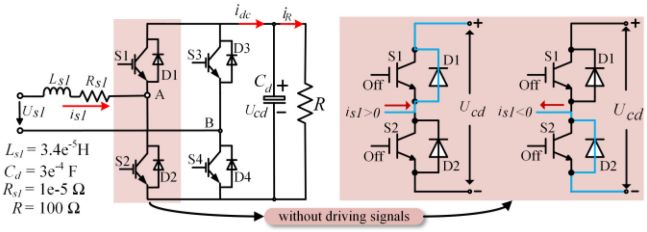

Fig. 1. Illustrative example of 4-QC with a light-load.

A. Illustration of the Chattering

In the example of 4-QC, no control signals are given to the four switching devices, and 4QC operates as the uncontrolled rectifier, in which the antiparallel freewheeling diode always operate. If the value of i_{s1} is positive, the current flows through D_1. Otherwise, D_2 turns ON. After implementing the model on the FPGA board (National Instruments PXIe 7975R) utilizing Forward Euler (FE) Method with a time-step of 100 ns, the results of i_{s1} are shown in Fig. 2, where Fig. 2(b) and (c) are two zoom-in parts of Fig. 2(a).

In Fig. 2, the current i_{s1} in either ISF model or R_{on}/R_{off} model has high-frequency oscillation when i_{s1} is around zero. During the chattering, the numerical results of i_{s1} are always near 0 but not "0." The reason causing chattering can be explained by the

Manuscript received December 2, 2020; revised February 21, 2021; accepted March 18, 2021. Date of publication March 26, 2021; date of current version June 30, 2021. This work was supported by the European Commission H2020 Grant PANDA (H2020-LC-GV-2018), EU Grant Agreement No. 824256. Recommended for publication by Associate Editor T. Suntio. *(Corresponding author: Chen Liu.)*

Chen Liu is with the School of Electrical Engineering, Zhengzhou University, Zhengzhou 450001, China (e-mail: chen.liu@zzu.edu.cn).

Hao Bai and Rui Ma are with the School of Automation, Northwestern Polytechnical University, Xi'an 710072, China (e-mail: hao.bai@nwpu.edu.cn; rui.ma@nwpu.edu.cn).

Yaoqiang Wang is with the School of Electrical Engineering, Zhengzhou University, Zhengzhou 450001, China, and also with the Henan Engineering Research Center of Power Electronics and Energy Systems, Zhengzhou 450001, China (e-mail: wangyqee@163.com).

Fei Gao is with the FEMTO-ST (UMR CNRS 6174), Department of Energy, University of Bourgogne Franche-Comte, Université de Technologie de Belfort Montbéliard, 90010 Belfort, France, and Fuel Cell Lab (FR CNRS 3539), 90010 Belfort, France (e-mail: fei.gao@utbm.fr).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TPEL.2021.3069099.

Digital Object Identifier 10.1109/TPEL.2021.3069099

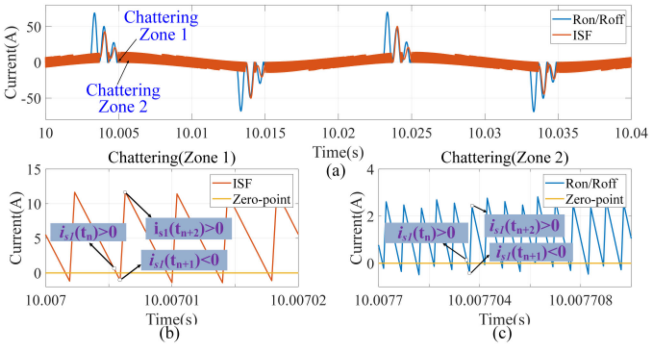

Fig. 2. Results of current i_{s1} in real-time simulation. (a) Simulation results of current i_{s1}. (b) Zoom-in part of Chattering Zone 1 (ISF model). (c) Zoom-in part of Chattering Zone 2 (R_{on}/R_{off} model).

calculation of i_{s1} using

$$L_{s1}\frac{di_{s1}}{dt} \approx \begin{cases} f_1\left(t_{n+1}, i_{s1}\right) \; if \; i_{s1}\left(t_n\right) > 0 \\ f_2\left(t_{n+1}, i_{s1}\right) \; if \; i_{s1}\left(t_n\right) \; = \; 0 \\ f_3\left(t_{n+1}, i_{s1}\right) \; if \; i_{s1}\left(t_n\right) < 0 \end{cases} \tag{1}$$

where $f_1\left(t_n, i_{s1}\right) = \; u_{s1} - R_{s1} \cdot i_{s1} - U_{cd}$, $f_2\left(t_n, i_{s1}\right) = \; 0$, and $f_3\left(t_n, i_{s1}\right) = \; u_{s1} - R_{s1} \cdot i_{s1} + U_{cd}$.

Due to the change of the topology, the calculation of i_{s1} is made up by three modes: two ON-modes ($i_{s1} > 0$ or $i_{s1} < 0$) and one blocked-mode ($i_{s1} = \; 0$). At time step t_{n+1} , if no iteration process is used, the mode selection in (1) is decided by the value of $i_{s1}(t_n)$ from last time-step t_n.

Assume that the 4-QC tries to pass from the ON mode ($i_{s1}(t_{n-1}) > 0$, $f_1(t_{n-1}, i_{s1}) < 0$) to the blocked-mode at the time step t_{n+1}, the value of $i_{s1}(t_{n+1})$ calculated from the ON mode $f_1(t_n, i_{s1})$ with $i_{s1}(t_n)$ is expected to be "0." However, under a fixed-step solver, it is impossible to get the exact accurate result of "'0." As a result, $i_{s1}(t_{n+1})$ keeps decreasing ($f_1(t_{n+1}, i_{s1}) < 0$) and changes directly to a negative value rather than 0. In the next step t_{n+2}, the negative value of $i_{s1}(t_{n+1})$ will cause the mode entering ON mode ($f_3(t_{n+1}, i_{s1})$), which will further bring about the sudden rise of $i_{s1}(t_{n+2})$. This is the beginning of each chattering shown in Fig. 2 (b) and (c).

The major factor causing the chattering is the fixed-step solver used in real-time simulation. In the offline simulation, the variable-step solver can reduce the chattering by reducing the time-step until the state variable lies within the band defined by the signal threshold. But the real-time simulation works in the fixed-step, which does not vary the size of the step.

To sum, the chattering is an undesirable phenomenon when switch passes from ON state to OFF state under the DCM. It is informally characterized by an infinite number of zero-crossing events occurring in a finite time interval. Since the chattering relates to the change of the circuit status in the blocked mode, the chattering occurs no matter the numerical solver is explicit or implicit.

B. Related Research

For the FPGA-based real-time simulation of power electronic system, most of the literature focuses on the continuous conduction mode (CCM), in which the effect of the chattering around zero is too small to notice. Few researchers have discussed the chattering problem under the light-load condition.

In the literature, the iteration process is a possible way to reduce the error during the "chattering." In this method, the iteration continues until the sign of the current and the voltage at the switch terminals are the same [10]. However, the sign of the current and the voltage is not always the same within one time-step. As a result, the maximum iteration numbers must be defined to avoid the overrun in the real-time simulations. The iteration processes, on the other hand, limit the calculation speed.

The other method is the iteration-free process, which is more suitable for real-time simulation. The method introduces the high-impedance status in the switch bridge [11]. The method forces the submodule (SM) of the modular multilevel converter (MMC) to HZ for a single time point of the simulation if the HZ state has not been properly captured at the zero-crossing of the arm current during blocked-mode [11]. But the condition of the blocked mode is actually a defined current bound, the value of which is deduced from the specific topology of the SM in MMC. Since the value is not small enough, this method is not general enough for different topologies. And the implementation of the high-impedance status needs additional equation, which is not suitable for a unified formula simulating the whole system.

To illustrate and solve the chattering problem, this article proposes a zero-regulation (ZR) method for the FPGA-based real-time simulation of the power converter. The remainder of this article is organized as follows. Section II presents the zero-crossing method, as well as the improved results of the 4-QC model in Section I. Section III discusses the effect of the chattering by using the series load resonant (SLR) converter as a case study. Finally, Section IV gives the conclusion.

II. Zero-Regulation Method

To solve the chattering problem, this section first analyze the pattern of the chattering. Based on this pattern, the zero-regulation (ZR) method is then presented.

A. First-Point of the Chattering

Consider the finite-dimensional time invariant system given by the ordinary differential equation (ODE) of the form

$$\frac{dx_n}{dt} = \; f\left(t_n, x\right) \tag{2}$$

where x is the state vector and $f(t, x)$ is the derivative function.

To describe different modes of one bridge in Fig. 1 , (2) can be rewritten by defining different switch status in the form

$$\frac{dx_n}{dt} = \begin{cases} f_1\left(t_n, x\right) \text{ if } g\left(t_n, x\right) > 0 \\ f_2\left(t_n, x\right) \text{ if } g\left(t_n, x\right) = 0 \\ f_3\left(t_n, x\right) \text{ if } g\left(t_n, x\right) < -0 \end{cases} \tag{3}$$

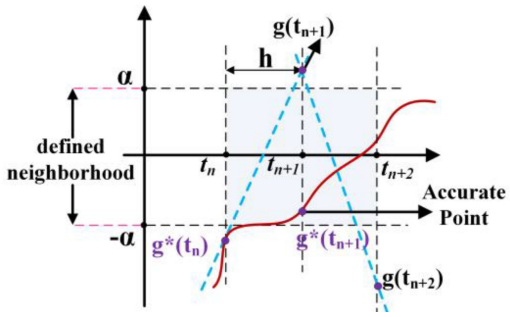

Fig. 3. Chattering point location.

where f_1 and f_3 are the function when the S1 or S3 turns ON (CCM mode), f_2 is the function under the blocked mode, and $g(t,x)$ is the guard function.

The event of $g(t_n,x)$ changing its sign is referred to as a zero-crossing. In practice, $g\ (t_n,x)=\ 0$ can be approximated with a defined neighborhood $|g(t_n,x)|\le\alpha$

$$\frac{dx_n}{dt}=\begin{cases} f_1\ (t_n,x)\ \text{if}\ g\ (t_n,x)>\alpha \\ f_2\ (t_n,x)\ \text{if}\ \ |g\ (t_n,x)|\le\alpha \\ f_3\ (t_n,x)\ \text{if}\ g\ (t_n,x)<-\alpha. \end{cases} \tag{4}$$

Suppose t_{n+1} and t_n are two points embracing the zero point in Fig. 3. In the real-time simulation, if results are not obtained from an iterative solution, at time-step t_{n+1} , the previous time-point of the simulation x_n instead of x_{n+1} is used to identify the boundary at which the change takes place. Assume $g(t_{n+1},x)$ at time point t_{n+1} has the linear form

$$g\ (t_{n+1},x)=\ Cx_n+D. \tag{5}$$

Whereas the accurate value $g^*(t_{n+1},\ x)$ at the time-point t_{n+1} has the form of (6) with x_{n+1}, if t_{n+1} is a point of blocked-mode

$$\begin{aligned} g^*\ (t_{n+1},\ x)=\ Cx_{n+1}+D\ &=\ C\left(\int_{t_n}^{t_{n+1}} f_2\ (t,x)\,dt+x_n\right) \\ &+D\ =\ C\int_{t_n}^{t_{n+1}} f_2\ (t,x)\,dt+g\ (t_{n+1},x)\,. \end{aligned} \tag{6}$$

The difference between (5) and (6) are errors between $g(t_{n+1},x)$ and $g^*(t_{n+1},\ x)$ due to the step latency existing in the calculation. If the condition $|g(t_{n+1},x)|\le\alpha$ can be met, the system will enter the blocked mode. Otherwise, the chattering is likely to happen. For each chattering begins, the start-point in Fig. 2 has one of the following two patterns:

1) $g(t_{n+1},x)>\alpha$ && $g(t_{n+2},x)<-\alpha$;
2) $g(t_{n+1},x)<-\alpha$ && $g(t_{n+2},x)>\alpha$.

To approximate the value of $g(t_{n+2},x)$ at the current step t_n, the predictor–corrector (PC) method [12] is used

$$\begin{aligned} x_{n+1}^p&=x_{n-1}\ +2h\cdot f\ (t_{n-1},x_{n-1}) \\ x_n&=x_{n-1}\ +h\cdot f\ (t_n,x_n^p)\,. \end{aligned} \tag{7}$$

TABLE I
ZERO-REGULATION RULES AT TIME-STEP t_n

during the chattering			
1-th point t_n	Condition	***if*** Condition I	***if*** Condition II
	Revised Parameter	$sgn(t_n)=1$ $x_n=\beta$ $x_{n+1}^p=-\beta$	$sgn(t_n)=1$ $x_n=-\beta$ $x_{n+1}^p=\beta$
2-th point t_{n+1}	Condition	***if*** $sgn(t_n)==1$ && $flag(t_{n+1})==0$	
	Revised Parameter	$sgn(t_{n+1})=2$ $x_{n+1}=-x_n$ $x_{n+2}^p=-x_{n+1}^p$	
out-of-chattering			
3-th point t_{n+2}	Condition	***if*** $\begin{cases} sgn(t_{n+1})==2 \\ g^p(t_{n+4},x)>0 \\ g(t_{n+3},x)>0 \end{cases}$	***if*** $\begin{cases} sgn(t_{n+1})==2 \\ g^p(t_{n+4},x)<0 \\ g(t_{n+3},x)<0 \end{cases}$
	Revised parameter	$sgn(t_n)=3$ $x_{n+2}=\alpha$ $x_{n+3}^p=\alpha$	$sgn(t_n)=3$ $x_{n+2}=-\alpha$ $x_{n+3}^p=-\alpha$

In (7), the PC method calculates the predictor value x_{n+1}^p and the corrector x_n within the same time-step x_n [12]. Meanwhile, $x_{n+1}\approx x_{n+1}^p$. Thus, at time-step t_n, $g^p(t_{n+1},x)$ can be obtained using (8) to replace $g(t_{n+2},x)$ in Fig. 3

$$g\ (t_{n+2},x)=\ Cx_{n+1}+D\approx Cx_{n+1}^p+D\ =g^p\ (t_{n+1},x)\,. \tag{8}$$

Based on (8), the value of $g(t_n,x)$ and $g^p(t_{n+1},x)$ can be obtained at the time-step t_n, the start-point of the chattering can be found. Accordingly, the conditions of the start-point can be defined as:

Definition 2.1: We say that t_n is the start-point of chattering, *if*

Condition I: $g^p(t_{n+1},x)>\alpha$ && $g(t_{n+1},x)<-\alpha$;
Condition II: $g^p(t_{n+1},x)<-\alpha$ && $g(t_{n+1},x)>\alpha$.

B. Zero-Regulation (ZR) Method

Using (7), the solving of the system has the form

$$x_n=\begin{cases} x_{n-1}+h\cdot f_1\ (t_n,x_n^p) & \text{if}\ g^p\ (t_n,x)\ge 0 \\ x_{n-1}+h\cdot f_3\ (t_n,x_n^p) & \text{if}\ g^p\ (t_n,x)<0 \end{cases} \tag{9}$$

$$x_{n+1}^p=\begin{cases} x_{n-1}+2h\cdot f_1\ (t_{n-1},x_{n-1})\ \text{if}\ g\ (t_n,x)\ge 0 \\ x_{n-1}+2h\cdot f_3\ (t_{n-1},x_{n-1})\ \text{if}\ g\ (t_n,x)<0. \end{cases} \tag{10}$$

With x_{n+1}^p and x_n, boundary conditions $g^p(t_{n+1},x)$ and $g(t_{n+1},x)$ can be obtained. Then, based on Definition 2.1, the function $flag(t_n)$ is introduced to check whether t_n is the first-point of chattering

$$flag\ (t_n)=\begin{cases} 0, & \text{else} \\ 1, & \text{if Condition I or II.} \end{cases} \tag{11}$$

Based on the results of (9)–(11), the next step is to regulate the values during the chattering. Once chattering happens at the time-point t_n, zero-regulation rules in Table I are used to revise the result of x_{n+1}^p and x_n calculated by (9) and (10). In

Table I, the function $\mathrm{sgn}(t_n)$ is used to monitor the points during the chattering.

At time step t_n, if flag $(t_n) = 1$ (Condition I or II is met), x_n will be recognized as the start-point of the chattering. $\mathrm{sgn}(t_n)$ is defined to 1, and x_n and x^p_{n+1} are assigned to β or -β. So, at the next step t_{n+1}, due to the opposite sign of x^p_{n+1} and x_n, (9) and (10) calculates x^p_{n+2} and x_{n+1} with different derivative function. If flag $(t_{n+1}) = 1$ and t_{n+1} is detected again as the chattering point, the value of x_{n+1} and x^p_{n+3} will be again assigned to -β or β again, and $\mathrm{sgn}(t_{n+1})$ is 1.

However, there are two possible assignment for [x_n, x^p_{n+1}], namely [β, -β], and [-β, β]. It is impossible to know which combination causes the next chattering. As a result, even if flag $(t_{n+1}) = 0$, x_{n+1} is still taken as the chattering point as long as sgn $(t_n) = 1$. In this situation, the assignment of x_{n+1} and x^p_{n+2} is opposite with x_n and x^p_{n+1}, but $\mathrm{sgn}(t_{n+1})$ is changed to 2.

Thus, at the time-step t_{n+2}, since the sign of x^p_{n+2} and x_{n+1} are still different, the (9) and (10) still calculate x^p_{n+3} and x_{n+2} with different derivative function. If x_{n+2} is detected again as the start-point of the chattering and flag $(t_{n+2}) = 1$, the value of x_{n+2} and x^p_{n+3} will be again assigned to -β or β.

On the other hand, if flag $(t_{n+2}) = 0$ and the guard function has the same sign, the out-of-chattering state will be checked. Since no blocked-mode function $f_2(t_n, x_n)$ is involved, errors are prone to be introduced if $g^p(t_{n+2}, x)$ and $g(t_{n+2}, x)$ are within the interval $[-\alpha, \alpha]$ and have the same sign. To avoid this error, the ZR method forces x_{n+2} and x^p_{n+3} to α or -α if the out-of-chattering state has been captured. Unless the following point $(\boldsymbol{t_{n+3}})$ do not meet the chattering condition $(f\ (\boldsymbol{t_{n+3}}) = 0)$, the chattering loop breaks.

In this method, although the chattering keeps existing, the value inside the chattering zone is regulated to -β or β. Besides, compared with the traditional method using blocked-mode function $f_2(t_n, x_n)$ to simulate the high-impedance status, the proposed method only utilizes the derivative function $f_1(t_n, x_n)$ and $f_3(t_n, x_n)$ in the CCM mode and no blocked-mode function $f_2(t_n, x_n)$ is used. It means that the proposed method is capable of simulating the blocked-mode with the same function in the CCM mode. This feature permits a unified formula to simulate the whole power electronic system, which could largely simplify the modeling process.

C. Parallel Implementation Structure on FPGAs

To sum up, the proposed ZR method consists of three parts: the prediction solving process (9), the correction solving process (10), and the zero-crossing unit (11) and Table I. Once chattering happens, the values in the prediction process x_n and the correction x^p_{n+1} are regulated by the zero-crossing unit to the value of β, which is negligible enough to be ignored. Fig. 5 is the corresponding FPGAs structure.

In Fig. 4 , the implementation of (9) and (10) is following the general structure summarized by [13]. Due to each variable in (9) and (10) is independent, the (9) and (10) can be calculated at the same time. Besides, since no math operation is involved, the zero-crossing unit would not ruffle the overall calculation

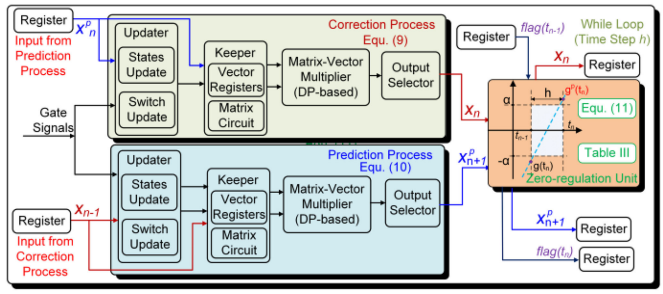

Fig. 4. Implementation of ZR method.

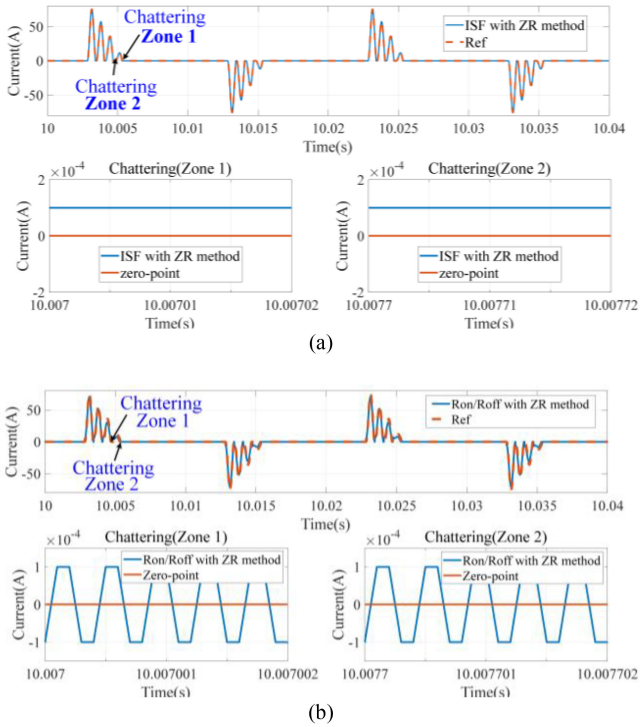

Fig. 5. Results of i_{s1} on FPGAs with the proposed ZR method (a) ISF Model. (2) $\mathrm{R_{on}}$/$\mathrm{R_{off}}$ Model.

speed. Benefiting the parallel structure and the simple structure in the zero-crossing unit, the proposed method has almost the same speed as the Forward Euler (FE) method.

D. Improved Results of the 4-QC Under ZR Method

After adding the proposed zero-crossing method to the 4-QC study in Section II, the results of current i_{s1} in the ISF model and the $\mathrm{R_{on}}$/$\mathrm{R_{off}}$ model are shown in Fig. 5 .

In the 4-QC modeling with the proposed ZR method, β in (12) and (13) is set to $1e^{-4}$ (A) and the α in (11) is $1e^{-3}$ (A). Compared with Fig. 2 , the values of chattering zones in Fig. 5 only have two values: $1e^{-4}$ (A) or $-1e^{-4}$ (A). Despite the switch still oscillates between ON and OFF, the effect of the chattering is limited to a value that can be truly "neglected."

Fig. 6 is the absolute error of U_{cd} with/without the proposed ZR method with the MATLAB/Simulink as the reference. In

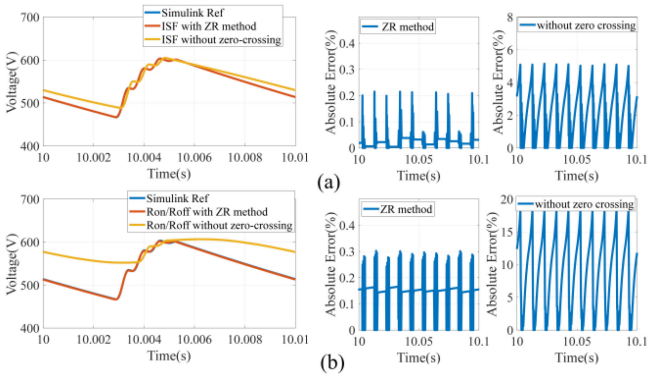

Fig. 6. Absolute Error of U_{cd} with/without the proposed ZR method (a) ISF Model. (b) R_{on}/R_{off} Model.

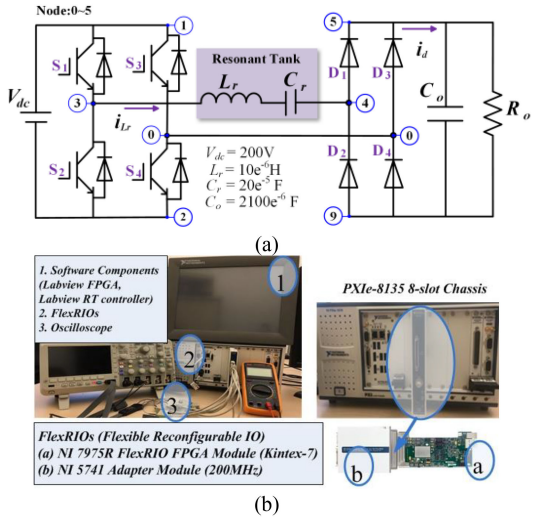

Fig. 7. Topology and the real-time test bench. (a) Topology of SLR converter. (b) NI PXIe Platform.

the ISF model, when the diodes operate in the blocked-mode, the error in the model without the zero-crossing method can reach 5% while the error in the model with the zero-crossing is less than 0.2%. And in the R_{on}/R_{off} model, with the proposed zero-crossing method, the error is reduced from 20% to only 0.3%. From the comparison, it can be seen that the accuracy of the model with the zero-crossing unit is greatly improved.

III. Case Study

The other case study is the series load resonant (SLR) converter shown in Fig. 7(a) with a resonant tank (L_r and C_r) [14]. An open-loop control frequency f_s of 10 kHz and dead-time of 500 ns is used to trigger the switch S1, S2, S3, and S4. The open-loop control method ensures that the error comes from the modeling part.

Fig. 7 (b) is the NI PXIe Platform we used for the implementation of the proposed ZR method and the case study. It has FPGA module (NI 7975R) and an adapter module (NI 5741) to output the analog signals and the digital signals. The used FPGA is the Kintex-7 XC7K410T, which contains 508400 Slice Registers, 63550 Total Slices, 254200 Slice LUTs, 795 Block RAMs, and 1540 DSP48s [15].

A. Real-Time Simulation With ISF Model

In the ISF model, the switch's status is defined by the switch function S_i $(i = 12, 34)$ and D_i. During the natural commutation process, as shown in Table I, the value of the switch function at the time step t_n is decided by the predictor value $i^p_{Lr(n)}$ and the corrector value $i_{Lr(n-1)}$ calculated from last step t_{n-1}. As a result, the guard function is $g\ (t_{n+1}, i_{Lr}) = i_{Lr(n)}$.

According these switch functions, the nodal voltage U_{30}, U_{40} and the branch current I_d can be obtained by

$$\begin{aligned} U^p_{30(n)} &= u_{dc}\ (S^p_1 - S^p_3) \\ U^p_{40(n)} &= u^p_{Co(n)}\ (D^p_1 - D^p_3) \\ i^p_{d(n)} &= i^p_{d1(n)} + i^p_{d2(n)} = i^p_{Lr(n)}\ D^p_1 - i^p_{Lr(n)} D^p_1 \end{aligned} \tag{12}$$

$$\begin{aligned} u_{30(n-1)} &= u_{dc}\ (S_1 - S_3) \\ u_{40(n-1)} &= u_{Co}\ (D_1 - D_3) \\ i_{d(n-1)} &= i_{Lr(n-1)}\ D_1 - i_{Lr(n-1)} D_3. \end{aligned} \tag{13}$$

And the calculatioin of i_{Lr} and u_{Cr} is as follows:

$$\begin{aligned} L_r \frac{di_{Lr}}{dt} &= u_{Lr} = U_{30} - U_{40} - u_{Cr} \\ U_{cr} \frac{du_{Cr}}{dt} &= i_{Cr} = i_{Lr} \\ U_{cr} \frac{du_{Co}}{dt} &= i_{Co} = i_d - \frac{u_{Co}}{R_o}. \end{aligned} \tag{14}$$

The discretization formulation of (14) using (9) and (10) can be derived as follows:

$$\begin{aligned} i^p_{Lr(n+1)} &= i_{Lr(n-1)} + \left(\frac{2h}{L_r}\right) u_{Lr(n-1)} \\ i_{Lr(n)} &= i_{Lr(n-1)} + \left(\frac{h}{L_r}\right) u^p_{Lr(n)} \\ u^p_{Cr(n+1)} &= u_{Cr(n-1)} + \left(\frac{2h}{L_r}\right) i_{Lr(n-1)} \\ u_{Cr(n)} &= u_{Cr(n-1)} + \left(\frac{h}{L_r}\right) i^p_{Lr(n)} \\ u^p_{Co(n+1)} &= u_{Co(n-1)} + \left(\frac{2h}{L_r}\right) i_{Co(n-1)} \\ u_{Co(n)} &= u_{Co(n-1)} + \left(\frac{h}{L_r}\right) i^p_{Co(n)}. \end{aligned} \tag{15}$$

After modeling, the whole model is implemented with a single-cycle time-loop (SCTL) on the National Instrument (NI) shown in Fig 7 (b). The SCTL executes all functions inside within one tick of the FPGA clock [15], [16]. Based on Fig. 4 , the implementation of the above equations is shown in Fig. 8.

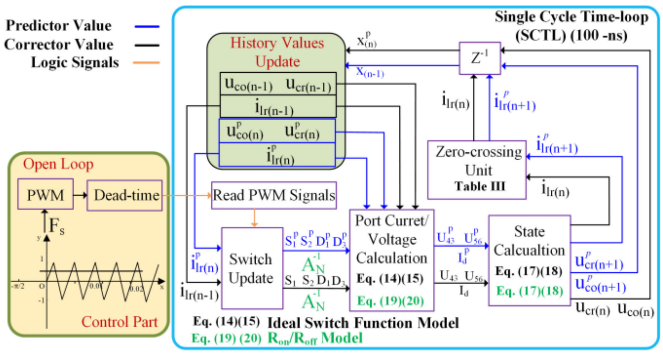

Fig. 8. Implementation on LabVIEW FPGA.

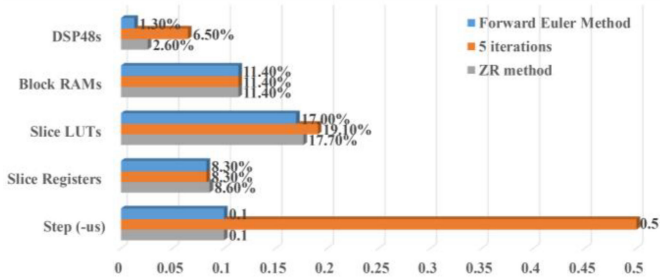

Fig. 9. Resource utilization of different solving methods.

The whole model modeling has three sequential steps. First, the driving signals, the predictor $i^{p}_{\mathrm{Lr(n)}}$, and the corrector $i_{\mathrm{Lr(n-1)}}$ determine the switch status in the correction process and the prediction process. Then, utilizing the corresponding switch function, the nodal voltage and the branch current will be calculated by (12), (13) and (15), (16). At last, the current results $i_{\mathrm{Lr(n)}}$ and $i^{p}_{\mathrm{Lr(n+1)}}$ are checked by the zero-regulation rule summarized in Table I.

To show the feature of the proposed ZR, FE method and Backward Euler (BE) method with five iterations are compared from the aspect of calculation speed, FPGA resource utilization, and the simulation accuracy with the MATLAB/Simulink results as the reference.

In Fig. 9, the time-step in the FE and the proposed ZR method can reach 100 ns, while the BE with five iteration is 500 ns due to the five-iteration procedure. And since the proposed zero-crossing method utilizes the prediction process and the correction process to regulate the chattering situation, the DSP48s is doubled to 2.6% while the FE method is only 1.3%. From the comparison in Fig. 10, the proposed ZR method is able to achieve the same calculation speed as the FE method but requires double DSP48s resources than the FE method.

Fig. 10 presents the results of i_{Lr}. In Fig. 10 (a), when $\mathrm{R}_o = 1\ \Omega$, there is little difference among FE, BE, and the proposed ZR method. When the value of R_o increases to 20 Ω, the current is discontinuous. Oscillation in both FE and the iteration method exists but the range of the oscillation is relatively small, and sometimes can be neglected. However, in Fig. 10 (c), when $\mathrm{R}_o = 80\ \Omega$ and the current i_{Lr} is around zero, the oscillation in the FE and the iteration method is too large to neglect. Fig. 10 (d) is the

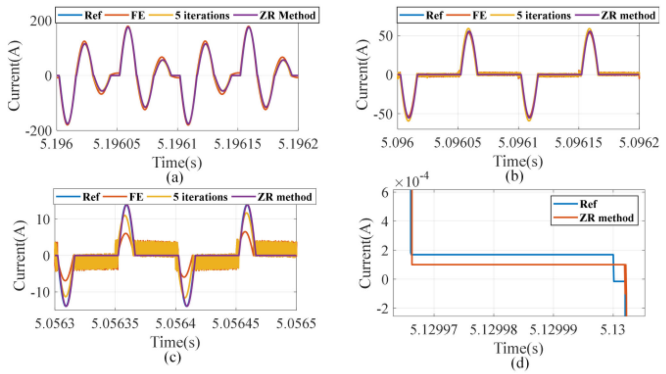

Fig. 10. Simulation results of i_{Lr} on FPGAs. (a) $\mathrm{R}_o = 1\ \Omega$; (b) $\mathrm{R}_o = 20\ \Omega$; (c) $\mathrm{R}_o = 80\ \Omega$; (d) zoom-in part of i_{Lr} when i_{Lr} is around zero ($\mathrm{R}_o = 80\ \Omega$).

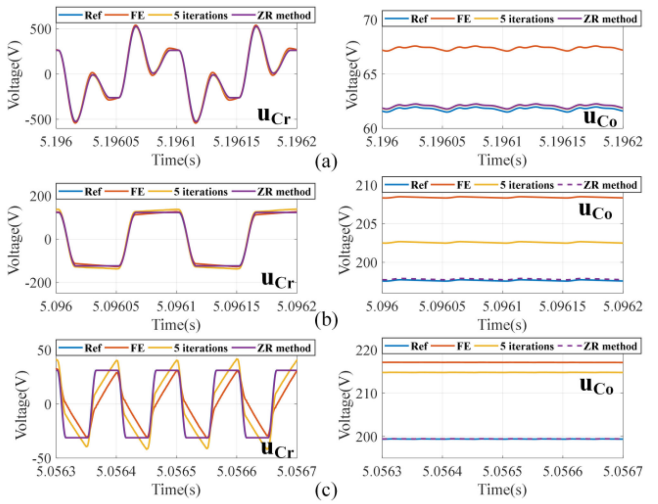

Fig. 11. Simulation results of u_{Cr} and u_{Co} on FPGAs. (a) $\mathrm{R}_o = 1\ \Omega$; (b) $\mathrm{R}_o = 20\ \Omega$; (c) $\mathrm{R}_o = 80\ \Omega$.

result of the proposed ZR method and the reference when i_{Lr} is around zero. Compared with the FE and the iteration method, the proposed ZR method regulates the current oscillation to -1e^{-4} (A), which are small enough to be neglected.

For FE and iteration methods shown in Fig. 10 , high oscillation exists when i_{Lr} is close to zero and the load is light. Although the system can work normally during chattering, great errors are likely to be introduced to the system and cause a potentially unstable problem. To further illustrate this side effect, simulation results of u_{Cr} and u_{Co} on FPGAs are shown in Fig. 11.

In Fig. 11 (a) and fig11(b), it can be noticed that the iteration method is more accurate than the FE method. However, when the load R_o is light and reaches 80 Ω, huge differences among FE, the iteration method, and the reference model can be observed in Fig. 11(c). But by regulating the chattering value when the current is around zero, the proposed ZR method always keeps high consistent with the Simulink model no matter how the load changes.

TABLE II
SWITCH FUNCTION DESCRIPTION

i_{Lr}	S_1	S_1^p	S_3	S_3^p	D_1	D_1^p	D_3	D_3^p
$i_{Lr(n-1)} < 0$	1	-	0	-	0	-	1	-
$i_{Lr(n-1)} \geq 0$	0	-	1	-	1	-	0	-
$i^p_{Lr(n)} < 0$	-	1	-	0	-	0	-	1
$i^p_{Lr(n)} < 0$	-	0	-	1	-	1	-	0

TABLE III
SWITCH FUNCTION DESCRIPTION

I_{Lr}	$sign$	R_{S1}	R^p_{S1}	R_{S2}	R^p_{S2}	R_{S3}	R^p_{S3}	R_{S4}	R^p_{S4}
$I_{Lr(n-1)}$	< 0	R_{off}	-	R_{on}	-	R_{off}	-	R_{on}	-
	≥ 0	R_{on}	-	R_{off}	-	R_{on}	-	R_{off}	-
$I^p_{Lr(n)}$	< 0	-	R_{off}	-	R_{on}	-	R_{off}	-	R_{on}
	≥ 0	-	R_{on}	-	R_{off}	-	R_{on}	-	R_{off}

I_{Lr}	$sign$	R_{D1}	R^p_{D1}	R_{D2}	R^p_{D2}	R_{D3}	R^p_{D3}	R_{D4}	R^p_{D4}
$I_{Lr(n-1)}$	< 0	R_{off}	-	R_{on}	-	R_{off}	-	R_{on}	-
	≥ 0	R_{on}	-	R_{off}	-	R_{on}	-	R_{off}	-
$I^p_{Lr(n)}$	< 0	-	R_{off}	-	R_{on}	-	R_{off}	-	R_{on}
	≥ 0	-	R_{on}	-	R_{off}	-	R_{on}	-	R_{off}

B. Real-Time Simulation With R_{on}/R_{off} Model

In the R_{on}/R_{off} model, the switch's status is judged by Table II. Utilizing the modified nodal analysis, the calculation of the status of the whole system is shown in Table III. The guard function is $g\,(t_{n+1}, I_{Lr}) = I_{Lr(n)}$

$$A_N \begin{bmatrix} V_n \\ I_w \end{bmatrix} = \begin{bmatrix} J_n \\ V_{ws} \end{bmatrix} \tag{16}$$

where $V_n^T = [\,V_1\ V_2\ V_3\ V_4\ V_5\ V_6\,]$, $I_w^T = [\,I_1\ I_d\,]$

$$J_n^T = \begin{bmatrix} 0\ 0\ i_{Lr}\ -i_{Lr}\ 0\ 0 \end{bmatrix},\ V_{ws}^T = \begin{bmatrix} V_{dc}\ u_{co} \end{bmatrix}$$

$$A_N = \left[\begin{array}{cc|c} Y_{n1} & 0 & \multirow{2}{*}{A_w} \\ 0 & Y_{n2} & \\ \hline A_w^T & & 0 \end{array}\right],\ A_w^T = \begin{bmatrix} 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -1 \end{bmatrix}$$

$$Y_{n1} = \begin{bmatrix} \frac{1}{R_{S1}} + \frac{1}{R_{S3}} & 0 & -\frac{1}{R_{S1}} \\ 0 & \frac{1}{R_{S2}} + \frac{1}{R_{S4}} & -\frac{1}{R_{S2}} \\ -\frac{1}{R_{S1}} & -\frac{1}{R_{S2}} & \frac{1}{R_{S1}} + \frac{1}{R_{S2}} \end{bmatrix}$$

$$Y_{n2} = \begin{bmatrix} \frac{1}{R_{D1}} + \frac{1}{R_{D2}} & -\frac{1}{R_{D1}} & -\frac{1}{R_{D2}} \\ -\frac{1}{R_{D1}} & \frac{1}{R_{D1}} + \frac{1}{R_{D3}} & -\frac{1}{R_o} \\ -\frac{1}{R_{D2}} & -\frac{1}{R_o} & \frac{1}{R_{D2}} + \frac{1}{R_{D4}} \end{bmatrix}.$$

In the real-time simulation, the calculation of the inverse matrics is too complex to implement. To achieve an ultrafast calculation speed, all the possible results of the inverse matrices are precomputed and stored in the FPGA. So, after obtaining the results of the inverse matrix A_N^{-1}, the status of the whole system can be calculated as follows:

$$\begin{bmatrix} V_{n(n)} \\ I_{w(n)} \end{bmatrix} = A_N^{-1} \begin{bmatrix} J_{n(n)} \\ V_{ws(n)} \end{bmatrix} \tag{17}$$

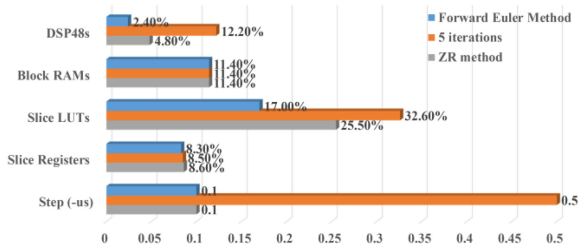

Fig. 12. Fig. 12 Hardware resource utilization.

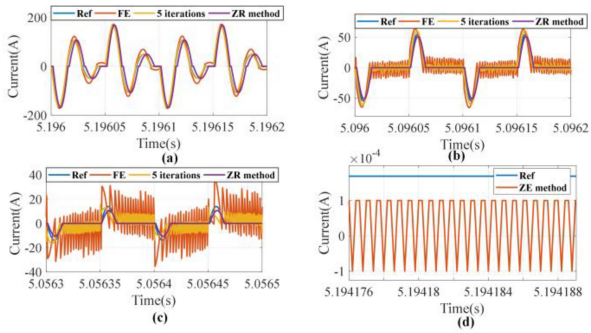

Fig. 13. Simulation results of i_{Lr} on FPGAs. (a) $R_o = 1\ \Omega$; (b) $R_o = 20\ \Omega$; (c) $R_o = 80\ \Omega$; (d) zoom-in part of i_{Lr} when $R_o = 80\ \Omega$ and i_{Lr} is around zero.

$$\begin{bmatrix} V^p_{n(n+1)} \\ I^p_{w(n+1)} \end{bmatrix} = A_N^{-1} \begin{bmatrix} J^p_{n(n)} \\ V^p_{sw(n)} \end{bmatrix}. \tag{18}$$

Using the zero-crossing method proposed, in one bridge, there are only two possible combinations for the switch [ON, OFF] or [OFF, ON] and the high impedance status [OFF, OFF] is eliminated. As a result, in A_N, results of $(\frac{1}{R_{S1}} + \frac{1}{R_{S2}})$, $(\frac{1}{R_{S3}} + \frac{1}{R_{S4}})$, $(\frac{1}{R_{D1}} + \frac{1}{R_{D2}})$ and $(\frac{1}{R_{D3}} + \frac{1}{R_{D4}})$ are always equal to $(\frac{1}{R_{on}} + \frac{1}{R_{off}})$ rather than $(\frac{1}{R_{off}} + \frac{1}{R_{off}})$. This will largely reduced the requirement of the storage room.

After obtaining V_{43}, V_{56}, and I_d, (15) and (16) can be reused to calculate the voltage/current status of the system. The implementation structure on the FPGA board using Labview is also based on Fig. 9, where the results of A_N^{-1} are stored in a look-up table and searched by the sign of current I_{Lr} and driving signals.

Under the SCTL of Labview FPGA, the time clock of 100 ns can also be met. It means that the total calculation time of the SLR converter case study in R_{on}/R_{off} model is less than 100 ns. The FPGA compilation results are shown in Fig. 12. And the simulation results of i_{Lr}, u_{Cr}, and u_{C0} under different loads are shown in Figs. 1 3 and 14.

In Fig. 12, both FE method and ZR method can meet the time-constraint of 100 ns, while the iteration with five times needs 500 ns to finish the calculation. Compared with the FE method, the DSP48s in the proposed ZR method is 4.8%, which is twice as much as the FE method (2.4%).

In Figs. 13(a) and 14(a), when $R_o = 1\ \Omega$, the FPGA's simulation results of FE, iteration methods, and the proposed method using R_{on}/R_{off} model have a high agreement with the reference model. As the load R_o increases to 20 Ω [see Figs. 13(b) and

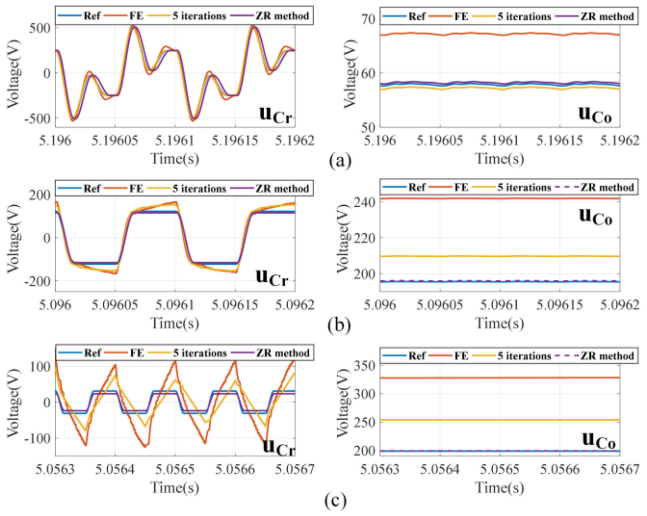

Fig. 14. Simulation results of u_{Cr} and u_{Co} on FPGAs: (a) $R_o = 1\ \Omega$; (b) $R_o = 20\ \Omega$; (c) $R_o = 80\ \Omega$.

14(b)], the relative error of i_{Lr}, u_{Cr}, and u_{C0} in the FE and the iteration methods in augments but still acceptable. However, in Figs. 13(c) and 14(c), when $R_o = 80\ \Omega$ and the current i_{Lr} is around zero, huge errors exist in the FE and the iteration methods while the proposed zero-crossing method can keep high agreement with the Simulink model. In Fig. 13(d), the chattering in the proposed method is regulated to -1e^{-4} (A) and 1e^{-4} (A). Compared with the reference model in the Simulink, the value of oscillation can be truly neglected.

C. Summary and Comparison

Regarding the implementation, at time step t_n, since each variable $[x_{n-1}, x_n^p]$ in (12)–(15), (17), and (18) can be known from last step t_{n-1}, the calculation of x_n in the correction process and x_{n+1}^p in the prediction process can be calculated at the same time. This independent feature can be easily tailored for the parallel calculation on the FPGAs, the implementation of which is referred to Fig. 5 . Besides, (12)–(15), (17), and (18) are equations in the CCM mode without introducing DCM mode equations considering the discontinuous conditions, the proposed ZR method allows the unified formation simulating the whole system.

Regarding the FPGAs resource utilization, due to the different calculation formation in ISF model and R_{on}/R_{off} model, the FPGAs resources required are different. R_{on}/R_{off} model requires more math operations and occupies more FPGAs resources than the ISF model. However, the similarity in Figs. 9 and 12 can be summarized as follows:

1) Compared with the FE method, the proposed method doubles the DSP48s resources.
2) The proposed ZR method could achieve the same calculation speed as the FE method (100 ns).
3) The proposed zero-crossing regulation rule can be used in both ISF model and R_{on}/R_{off} model.

As for the simulation results, in Figs. 5, 10, and 13, the proposed ZR not only provides accurate results in both the high-load condition and the light-load condition. Compared with Figs. 6 , 10(d), and 13(d), it can be found that, in Figs. 6 and 13(d), the regulated value during chattering is the repeating 1e^{-4} (A) and -1e^{-4} (A) while in Fig. 10(d), the regulated value in Fig. 10(d) is $-1e^{-4}$ while Fig. 13(d) is $-1e^{-4}$. It means that for different topologies and switch model, the condition triggering the chattering is possibly different.

IV. Conclusion

In this article, a zero-regulation (ZR) method for FPGA-based real-time simulation is proposed to build the model of the power electronic system and solve the "chattering" problem under the light-load. The proposed method has two independent calculation processes: the prediction process and the correction process. By combing the CCM model function with the zero-regulation method to simulate the blocked-mode, the proposed method allows a unified formula simulating the whole system.

Performance evaluations on different switch models (ISF model and R_{on}/R_{off} model) using SLR converter are carried out. The proposed zero-crossing method prevails the method without the zero-crossing while keeps the same calculation speed as the FE method. Using the proposed method, the total calculation time of the case study is only 100 ns under the SCTL of the LabVIEW FPGA. When the converter operates with a light-load, the proposed ZR method can regulate the current to $-1e^{-4}$ (A) or 1e^{-4} (A). Thus, the oscillation can be truly "negligible." Compared with the FE method or BE method without using zero-crossing, the accuracy of the ZR method is largely improved. The application on the ISF model and R_{on}/R_{off} model also shows the generality of the proposed method.

References

[1] S. Mojlish, N. Erdogan, D. Levine, and A. Davoudi, "Review of hardware platforms for real-time simulation of electric machines," *IEEE Trans. Transp. Electrific.*, vol. 3, no. 1, pp. 130–146, Mar. 2017.
[2] C. Liu, X. Guo, R. Ma, Z. Li, F. Gechter, and F. Gao, "A system-level FPGA-based hardware-in-the-loop test of high-speed train," *IEEE Trans. Transp. Electrific.*, vol. 4, no. 4, pp. 912–921, Dec. 2018.
[3] A. Emadi, "Modeling and analysis of multiconverter DC power electronic systems using the generalized state-space averaging method," *IEEE Trans. Power Electron.*, vol. 51, no. 3, pp. 661–668, Jun. 2004.
[4] D. Majstorovic, I. Celanovic, N. D. Teslic, N. Celanovic, and V. A. Katic, "Ultralow-latency hardware-in-the-loop platform for rapid validation of power electronics designs," in *IEEE Trans. Ind. Electron.*, vol. 58, no. 10, pp. 4708–4716, Oct. 2011.
[5] T. Ould-Bachir, H. F. Blanchette, and K. Al-Haddad, "A network tearing technique for FPGA-based real-time simulation of power converters," *IEEE Trans. Ind. Electron.*, vol. 62, no. 6, pp. 3409–3418, Jun. 2015.
[6] C. Dufour, J. Mahseredjian, and J. Bélanger, "A combined state-space nodal method for the simulation of power system transients," *IEEE Trans. Power Del.*, vol. 26, no. 2, pp. 928–935, Apr. 2011.
[7] P. Pejovic and D. Maksimovic, "A method for fast time-domain simulation of networks with switches," *IEEE Trans. Power Electron.*, vol. 9, no. 4, pp. 449–456, Jul. 1994.
[8] A. Hadizadeh, M. Hashemi, M. Labbaf, and M. Parniani, "A matrix-inversion technique for FPGA-based real-time EMT simulation of power converters," *IEEE Trans. Ind. Electron.*, vol. 66, no. 2, pp. 1224–1234, Feb. 2019.
[9] C. Liu, H. Bai, R. Ma, X. Zhang, F. Gechter, and F. Gao, "A network analysis modeling method of the power electronic converter for hardware-in-the-loop application," *IEEE Trans. Transp. Electrific.*, vol. 5, no. 3, pp. 650–658, Sep. 2019.

[10] H. Saad *et al.*, "Modular multilevel converter models for electromagnetic transients," *IEEE Trans. Power Del.*, vol. 29, no. 3, pp. 1481–1489, Jun. 2014.
[11] T. Ould-Bachir, H. Saad, S. Dennetière, and J. Mahseredjian, "CPU/FPGA-based real-time simulation of a two-terminal MMC-HVDC system," *IEEE Trans. Power Del.*, vol. 32, no. 2, pp. 647–655, Apr. 2017.
[12] C. Liu, H. Bai, S. Zhuo, X. Zhang, R. Ma, and F. Gao, "Real-time simulation of power electronic systems based on predictive behavior," *IEEE Trans. Ind. Electron*, vol. 67, no. 9, pp. 8044–8053, Sep. 2020.
[13] F. Montano, T. Ould-Bachir, and J. P. David, "An evaluation of a high-level synthesis approach to the FPGA-based submicrosecond real-time simulation of power converters," *IEEE Trans. Ind. Electron*, vol. 65, no. 1, pp. 636–644, Jan. 2018.
[14] T. Taufik, M. McCarthy, S. Watkins, and M. Anwari, "Performance study of series loaded resonant converter using super barrier rectifiers," in *Proc. IEEE Region 10 Conf.*, 2009, pp. 1–5.
[15] NI FPGA Module for PXI Express. Jun. 2019. [Online]. Available: http://www.ni.com/pdf/product-flyers/flexrio-custom-instrumentation.pdf
[16] NI PXIe-7975R, NI FlexRIO^{TM} FPGA Module for PXI Express, Revision No. 1.1, Feb. 2014. [Online]. Available: http://download.ni.com/ pub/gdc/ tut/labview_high-perf_fpga_v1.1.pdf

Chen Liu (Member, IEEE) received the M.S. degrees from Beijing Jiaotong University, Beijing, China, in 2014, and the Ph.D. degree from the University of Technology of Belfort-Montbéliard, Belfort, France, in 2018, all in electrical engineering.

From 2018 to 2020, he worked as a Postdoctoral Researcher with the School of Energy and Computer Science, UTBM, Belfort, France. He is currently an Associate Professor of Zhengzhou University, Zhengzhou, China. His main research interests include real-time simulation of power electronic system and hardware-in-the-loop test.

Hao Bai (Member, IEEE) received the Ph.D. degree in electrical engineering from the University of Technology of Belfort-Montbeliard, Belfort, France, in 2019.

He is currently an Associate Professor with Northwestern Polytechnical University, Xi'an, China. His main research interests include real-time simulation of power electronics systems, digital twin-based health monitoring, and prognosis of fuel cell hybrid propulsion systems.

Rui Ma (Member, IEEE) received the M.S. degree in electrical engineering from Northwestern Polytechnical University (NPU), Xi'an, China, in 2016. He received the dual Ph.D. degree in electrical engineering from both the University of Technology of Belfort-Montbéliard (UTBM), Belfort, France, and NPU, Xi'an, China, in 2018 and 2019, respectively.

He is currently an Associate Professor with the School of Automation at NPU. His main research interests include fuel cell modeling, fuel cell applications, real-time simulation systems, and machine learning.

Dr. Ma is an Associate Editor for IEEE IES Industrial Electronics Technology News, and a Distinguished Reviewer for IEEE TRANSACTIONS ON INDUSTRIAL ELECTRONICS. He also served as a member of the Technical Committee on Transportation Electrification of the IEEE IES, and Special Session Chairs of 45th and the 46th IEEE IECON.

Yaoqiang Wang (Senior Member, IEEE) received the B.S. degree in measurement and control technology and instruments from Hangzhou Dianzi University, Hangzhou, China, in 2006, and the M.S. and Ph.D. degrees in electrical engineering from the Harbin Institute of Technology, Harbin, China, in 2008 and 2013, respectively.

He is currently working with the School of Electrical Engineering, Zhengzhou University, Zhengzhou, China. He is also serving as the Director of the Institute of Power Electronics and Energy Systems of Zhengzhou University, the Zhengzhou Engineering Research Center of Power Control and Systems, and the Henan Engineering Research Center of Power Electronics and Energy Systems. He has published more than 50 peer-reviewed papers including over 40 journal papers, and is the holder of more than ten patents. His current research interests include power electronics, renewable energy generation, flexible power distribution, MVdc, electric motor drives, and electrified transport.

Fei Gao (Senior Member, IEEE) received the Ph.D. degree in renewable energy from University of Technology of Belfort-Montbeliard, Belfort, France, in 2010.

He is currently the Deputy Director of the French National CNRS Research Institute FEMTO-ST and a Full Professor with the School of Energy and Computer Science, University of Technology of Belfort-Montbeliard (UTBM). His main research fields include fuel cells and their applications in transportation, multiphysical modeling and real time simulation systems.

Prof. Gao is the recipient of 2020 IEEE J. David Irwin Early Career Award from IEEE Industrial Electronics Society, and also the Distinguished Youth Doctor Award in 2010. He is fellow of IET and holder of the French Research Expertise Bonus (PEDR) from the French Ministry of Higher Education and Research. He is also the Editor-In-Chief of IEEE INDUSTRIAL ELECTRONICS TECHNOLOGY NEWS, the Assistant Deputy Editor-In-Chief of IEEE TRANSACTIONS ON TRANSPORTATION ELECTRIFICATION, and an Associate Editor of IEEE TRANSACTIONS ON INDUSTRIAL ELECTRONICS, IEEE TRANSACTIONS ON INDUSTRY APPLICATIONS, IEEE TRANSACTIONS ON ENERGY CONVERSION, and IEEE Open Journal of Industrial Electronics Society. He is nominated in 2017 as Conferences Committee Chair of IEEE Transportation Electrification Community. He serves, since 2019, as Secretary of the Technical Committee on Vehicle and Transportation Systems of IEEE Power Electronics Society and was the Chair of the Technical Committee on Transportation Electrification of the IEEE Industry Electronic Society from 2018 to 2019.