Validation Method for Hardware-in-the-Loop Simulation Models

Tamás Kökényesi István Varjasi

Department of Automation and Applied Informatics, Budapest University of Technology and Economics, Hungary,
{kokenyesi.tamas, varjasi.istvan}@aut.bme.hu

Abstract

The advances in FPGA technology have enabled fast real-time simulation of power converters, filters and loads. HIL (Hardware-in-the-Loop) simulators taking advantage of this technology have revolutionized control hardware and software development for power electronics. Switching frequencies in today's power converters are getting higher and higher, so reducing calculation time steps in HIL simulators is critical, especially if simulating lower power circuits. Faster calculation can be achieved with simpler models or lower resolution. Both possibilities require the validation of the FPGA-synthesizable simulation models to check whether they are correct representations of the simulated main circuit or not. The subject of this paper is a validation method, which treats the simulation error similar as production variance, which can be measured between different instances of the original main circuit.

Keywords: circuit simulation, power circuit modeling, signal resolution, discrete-time systems

1 Introduction

General power converters consist of two main parts: a power stage (main circuit) and a digital controller unit, which is usually realized using a DSP or FPGA. Testing such a controller unit on its original main circuit is expensive and dangerous. That's why offline computer simulation is often used for testing such converters (Rajapakse, 2005; Sybille, 2007). There are very precise models for offline simulation (e.g. PSPICE based simulators), but they can be only used for initial testing of the control algorithms, not the implementation. A low-power model of the main circuit can be built under laboratory conditions, but it will have parameters differing from the ones of the original system.

A very effective way to test controller units' both hardware and software is HIL (Hardware-In-the-Loop) simulation (Kokenyesi, 2013 JEPE). It combines the advantages of other testing methods: low cost like offline computer simulation, complex tests like laboratory testing and realistic conditions like testing on the field. HIL technology also allows the simulation of rare events like failures of certain components which otherwise would be hard to test on an ordinary test bench.

The main concept of using HIL simulation in power electronic systems is that computational models can substitute the high-power parts of the system. These parts can be the power converter itself (Raihan, 2013) or all other power components on both sides of the converter. For example, in the case of a three-phase inverter, models of the motor (Bachir, 2010; Kokenyesi, 2014) or the filter and grid (Kokenyesi, 2013 IYCE). Simulators are connected through real physical interfaces like analog and digital channels to the control boards under development, so they can be tested and validated in their seemingly real environment. A good HIL simulator is completely transparent for the controller unit, so that the controller is unable to distinguish between the simulator and a real system. Therefore HIL simulation can significantly shorten development time and reduce costs (Suto, 2014). HIL simulators are typically realized using FPGA circuits (Cherragui, 2015).

Nowadays and in the near future, switching frequency of power converters is increasing, especially when silicon carbide semiconductor devices are spreading (Biela, 2011). Time constants of these converters are also getting smaller and smaller. To keep the accuracy of HIL simulators acceptable, simulation time steps need to be decreased as well.

To be able to do this, the detail level of HIL simulation models need to be chosen carefully. For example, time constants of snubber circuits are often much lower than the ones of the main power parts, neglecting them can be a significant reduction in computation demand, so the overall result can be improved. Similar situation can occur with parasitic effects; such as serial resistance and saturation of inductors, ESR or voltage dependency of capacitors, semiconductor voltage drops, etc. In some cases, magnetic saturation can be an important effect, which can be the essence of the control loop, so it needs to be simulated properly (Kokenyesi, 2014).

Another possibility to increase the accuracy is to choose a more complex numeric solver for discretization (Kokenyesi, 2013 IYCE). In this case, a good compromise needs to be found between the method's accuracy and the achievable simulation time step. If fixed-point arithmetic is used in the FPGA, the

precision of each variable is also a critical point. How much can they be reduced to make smaller time step possible? What is the minimal required precision?

The proposed validation method is intended to help in these problems. Modeling deficiency, limited resolution or time step produces some deviation in the output signals of the simulated system compared to the real one. This effect is similar to what is caused by standard production variance of the real model's parameters (Kokenyesi, 2014). A properly designed controller unit compensates this deviation (similarly as disturbance signals) and can work with many different instances of the main circuit. The main concept is to treat the modeling error similarly, if it can be compensated, the simulator is passed the validation test.

2 Validation Methods

2.1 Open-Loop Operation

The first approach which comes into view for validation is the open-loop test, its scheme can be seen in Figure 1. In this case, Model A (which is the reference model or circuit) is operated in closed-loop with a properly tuned controller unit and a PWM generator module. Model B (which is under test) is operated in open-loop and its output signals are compared to the ones from Model A. Model B can be a slightly modified (or simplified) version of Model A. It can differ in parameters (like real main circuits), the simulation time step or fixed-point precision. The error in the output signals shows the modeling error. One possible error calculation method is described in (Kokenyesi, 2013 IYCE), which is actually the RMS value of the error. If this is in an acceptable range, Model B considered to be valid and accepted as a HIL simulation model.

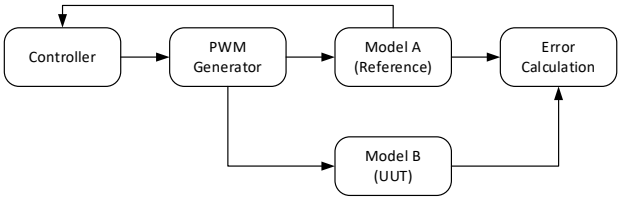

Figure 1. Open-loop validation scheme.

The problem with this method is that the error caused by very small, even negligible model differences is accumulated and it grows continuously as simulation time elapsing. Models with very low damping are especially problematic, e.g. inductors with small serial resistance or resonant circuits with low damping factor. If it was done with two real main circuits, the open-loop operated one would be even damaged after some time.

2.2 Independent Closed-Loop Operation

In this method, the two previous models (A and B) are used again from section 2.1. Both of them are operated with independent controller units with the same structure and tuning, as it can be seen in Figure 2.

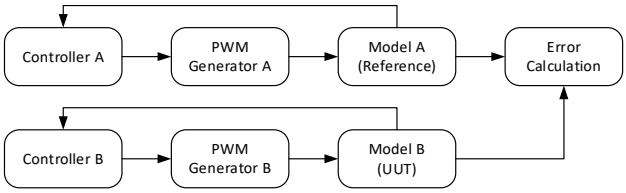

Figure 2. Independent closed-loop validation scheme.

With this approach, accumulated error can be avoided. If a controller is tuned properly for the possible parameter range of the main circuit (or model), it can hide the effect of disturbance signals or model variance and can produce nearly identical output signals. It doesn't mean that all inner variables should be the same. For example, in case of two, slightly different induction motors operated in speed control, the same rotation speed can be achievable with different phase currents. This behavior is one of the main goals in control theory but disadvantageous for this validation, because the simulation error would be eventually hidden.

2.3 Compensated Closed-Loop Operation

The proposed validation method can be seen in Figure 3. Model A is still operated in closed-loop with its properly tuned controller unit. The control signals from the PWM generator are also lead to Model B, with a small intervention based on the output error, which is caused by the difference between the models. The error isn't expected to completely disappear, but it has to be small enough as it would be in the case of two real main circuits. The compensator itself is a special controller unit, which is attempting to reduce the output error to zero. It can modify only the PWM control signals directly, which means inserting switching on or off delays. Otherwise it is similar to the main controller.

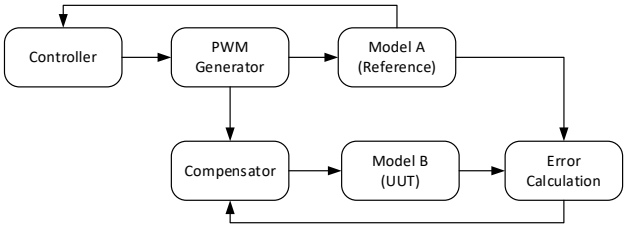

Figure 3. Compensated closed-loop validation scheme.

An important feature of the compensator is that its output is saturated, so the compensable output error is also limited. Switching delays of semiconductors in the real main circuit have a small variance specified in the datasheets. It is a natural difference between the circuit instances, which would be eliminated if they were operated with independent controller units. The controllers would produce slightly different PWM signals, containing the delay variance in this difference. In the validation method, the compensator's output will

contain this variance. If this intervention is saturated to the maximum possible switching delay variance for the semiconductors, the compensator will only be able to eliminate modeling errors, which are less or equivalent to the production variance in effect. If the output error is less than the given tolerance, Model B is accepted as a valid simulation model (Kokenyesi, 2014).

3 Related Work

3.1 Example Circuit

In the following sections, this validation method will be described through a simple example in offline simulation. With two absolutely identical simulation models it is naturally possible to produce the same output signals, so in this case dummy parameter modifications are required to test the validation method.

When choosing the right example, it is important to choose a circuit with very low internal attenuation, which makes it sensitive to proper controller tuning and makes open-loop operation difficult. Taking this into account, a buck converter based battery charger with current control seems to be a good choice. Its schematic can be seen in Figure 4.

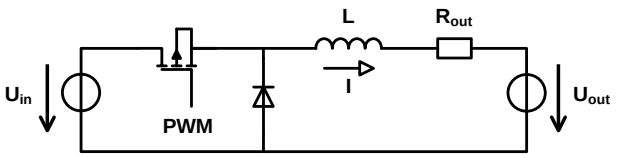

Figure 4. Schematic of the example circuit.

The circuit in Figure 4 contains two DC voltage sources, one for the input DC-link (U_{in}) and another one representing the battery voltage (U_{out}). The semiconductors are considered ideal, only the on and off switching delays (t_{don}, t_{doff}) of the MOSFET are taken into account. Both have a possible minimal and maximal value, which define the acceptable compensator output range. The R_{out} resistance of the L inductor is very small, which makes the attenuation low as required. The control signal is marked as PWM, with f_s switching frequency. The exact parameter values are in Table 1.

Table 1. Parameters of the circuit.

U_{in}	600 V
U_{out}	400 V
I_{ref}	100 A
R_{out}	40 mΩ
L	1.33 mH
f_s	10 kHz
$t_{don,min}$	0.8 µs
$t_{don,max}$	1.2 µs
$t_{doff,min}$	1.8 µs
$t_{doff,max}$	2.2 µs

From these parameters, the switching variance can be calculated:

$$t_{don,diff} = t_{don,max} - t_{don,min} = 0.4\ \mu s, \quad (1)$$

$$t_{doff,diff} = t_{doff,max} - t_{doff,min} = 0.4\ \mu s, \quad (2)$$

$$t_{ddiff} = t_{don,diff} + t_{doff,diff} = 0.8\ \mu s. \quad (3)$$

In the worst case, this t_{ddiff} value is the difference between two instances of the circuit, so it will also be the saturation limit of the compensator.

3.2 Simulation Models

The simulation models of this circuit were built in Matlab/Simulink environment, which is an excellent offline simulation platform, and HDL code generation is also supported for realization of HIL simulators (Suto, 2014). Two different models were made: a floating-point, continuous time model with variable on and off switching delays and a fixed-point discrete time model with variable precision and time step. The two models can be seen in Figure 5 and 6.

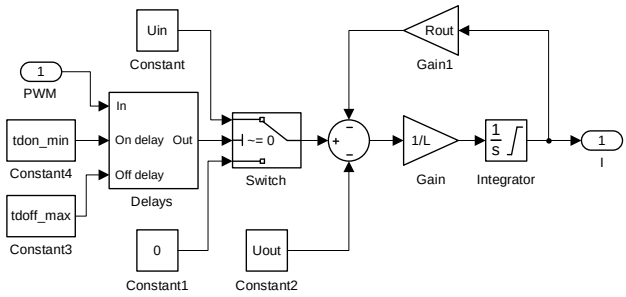

Figure 5. Continuous time model of the example circuit.

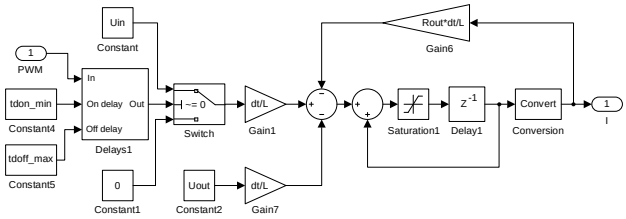

Figure 6. Discrete time model of the example circuit.

Both models contain the same delay generator blocks on the PWM inputs, which allow modifying the switching delays easily. Semiconductors are considered lossless, so a simple switch is used for modeling them. The current integrators are saturated with a lower limit of zero, because of the unidirectional current flowing through the FET and the diode.

In the discrete time model, forward Euler discretization method was used for the integrator (Kokenyesi, 2013 IYCE), where dt is the simulation time step. Because of the fixed-point representation, the integrators' precision are extended to avoid accumulated error (Kokenyesi, 2013 JEPE). The least significant bits are removed with the Convert block after the integrator. Otherwise, the same overall resolution is used for all variables.

For current control, a saturated continuous time PI controller was used, which can be seen in Figure 7. The output saturation ensures that the output voltage doesn't go above the input DC-link voltage and the PWM duty factor remains between 0 and 1. If the saturation is active, there is a difference between this block's input and output signal, which can be used for correction of the integrator and avoid growing its value beyond the limits. It would increase the response time when the sign of the error signal changes next time. When not saturated, this controller operates the same way as any other PI controller, and its tuning can be performed using the traditional methods.

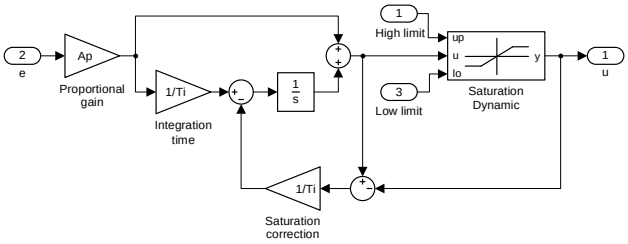

Figure 7. Model of the PI controller.

The compensator itself is also a saturated PI controller with the same structure in Figure 7 and with the same tuning. The difference is only that the saturation limits are calculated from the switching delay variance. The delay differences cause voltage difference, its maximal possible value is the following:

$$U_{diff} = t_{ddiff} f_s U_{in} = 4.8\ \text{V}. \qquad (4)$$

Using this formula, the compensator's voltage output can be converted into delay values, which is used to modify the PWM control signals for the model under validation. In case of positive current error, the compensator produces positive correction output, and the falling edge of the control signal has to be delayed, which causes the FET to switch off later. In case of negative current error, the rising edge and the switching on event has to be delayed, so the current will be reduced.

4 Simulation Results

4.1 Models with Different Switching Delays

First, two continuous time, floating-point models were tested. Differences were only in the switching delays. In the first example, the switching on delay was set to the minimum value in Model A and to the maximum value in Model B. The switching off delays were just the opposite, which means a total t_{ddiff} delay difference. The second example contains two models with switching delay difference $2t_{ddiff}$, which is greater than the allowable maximum.

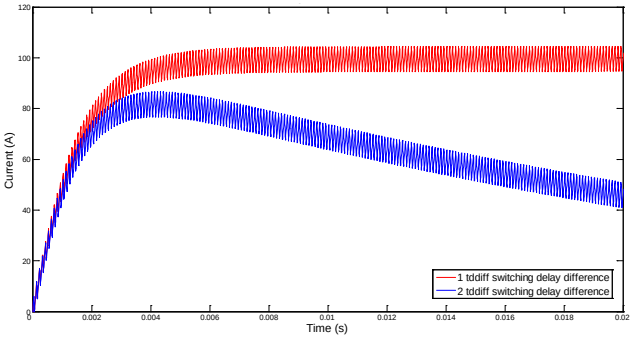

Figure 8. Current signals with different switching delays.

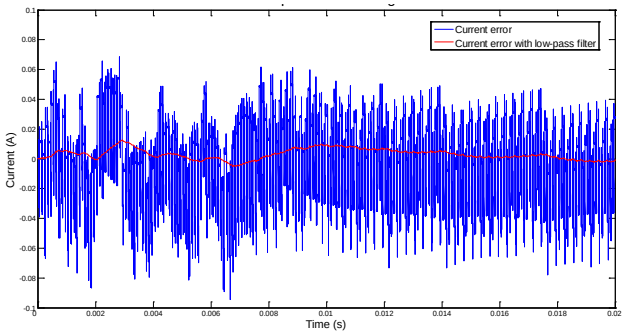

Figure 9. Filtered and non-filtered current error signal (tddiff switching delay).

The simulation results can be seen in Figure 8. The output current signals of Model B in the two simulations are visible. In the case of one t_{ddiff} delay, the current signal can be controlled properly and it is nearly identical to Model A's (which is not in Figure 8). In the other case, the compensator fails to eliminate the current error, which is growing constantly.

It is better to calculate the current error and compare them in different cases than the current signals themselves. In Figure 9 the error is visible in the first case with one t_{ddiff} delay difference. The compensator can only react once every switching period, which causes the switching frequency ripple in the current error. However, it can be stated, that the compensator can keep the current error around zero, additional low-pass filtering of the current error can enhance its visibility as it can be seen in Figure 9. Hereinafter, only the filtered current error signals will be shown.

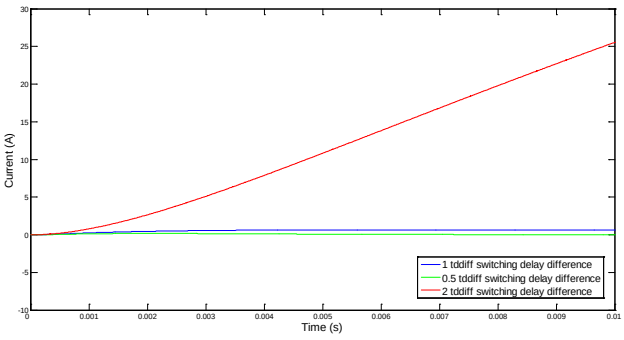

Figure 10. Current error signals with different switching delays.

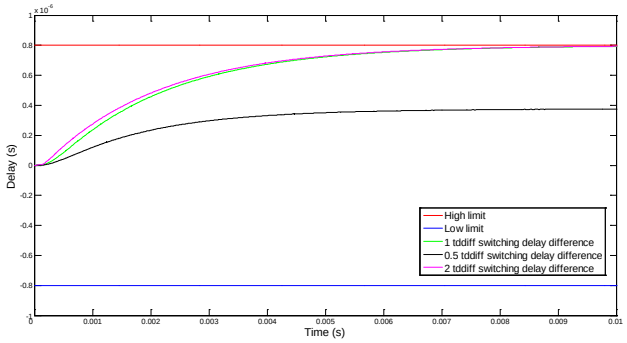

Figure 11. Compensator output signals with different switching delays.

In Figure 10, the filtered current errors are visible in different cases ($t_{ddiff}/2$, t_{ddiff} and $2t_{ddiff}$ delay differences). In the first two example, the current error can be eliminated by the compensator, while in the third one, it is growing. The compensator's output delay is in Figure 11. It is also filtered similarly as the current error. The limits of the compensator's output is also visible. In the case of $t_{ddiff}/2$ delay difference, the intervention in the control signals remains in the allowable range. In the other two cases, it reaches the limit. When the delay is exactly t_{ddiff}, it is just enough to eliminate the error, when it is larger, the error is growing linearly.

These examples were used to test the basic functionality of the compensator and the validation method. It can compensate the original switching delays which correspond to the production variance, so it can be tested on discrete-time models too.

4.2 Discrete-Time Models

When discretizing the model, choosing the simulation time step is critical. This validation method helps to determine the minimum required time step to achieve the required accuracy. It is shown through the following examples.

First, a discrete time model with 100 ns time step was simulated as Model B. Other parameters were the same as the reference continuous time model (Model A) as well as the switching delays. Two additional simulations were run: one with a 2 µs time step and one with a 2.5 µs time step, all other parameters were the same. The general variable precision was 18 bits, uniformly.

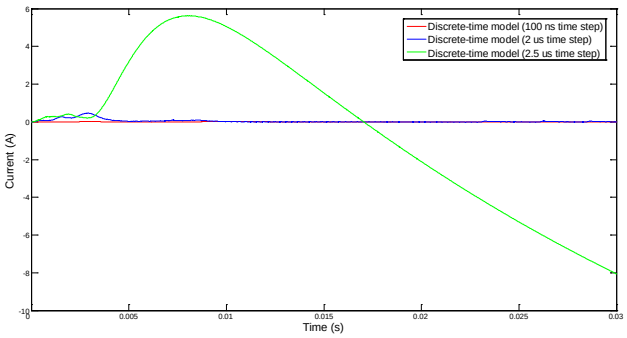

Figure 12. Current error signals with different simulation time steps.

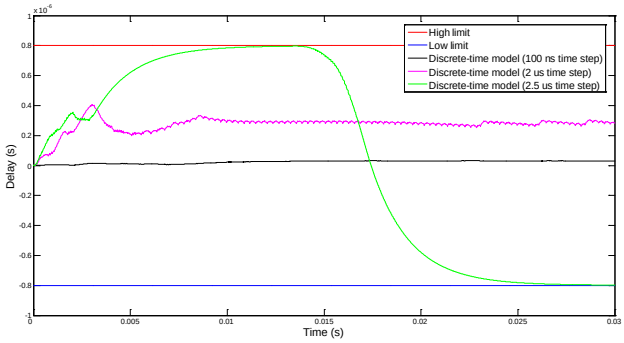

Figure 13. Compensator output signals with different simulation time steps.

The current errors and compensator outputs are in Figure 12 and 13, respectively. In the first two cases, the compensator stays in the normal operating range and the filtered error remains roughly zero. This shows that the model is valid; its simulation error can be compensated as it would be standard production variance.

The last example was simulated with 2.5 µs time step. It is clearly visible that the model is not valid with this time resolution, the current error is growing constantly after an initial transient, while the compensator's output is saturated. It is important to mention that it is not the numerical instability of the forward Euler method (Kokenyesi, 2013 IYCE) which causes the problem, because the system's time constant is 33 ms, which is much higher. The simulation is stable, but not accurate enough.

4.3 Fixed-Point Models

Another important aspect of discretization is to choose an adequate fixed-point representation for the variables. The proposed validation method is also usable to determine the minimum required resolution similarly to the time step.

Three different resolutions were tested as Model B: 18 bits, 9 bits and 8 bits. Like the time step test, all other parameters were the same as the continuous time floating-point Model A. The simulation time step was 100 ns in all cases.

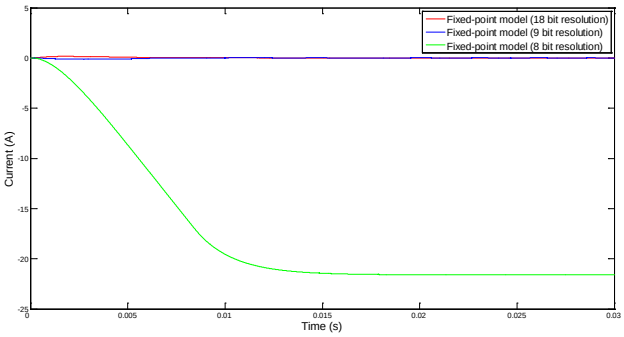

Figure 14. Current error signals with different fixed-point precision.

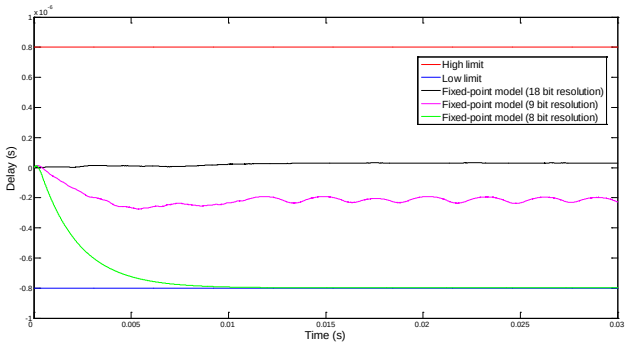

Figure 15. Compensator output signals with different fixed-point precision.

The current errors and compensator outputs are in Figure 14 and 15, respectively. In the case of 18 or 9 bit precision, the compensator eliminates the error successfully with allowable output signals. When using 8 bit precision, the current error stabilizes at -20 A and the compensator is saturated, so 9 bits seems to be the required minimum precision.

5 Conclusions

In this paper, a validation method for HIL simulation models was introduced, treating the simulation errors like the effects of production variance of the main circuit. It considers the simulation model valid, if small intervention in the control signals can compensate the model's error. The limits of the intervention are defined from the production deviation (catalogue data), so there is no essential difference between valid models or real main circuits from the controller unit's aspect. The minimal required simulation time step or fixed-point precision can be determined using this method.

Acknowledgements

This work was performed in the frame of FIEK_16-1-2016-0007 project, implemented with the support provided from the National Research, Development and Innovation Fund of Hungary, financed under the FIEK_16 funding scheme.

References

T. O. Bachir, J. P. David, C. Dufour and J. Belanger. Effective FPGA-based Electric Motor Modeling with Floating-Point Cores. *Proceedings IECON'2010, Glendale (USA)*: 829–834, 2010. doi:10.1109/IECON.2010.5675179.

J. Biela, M. Schweizer, S. Waffler and J. W. Kolar. SiC versus Si—Evaluation of Potentials for Performance Improvement of Inverter and DC–DC Converter Systems by SiC Power Semiconductors. *IEEE Transactions on Industrial Electronics*, 58(7):2872–2882, 2011.

H. Cherragui, M. Hilairet and S. Giurgea. Hardware-in-the-loop simulation of a boost converter with the Xilinx System Generator from Matlab/Simulink. *Proceedings IECON'2015, Yokohama (Japan)*: 1837–1842, 2015. doi:10.1109/IECON.2015.7392368.

T. Kokenyesi and I. Varjasi. FPGA-Based Real-Time Simulation of Renewable Energy Source Power Converters. *Journal of Energy and Power Engineering*, 7(1):168–177, 2013.

T. Kokenyesi and I. Varjasi. Comparison of Real-Time Simulation Methods for Power Electronic Applications. *Proceedings IYCE'2013, Siofok (Hungary)*, 2013. doi:10.1109/IYCE.2013.6604137.

T. Kokenyesi. FPGA-based Real-Time Simulation of a PMSM Drive. *Proceedings AACS'2014, Budapest (Hungary)*: 160–171, 2014.

S. R. S. Raihan and N. A. Rahim. Comparative Analysis of Three-Phase AC-DC Converters Using HIL-Simulation. *Journal of Power Electronics*, 13(1):104–112, 2013. doi:10.6113/JPE.2013.13.1.104.

A. D. Rajapakse, A. M. Gole, P. L. Wilson. Electromagnetic transients simulation models for accurate representation of switching losses and thermal performance in power electronic systems. *IEEE Transactions on Power Delivery*, 20(1):319–327, 2005. doi:10.1109/TPWRD.2004.839726.

Z. Suto, T. Debreceni, T. Kokenyesi, A. Futo and I. Varjasi. Matlab/Simulink Generated FPGA Based Real-time HIL Simulator and DSP Controller: A Case Study. *Proceedings ICREPQ'14, Cordoba (Spain)*: 1–16, 2014.

G. Sybille, H. Le-Huy, R. Gagnon and P. Brunelle. Analysis and Implementation of an Interpolation Algorithm for Fixed Time-Step Digital Simulation of PWM Converters. *IEEE International Symposium on Industrial Electronics'2007, Vigo (Spain)*: 793–798, 2007. doi:10.1109/ISIE.2007.4374698.