JPE 17-6-14

https://doi.org/10.6113/JPE.2017.17.6.1535
ISSN(Print): 1598-2092 / ISSN(Online): 2093-4718

Real-Time HIL Simulation of the Discontinuous Conduction Mode in Voltage Source PWM Power Converters

András Futó[†], Tamás Kökényesi[*], István Varjasi[*], Zoltán Sütő[*], István Vajk[*], Attila Balogh[*], and Gergely György Balázs[**]

[†,*]Dept. of Automation and Informatics, Budapest University of Technology and Economics, Budapest, Hungary
[**]eAircraft Division, Siemens Zrt, Budapest, Hungary

Abstract

Advances in FPGA technology have enabled fast real-time simulation of power converters, filters and loads. FPGA based HIL (Hardware-In-the-Loop) simulators have revolutionized control hardware and software development for power electronics. Common time step sizes in the order of 100ns are sufficient for simulating switching frequency current and voltage ripples. In order to keep the time step as small as possible, ideal switching function models are often used to simulate the phase legs. This often produces inferior results when simulating the discontinuous conduction mode (DCM) and disabled operational states. Therefore, the corresponding measurement and protection units cannot be tested properly. This paper describes a new solution for this problem utilizing a discrete-time PI controller. The PI controller simulates the proper DC and low frequency AC components of the phase leg voltage during disabled operation. It also retains the advantage of fast real-time execution of switch-based models when an accurate simulation of high frequency junction capacitor oscillations is not necessary.

Key words: Discontinuous Conduction Mode (DCM), Field-Programmable Gate Array (FPGA), Hardware-In-the-Loop (HIL), Power converter, Real-time, Simulation

I. INTRODUCTION

General power converter systems consist of two main parts: a power level (main) circuit, and one or more controller units usually realized using a DSP or FPGA. Since testing such units on the original main circuit is expensive and dangerous, computer simulation is often used instead [1], [2].

A. Real-Time HIL Simulation in Power Electronics

An effective way to test controller hardware and software is real-time HIL (Hardware-In-the-Loop) simulation [3]. The main concept behind HIL simulation is that real-time computational models replace the high-power parts of the system, including power converters [4] and all other power and analog measurement components on both sides of the converter, as shown in Fig. 1. For example, in case of an inverter, the components on both the DC side [5]-[7] and the AC side (motor [8] or filter and grid model) are simulated. The simulator hardware is connected to the control board under development via real physical interfaces (analog and digital I/O). As a result, the control card and its software are unable to distinguish the simulator from a real system. Such a simulator can shorten development time and reduce costs [9].

The hardware used for real-time simulation may consist of microprocessor or DSP cores. This is useful for the HIL simulation of slow [10] or very large systems [11]. However, FPGA circuits are the core units of very complex [12] and fast HIL simulators [13], [14]. Modern FPGAs provide high speed interfaces, low latency and smaller time steps than microprocessors thanks to their paralleled structure [15]. As a result, large FPGAs are usually a better choice for the HIL simulators of power converters than DSPs or personal computers [8], [16].

Manuscript received Dec. 21, 2016; accepted Jun. 28, 2017
Recommended for publication by Associate Editor Se-Kyo Chung.
[†]Corresponding Author: Futo.Andras@aut.bme.hu
Tel: +36-1-463-1552, Budapest Univ. of Technology and Economics
[*]Dept. of Automation and Informatics, Budapest University of Technology and Economics, Hungary
[**]eAircraft Division, Siemens Zrt, Hungary

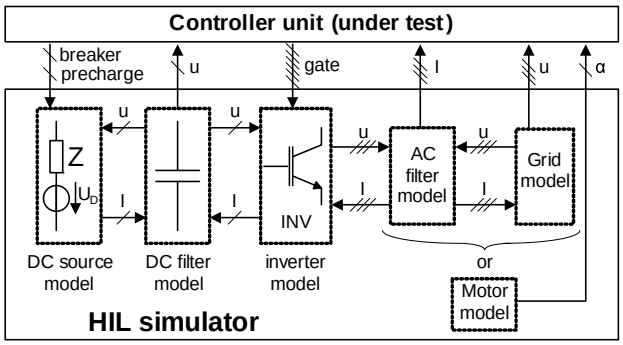


Fig. 1. Modular HIL block diagram for three phase inverters. The arrows show the direction of the data exchange.

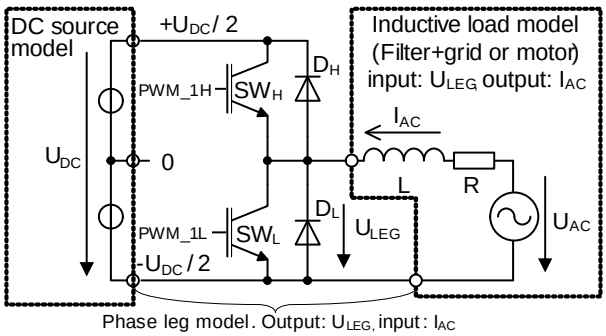


Fig. 2. Schematic of a typical inductively loaded 2-level phase leg.

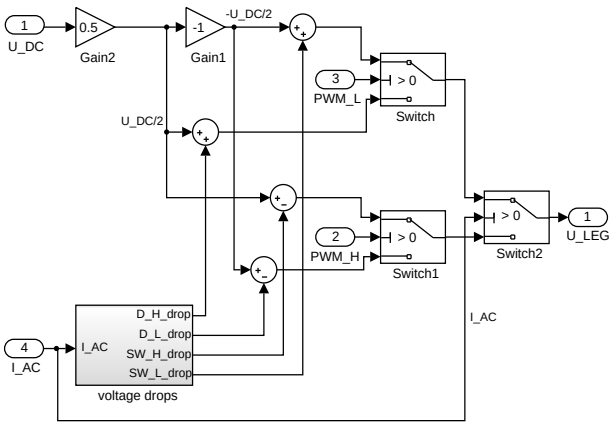


Fig. 3. Switching function based phase leg model in Matlab Simulink. The model also includes voltage drop simulation.

B. Modelling of Phase Legs

The main building block of most power converters is the inductively loaded phase leg, as shown in Fig. 2 for a two-level grid connected inverter. It consists of a pair of diodes clamping the switched point of the phase leg (marked as U_{LEG} in Fig. 2) to a DC bus, along with controlled semiconductor switches capable of connecting the switched point to certain fixed potentials, e.g. +Udc/2 and -Udc/2 for a two-level inverter.

Several very precise power semiconductor models exist for the described phase leg and its components. Some concentrate on the voltage-current characteristics and switching behavior of power semiconductors [17], [18]. Others try to implement idealized or identified characteristics [19] or to use parameter extraction and detailed physics based semiconductor models [20], [21]. Such models are very accurate. However, they are computationally intensive, and can hardly be optimized for real-time use. The fastest real-time simulators usually need simpler models using a forward Euler solver.

The simplest is the switching function model. This model can only chose one of the possible voltage levels in each time step, or it may switch between two resistance values. Such a model is very fast. However, it cannot handle the discontinuous conduction mode (DCM) [13]. The problem can be solved via simulating a capacitance in place of an open switch [22]. However, this can lead to stability problems, which necessitates the use of a backward Euler solver and further simplifications such as the associated discrete circuit (ADC) model [23]. The extra computations increase the delay of the model which in turn lengthens the time step.

The goal of the actual research described in this paper is to provide a model which combines the advantages of the two methods described above: the ability to simulate the DC and low-frequency voltage components during DCM, and stability with a forward Euler solver.

The problems with existing models are first discussed in Section II. Section III describes the new phase leg model. Section IV presents simulation results. The simulation results are compared with experimental results in section V. Some conclusions are presented in section VI.

II. REAL-TIME PHASE LEG MODELS

As described in the introduction, semiconductors can be modeled via simple switching functions. This is often used in real-time simulators, and sometimes in offline simulators [24]. Fig. 3 shows how this works. If the high side control is high, a positive voltage is forced to U_{LEG}. If the low side control is high, a negative voltage is forced. If both control signals are low, the sign of the forced voltage is chosen based on current direction. Voltage drop simulation is also possible as shown in Fig. 3. The problems of simulating DCM on switching function models are discussed in part A. Improving the model via junction capacitor simulation is discussed in part B.

A. Switching Function Based Models

In a phase leg working in a PWM converter, there are two situations when both switch control signals can be low: the uncontrolled state (the appliance is in standby or inoperative mode) and the control dead time (introduced to avoid short circuit caused by a delay of the semiconductor switches). DCM happens if the current is low before control turn-off.

In switching function models, if both transistors are OFF in a phase leg, the simple diode model chooses its output voltage for the actual time step based on the current direction. This is done by the "Switch2" block in Fig. 3. Long-time inactive

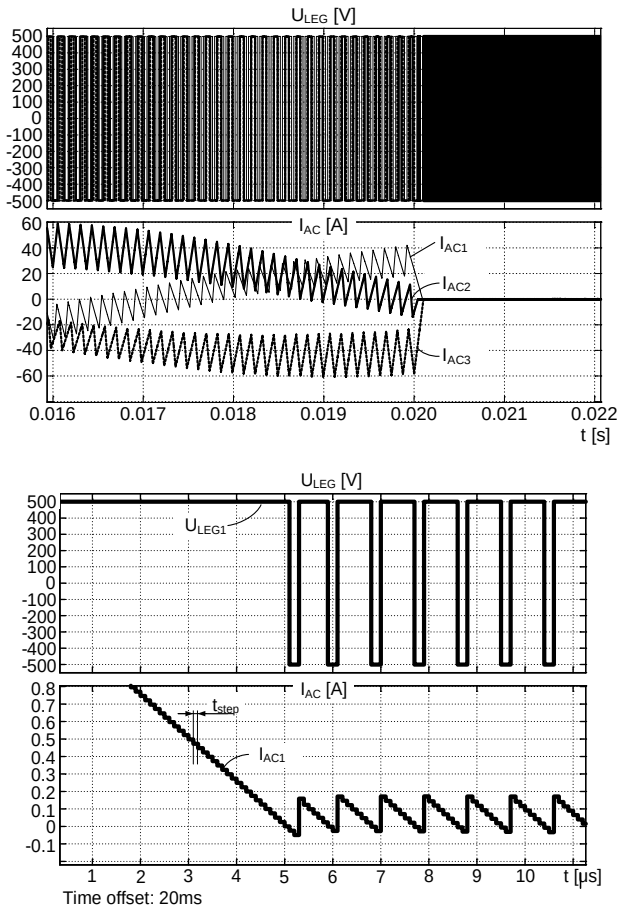


Fig. 4. 3-phase inverter waveforms of a switching function based fixed step model. Transition to DCM is magnified for leg 1.

control signals may let the current fall to zero, which causes the output voltage to alternate between the positive and the negative DC rails. In a variable-step solver, simulating the DCM with a switching function model causes the time step to decrease to its minimum setting. This results in a serious slow-down of the simulation. Extra logics and communication links between the modules (see Fig. 1) are needed to handle zero crossings, which degrade the reusability of the modules. Matlab/Simulink has tools to handle this problem in off-line simulations (e.g. the state ports of integrators cooperating with hit crossing blocks). However, these are not supported by HDL code generation, rendering the solution impractical with variable-step solvers.

Fixed step solvers do not suffer from such problems. However, a switching function model operating in DCM on a fixed step solver generates a false high-frequency phase leg voltage waveform, which has a period proportional to the time step. Such waveforms can be seen in Fig. 4. The minimal pulse width of this high frequency noise is one simulation time step.

The switching operation causes small ripples in the inductor current, which are visible on the bottom waveform of Fig. 4. The average of the simulated phase leg voltage, seen in the diagram, is still correct. It is equal to u_{AC} (Fig. 2).

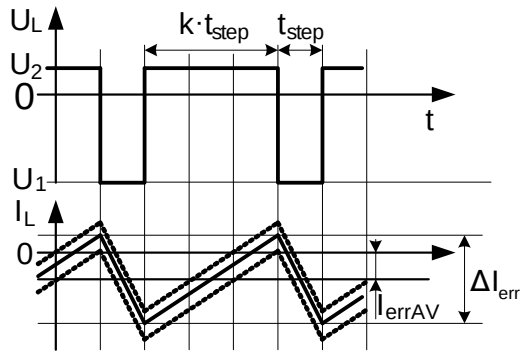


Fig. 5. DCM inductor voltage and current waveforms of a switching function based model.

There are problems with common mode currents and energy balance as a result of the simulation error described above. A small triangular current always flows through the inductively loaded phase leg during DCM. The typical inductor voltage and phase leg current waveforms caused by this can be seen in Fig. 5. In the equations below, U_1 and U_2 represent the two possible voltage values of the filter inductor L based on the current direction, with the following assumption:

$$|U_1| > |U_2| \tag{1}$$

The ripple of the error current can be calculated as:

$$\Delta I_{err} = \frac{|U_1| \cdot t_{step}}{L} = \frac{|U_2| \cdot k \cdot t_{step}}{L} \tag{2}$$

Equation (2) enables the calculation of the time ratio k in (3) and the noise frequency f_{err} in (4). k is a whole number. Thus, it needs to be rounded up, as shown by the operator $\lceil\ \rceil$ in (3).

$$k = \left\lceil \frac{|U_1|}{|U_2|} \right\rceil \tag{3}$$

$$f_{err} = \frac{1}{(1+k) \cdot t_{step}} \tag{4}$$

Based on Fig. 5 and (3), it is clear that the average simulated inductor current is nonzero if $U_1 \neq U_2$. The actual offset of the current may vary between periods, since the actual zero crossing might happen anytime up to one time step before the simulator recognizes the change of sign. Therefore, the offset of the triangular current can change between its minimum and maximum value, as shown with the dotted lines in Fig. 5. The average error current can be calculated as follows:

$$|I_{errAV}| = \frac{\Delta I_{err}}{2} - \frac{|U_2| \cdot t_{step}}{2 \cdot L} = \frac{(|U_1| - |U_2|) \cdot t_{step}}{2 \cdot L} \tag{5}$$

Note that for a multiphase converter, the DC current is this number multiplied by the number of phase legs. The actual sign of the average error current depends on the ratio of the voltages present on the two sides of the simulated converter, and on the reference directions of the currents and voltages.

The problem is that a false square wave phase leg voltage and current ripple makes it hard to test certain protection hardware and software. This includes hardware based dead

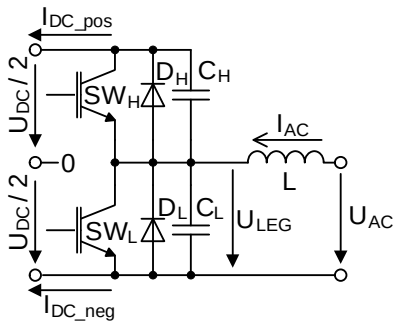


Fig. 6. Modeled capacitances of a phase leg.

time compensators using phase leg voltage measurement, back EMF measurement during DCM (e.g. sensorless BLDC drives), and leakage current protection (electric car chargers). A major problem is that if a DC capacitor model is attached to the phase leg model, the rectified false current ripple constantly charges the input capacitor. The resulting error becomes significant if the control of the phase leg is disabled for several milliseconds. This can falsely trigger DC overvoltage protection.

B. Simulation of junction capacitance

Adding a junction capacitor model, as shown in Fig. 6, can provide an acceptable phase leg voltage. For AC currents, the DC grid can be substituted with a short circuit. Thus, only one capacitor model is required for each leg. A saturated integrator can be used instead of the switch-based model, and the gate signals and current direction can be used to determine the saturation limits. The integrator used in the test phase leg model only simulates the diode leg operation. It is reset if any of the switches are on. Using the capacitor model, the DC currents on both the positive and negative DC rails can be easily calculated:

$$I_{DC\_pos} = I_{AC} \cdot \frac{1}{1+\frac{C_L}{C_H}} \tag{6}$$

$$I_{DC\_neg} = I_{AC} \cdot \frac{1}{1+\frac{C_H}{C_L}} \tag{7}$$

Using the saturated integrator model for the calculation of the DC currents solves the offset problem described in section II.A. The operation of the model can be followed based on the example of a three phase grid connected inverter (Fig. 7). After disabling the gate signals, a decaying high frequency oscillation appears on each of the phase legs superimposed onto the fundamental AC voltage. The frequency is so high that individual periods cannot be seen in Fig. 7. However, the resulting waveforms are realistic and enable the testing of protection and control units. The capacitor model can also be used to estimate the switching losses of the bridge if it is supported by the structure of the HIL. This can be done by subtracting the lost energy from the value of the integrator in the DC capacitor model. The structure shown in Fig. 1 does not

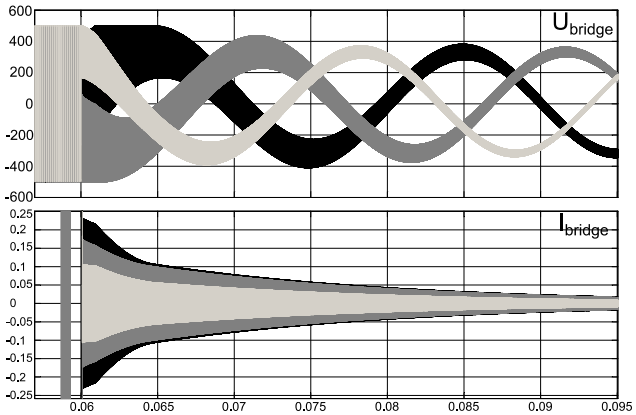


Fig. 7. Junction capacitor model: voltage and current waveforms of a three phase inverter after disabling the gate signals.

allow this. The time step in FPGA-based HIL simulators is usually too long for junction capacitance simulation. However, this method is still useful in offline models.

III. The Proposed Real-Time Phase Leg Model with a PI Controller

In an FPGA-based HIL simulator, the ordinary differential equations (ODE-s) of the dynamic components need to be solved in real-time. This is done using a small simulation time step and low latency. To satisfy these criteria, a simple fixed-step solver like the Forward Euler method is advantageous. More complex methods such as a backward Euler or a Runge-Kutta solver mean a much higher computational burden for the FPGA [25].

The Forward Euler method has accuracy and stability problems when simulating second order oscillating elements [26] [27]. This is a problem in the case of the junction capacitor-filter inductor circuit described in the previous section, since the period of the high frequency oscillation may be shorter or not much longer than the time step. Very accurate real-time simulation of these oscillations is practically hard, since the value of the capacitances are not constant, and often not accurately known. However, accurate simulation of this is usually not needed. The frequency is generally so high that it is instantly filtered by the analog input stage of the controller. As a result, direct junction capacitor modeling is rarely used in HIL apart from special applications like ZVS resonant bridges [28], where large external capacitances are paralleled to semiconductors. In other cases, an approximation of the DC the DC and low frequency components of the phase leg voltage are sufficient if the phase leg is not controlled and the current becomes discontinuous.

In order to solve the problems of the switching function based model shown in Fig. 4, a moving averaging stage can be added after the current and phase leg voltage outputs of the model. This gives the correct phase leg voltage. However, it also increases the delay of both the voltage and current signals,

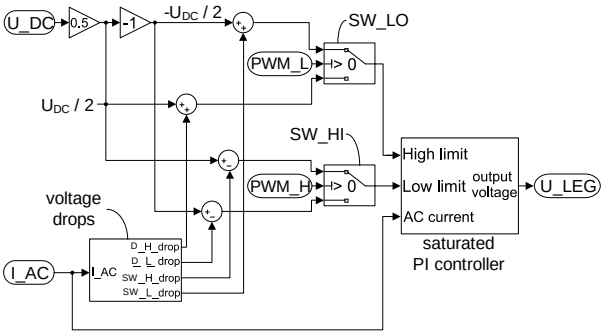


Fig. 8. Bridge arm model structure using a saturated PI controller.

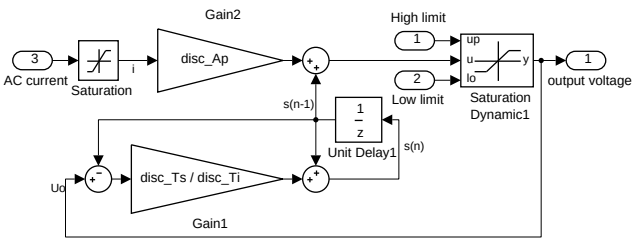


Fig. 9. Structure of the saturated PI controller in the simulator.

and it is unable to remove the DC component of the current. Another method is described in [29] using a resistor divider. This models IGBTs correctly. However, it needs an iterative algorithm for diode simulation.

The proposed method uses a discrete-time PI controller. The structure of the bridge model can be seen in Fig. 8. When both of the switches are off and the current has decreased to zero, the PI controller automatically finds the correct phase leg voltage required to keep the current at zero. The calculated phase leg voltage follows the voltage of the grid or the motor back EMF as long as the PI controller is not saturated. The output saturation limits are set to the full DC bus voltage range if both of the switches are off. The voltage drops of the diodes can also be modeled. If the control signal of a switch becomes high, the saturation limits are modified so that the output voltage remains in the range allowed by the corresponding switch and diodes. This operation may also be extended to multilevel converters by correctly choosing the low and high limits from the possible values based on the actual state of the switching semiconductors.

The structure of the actual PI controller can be seen in Fig. 9. The implementation of the saturation uses the automatic reset configuration of the PI controller, which enables a simple anti-windup scheme to be used [30]. In this case, the integral action is implemented by applying positive feedback with a first-order low-pass filter. The implementation seen in Fig. 9 was used to avoid the oscillations arising from rapid current changes in continuous conduction mode. This realization imitates the behavior of a series RC element similar to that of the circuit shown in Fig. 10. The described PI structure is actually mathematically identical: R_{sim} can be substituted for

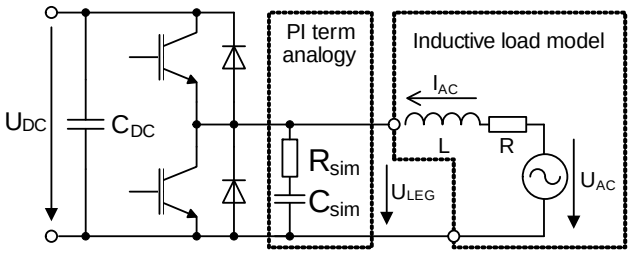


Fig. 10. Circuit emulated by the PI structure in Fig. 9.

disc_Ap and $R_{sim} \cdot C_{sim}$ can be substituted for the time constant *disc_Ti* in Fig. 9. The saturation is placed directly onto the output of the PI controller, which is equivalent to the phase leg voltage. The input value of the integrator resembles the input current of the capacitor, and it is proportional to the difference between the integrator value and the saturated output voltage.

To tune the PI controller, some information is required on the rest of the power circuit. This information includes the approximate inductance value during the DCM (L_{disc}), the DC voltage and the simulation response time (i.e. the control system dead time). Since two parameters are required, the inverter model cannot be completely independent from the load model as described in Fig. 1. However, these parameters are not required to be accurate (the high frequency oscillations of the junction capacitance voltages cannot be accurately simulated with long time steps anyway). The only goal is to keep the system stable. The value of L_{disc} is the approximate net inductance seen by the AC side of the converter. Iron core saturation does not need to be included in the calculation, since the PI controller only operates at nearly zero currents. Too high of an L_{disc} value might cause divergent operation, and too small values might lead to oscillations. The simulation response time (t_{SIM_RESP}) needs to be equal to $t_{step}/2$ (the average delay caused by the integrator in the PI controller) plus any other delay present in the controlled circuit. The formulas for calculating the PI controller parameters can be seen in (8)-(12). The tuning described below is for maximum disturbance rejection. However, other methods (e.g. Ziegler-Nichols) may be used with success. The calculation leaves a pre-defined phase margin φ_0 and uses up the remaining phase shift by leaving 2/3 of it for the dead time component and 1/3 for the integrator:

$$\varphi_0 = \frac{\pi}{2} - \varphi_{\text{margin}} \tag{8}$$

$$\omega_c = \frac{2}{3} \cdot \frac{\varphi_0}{t_{\text{SIM\_RESP}}} \tag{9}$$

$$disc\_T_i = \frac{1}{\omega_c \cdot \tan\left(\frac{1}{3}\varphi_0\right)} \tag{10}$$

The A_p parameter was calculated so that the absolute open loop transfer of the current loop is 1 at ω_C frequency:

$$disc\_A_p = \omega_c \cdot L_{\text{disc}} \tag{11}$$

It can be seen that Fig. 9 contains a saturation block directly

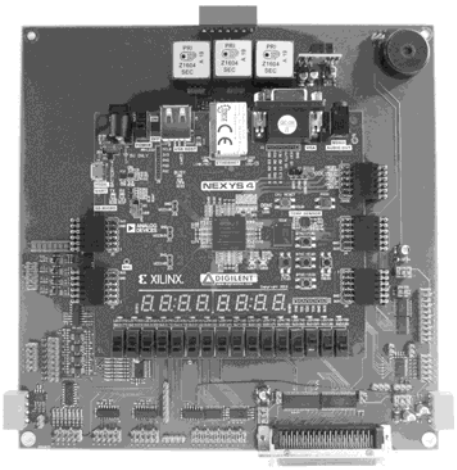


Fig. 11. HIL simulator hardware used for testing.

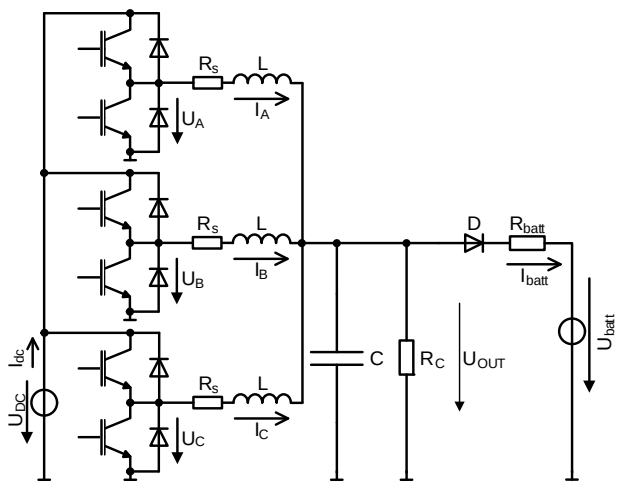


Fig. 12. Main circuit of the battery charger used for testing.

on the input of the PI controller. The I_{lim} value is defined as:

$$I_{\lim} = \frac{U_{DC}}{disc\_A_p} \tag{12}$$

I_{lim} is used to limit the input current of the PI controller so that it can cause no more than a U_{DC} voltage difference in the next time step. This can be allowed since any input current higher than I_{lim} has the same effect because of output saturation. The advantage of this is a reduction in the number of bits required for the fixed-point PI controller. A fixed-point numerical representation has many advantages in FPGAs and is widely used in HIL simulators [4] [13].

IV. Simulation Results

The HIL simulator hardware used for testing the new method can be seen in Fig. 11. Its central part is a Xilinx Artix7 XC7A100T FPGA on a Nexys4 card. The Nexys4 connects to a custom designed interface card, which converts the FPGA I/O banks to the signal levels of the control hardware being tested. Sigma-delta D/A converters, CAN and RS422 interfaces, and a resolver emulator interface are also available.

TABLE I

Parameters of the Battery Charger Test System

U_{DC}	U_{batt}	f_{sw}	L	Rs	C	R_C	R_{batt}	t_{step}
600V	450V	10kHz	2.5mH	50mΩ	1.1mF	4.7kΩ	0.2Ω	0.1μs

TABLE II

Parameters of the Simulator

disc_ts	φ_{margin}	t_{sim_resp}	ω_c	disc_T_i	disc_A_p	I_{lim}
t_{step}	π/3 rad	50ns	6.98Mrad/s	812 ns	17.5kΩ	0.23A

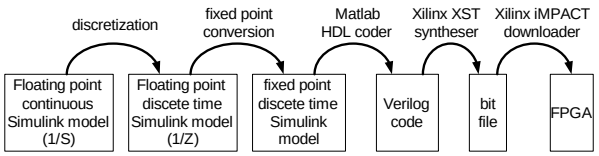
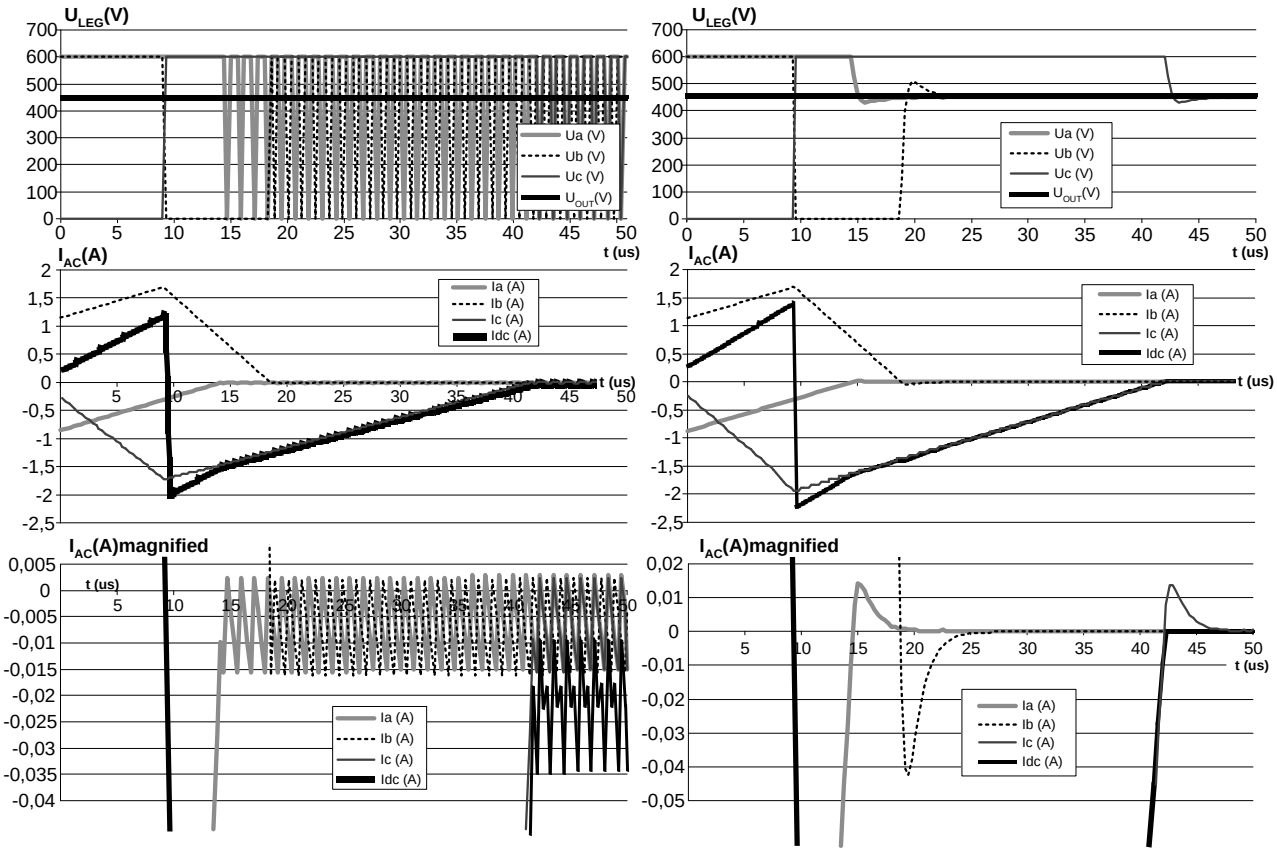


Fig. 13. HIL workflow.

The modeled main circuit is shown in Fig. 12. It is a battery charger circuit consisting of a three-phase synchronous buck converter. The input is an ideal DC source, and the load is the Thevenin-equivalent of a battery via a diode to prevent it from discharging. The parameters of the test system are presented in table I. The calculated parameters of the PI controller are presented in table II.

The complete HIL workflow used for the simulations can be seen in Fig 13. At first, the modeling is performed in Matlab Simulink using continuous time floating point blocks. This model is useful for offline testing. The next step is discretization, where the continuous-time integrators are replaced with discrete-time integrators. The resulting model can also be checked via offline simulations. When the discrete model is complete, the 64-bit floating point state variables are replaced via fixed-point variables, which work better on an FPGA. The resulting model is also tested in Simulink. It is then converted to Verilog code via the Matlab HDL coder to be synthesized and downloaded into the FPGA. The proposed model of the circuit in Fig. 12 is calculated in every 100ns time step. The time step is sufficient. As a result, latch (delay) blocks are only required before the lookup tables and in the integrators of the L and C components and the PI controller, as shown in Fig. 9. The model takes advantage of Xilinx DSP48E slices and uses 18-bit fixed point notation for signals. The currents are represented with 11-bit, and the voltages are represented with 21-bit fractional part. The accumulators use more bits to avoid data loss from small input values. The model has been translated into Verilog via the HDL Coder as a part of a complete HIL system to be run on the FPGA [9]. Monitoring was performed via Xilinx ChipScope Pro tool which allows for the on-line modification of model parameters.

Two IGBT phase leg models were tested real-time: the switch based model of Fig. 3 and the saturated PI controller based model of Fig. 9 and Fig. 10. The control of all six simulated switches were disabled a little before t = 10μs. After



(a) Switch-based model results. (b) PI controller based model results.

Fig. 14. Comparison of simulated voltage and current waveforms of the circuit in Fig. 12 after disabling the IGBTs at t = 10μs.

this, all of the phase legs went discontinuous at different times depending on the actual current at the moment of disabling the control. The results are shown in Fig. 14. The result of the simple model is the current ripple and DC capacitor charging, as described in section II. The mean of the phase currents seen in Fig. 14(a) are negative due to the difference in the voltages on the two sides of the inductor in the two possible switching states. The current reference direction in Fig. 12 is the opposite of what it was in Fig. 2. This means that the DC-link capacitor is charged by 20mA on average. Simulated waveforms of the saturated PI based model can be seen in Fig. 14(b). The currents are controlled to zero very quickly, while phase leg voltages stabilize at the voltage of the output capacitor. The resulting DC current I_{DC} also becomes zero, which solves the DC link capacitor charging problem.

V. Comparative Experimental Results

The simulated outputs of Fig. 14(b) have been compared to the response of a real system to the same switch-disable event. To perform this, the battery charger test circuit shown in Fig. 12 was used. This is part of a battery charger test system built using a modified Siemens Sinamics S120 motor module (MM) which contains IGBT based phase legs and their gate drive

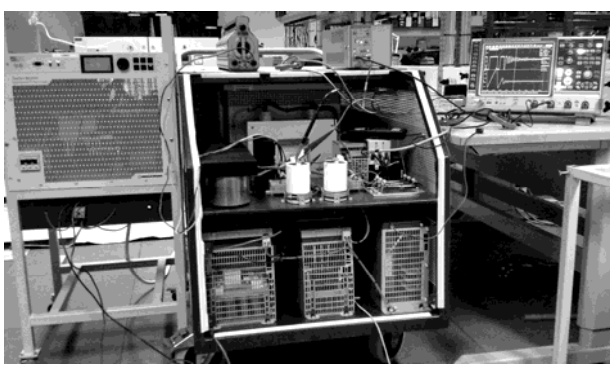


Fig. 15. Test system. The S120 MM module is on the bottom left of the cart. The inductors and output capacitors are on the top.

circuits. Fig. 15 shows the test system. Apart from the MM module, the bottom of the cart contains additional Sinamics modules not used during the test. The power and sensing circuits of the S120 MM were retained, and all of the extra L, C, R_c and D components were added externally to the top of the cart. The system was controlled via a custom made control card and software developed for research purposes. The input side DC bus of the MM has been fed externally via a 600V isolated DC power supply (Fig. 15 left, Regatron TC.P.20.1000.400.S). The output has been loaded with a 328 Ω

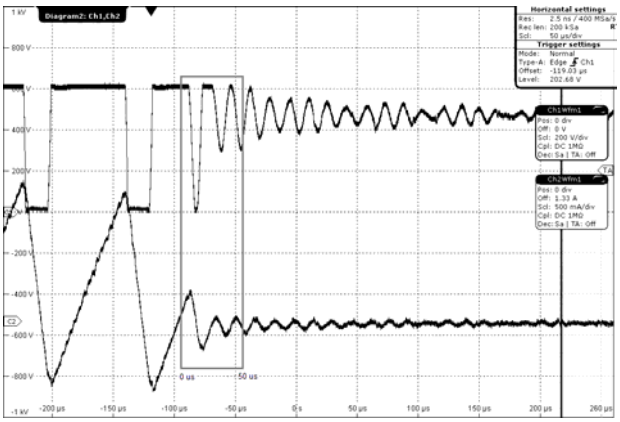


Fig. 16. Voltage (top) and current (bottom) measured on the test system. The time range seen in Fig. 14 is marked with a bracket.

resistor to ensure a load for the proper operation of the control software. The output voltage was 450V when the gate drivers were disabled. The inductor current and phase leg voltage of one phase has been recorded on a Rhode & Schwarz RTO1004 oscilloscope using a Tektronix P5200 differential voltage probe and a Tektronix A6302 DC/AC current probe on its AM503A amplifier. Fig. 16 shows the measurement results. The IGBTs are disabled at approximately -87µs. Diode conduction is continued until the current reaches zero. After this point, the phase leg voltage converges to the 450V output voltage. The resulting waveforms near the switch-over (the rectangular bracket in Fig. 16) show approximately the same time frame as those shown for phase "b" in Fig. 14(b). Immediately after switch-off, the real measurement results show the high frequency decaying oscillation described in section II.B. This is not reproduced by the PI controller based simulator. However, after the transition, the final current and voltage values are in agreement with the simulation results.

VI. CONCLUSIONS

With the emergence of cheap high-speed FPGAs, the real-time simulation of power electronics has become a reality. The goal of model building for HIL simulators is to compute results that are as realistic as possible for the time step sizes allowed by the FPGA hardware. With this goal in mind, this paper presents a new type of inverter bridge model. The new method is capable of simulating the discontinuous conduction mode (DCM) and the switched-off operation without using time steps smaller than the oscillation frequency of the bridge junction capacitance and its load inductance. This simple and fast model returns the correct fundamental frequency and DC component of the output voltage and current. In addition to the long-time switched-off operation, the new model also gives proper results in rectification mode ($U_{LEG} > U_{DC}$). It can also simulate discontinuous conduction during control dead-time.

In practice, the new model can be synthesized and used in real-time FPGA based HIL simulators with the same time step as the switching function based model. This time step is small enough to enable the use of a simple Forward Euler solver for most of the switching frequencies in silicon IGBT based high power converters. Testing in off-line simulation has also shown that the model is fast enough to simulate dead-time events such as the DCM during dead time.

ACKNOWLEDGMENT

This work was partially supported by the Hungarian Government, managed by the National Development Agency, and financed by the Research and Technology Innovation Fund through project eAutoTech (grant no.: KMR 12-1-2012-0188). The authors wish to thank the support to the Hungarian Research Fund (OTKA K100275) and the Control Research Group of the Hungarian Academy of Sciences.

REFERENCES

[1] A. M. Gole, A. Keri, C. Nwankpa, E. W. Gunther, H. W. Dommel, I. Hassan, J. R. Marti, J. A. Martinez, K. G. Fehrle, L. Tang, M. F. McGranaghan, O. B. Nayak, P. F. Ribeiro, R. Iravani, and R. Lasseter, "Guidelines for modeling power electronics in electric power engineering applications," *IEEE Trans. Power Del.*, Vol. 12, No. 1, pp. 505-514, Jan. 1997.

[2] H. Jin, "Behavior-mode simulation of power electronic circuits," *IEEE Trans. Power Electron.*, Vol. 12, No. 3, pp. 443-452, May 1997.

[3] T. Kokenyesi and I. Varjasi, "FPGA-based real-time simulation of renewable energy source power converters," *Journal of Energy and Power Engineering*, Vol. 7, No. 1, pp. 168-177, Jan. 2013.

[4] S. R. S. Raihan and N. A. Rahim, "Comparative analysis of three-phase AC-DC converters using HIL-simulation," *Journal of Power Electronics*, Vol. 13, No. 1, pp. 104-112, Jan. 2013.

[5] C. S. Edrington, O. Vodyakho, and B. A. Hacker, "Development of a unified research platform for plug-in hybrid electrical vehicle integration analysis utilizing the power hardware-in-the-loop concept," *Journal of Power Electronics*, Vol. 11, No. 4, pp. 471-478, Jul. 2011.

[6] J-H. Jung, "Real-time and power hardware-in-the-loop simulation of PEM fuel cell stack system," *Journal of Power Electronics*, Vol. 11, No. 2, pp. 202-210, Mar. 2011.

[7] T. Debreceni, P. Szabo, G. G. Balazs, and I. Varjasi, "FPGA-synthesizable electrical battery cell model for high performance real-time algorithms," *Periodica Polytechnica. Electrical Engineering and Computer Science*, Vol. 60, No. 3, pp. 163-170, 2016.

[8] T. O. Bachir, J. P. David, C. Dufour, and J. Belanger, "Effective FPGA-based electric motor modeling with floating-point cores," in *36th Annual Conference on IEEE Industrial Electronics Society (IECON)*, pp. 829-834, Nov. 2010.

[9] 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," in *International Conference on Renewable Energies and Power Quality (ICREPQ)*, Vol. 1, No. 12, pp. 431-436,

Apr. 2014.
[10] J.-H. Jeon, J.-Y. Kim, H.-M. Kim, S.-K. Kim, C. Cho, J.-M. Kim, J.-B. Ahn, and K.-Y. Nam, "Development of hardware in-the-loop simulation system for testing operation and control functions of microgrid," *IEEE Trans. Power Electron.*, Vol. 25, No. 12, pp. 2919-2929, Dec. 2010.
[11] C. Liu, P. Tian, Y. Wang, Q. Guo, X. Lin, and J. Wang, "A hardware-in-the-loop platform for modular multilevel converter simulations," *Journal of Power Electronics*, Vol. 16, No. 5, pp. 1698-1705, Sep. 2016.
[12] J. Liu and V. Dinavahi, "Detailed magnetic equivalent circuit based real-time nonlinear power transformer model on FPGA for electromagnetic transient studies," *IEEE Trans. Ind. Electron*., Vol. 63, No. 2, pp. 1191-1202, Feb. 2016.
[13] M. Matar and R. Iravani, "FPGA implementation of the power electronic converter model for real-time simulation of electromagnetic transients," *IEEE Trans. Power Del.*, Vol. 25, No. 2, pp. 852-860, Apr. 2010.
[14] G. G. Parma and V. Dinavahi, "Real-time digital hardware simulation of power electronics and drives," *IEEE Trans. Power Del.*, Vol. 22, No. 2, pp. 1235-1246, Apr. 2007.
[15] J. J. Rodríguez-Andina, M. D. Valdés-Peña, and M. J. Moure, "Advanced features and industrial applications of FPGAs—A Review," *IEEE Trans. Ind. Informat.*, Vol. 11, No. 4, pp. 853-864, Aug. 2015.
[16] J. C. G. Pimentel and H. Le-Huy, "Hardware emulation for real-time power system simulation," in *IEEE International Symposium on Industrial Electronics (ISIE)*, pp. 1560-1565, Jul. 2006.
[17] L. Yong, C. Tianqing, B. Fan, and C. Yulin, "Switching characteristics simulation of IGBT based on FPGA," in *International Conference on Computational Problem-Solving (ICCP)*, pp. 386-391, Dec. 2010.
[18] A. Myaing and V. Dinavahi, "FPGA-based real-time emulation of power electronic systems with detailed representation of device characteristics," in *IEEE Power and Energy Society General Meeting*, Jul. 2011.
[19] G. G. Parma and V. Dinavahi, "Real-time digital hardware simulation of power electronics and drives," *IEEE Trans. Power Del.*, Vol. 22, No. 2, pp. 1235-1246, Apr. 2007
[20] W. Wang, Z. Shen, and V. Dinavahi, "Physics-based device-level power electronic circuit hardware emulation on FPGA," *IEEE Trans. Ind. Informat.*, Vol. 10, No. 4, pp. 2166-2179, Nov. 2014.
[21] N. Krihely and S. Ben-Yaakov, "Modeling and evaluation of diode reverse recovery in discrete-transition simulators," in *IEEE Energy Conversion Congress and Exposition (ECCE)*, pp. 4514-4520, Sep. 2010.
[22] H. F. Blanchette, T. Ould-Bachir, and J. P. David, "A state-space modeling approach for the FPGA-based real-time simulation of high switching frequency power converters," *IEEE Trans. Ind. Electron.*, Vol. 59, No. 12, pp. 4555-4567, Dec. 2012.
[23] M. R. Larijani, M.-R. Zolghadri, and M. Shahbazi, "Design and implementation of an FPGA-based real-time simulator for H-bridge converter," in *7th Power Electronics and Drive Systems Technologies Conference (PEDSTC)*, pp. 504-510, Feb. 2016.
[24] C. Batard, F. Poitiers, C. Millet, and N. Ginot, "Simulation of power converters using matlab-simulink," in *MATLAB - A Fundamental Tool for Scientific Computing and Engineering Applications*, Vol. 1, pp. 43–68, Sep. 2012.
[25] T. Kokenyesi and I. Varjasi, "Comparison of real-time simulation methods for power electronic applications," in *4th International Youth conference on Energy (IYCE)*, pp. 1-5, Jun. 2013,
[26] R. Fedkiw, *Ordinary Differential Equations*, Society for Industrial and Applied Mathematics (SIAM), Jul. 2017.
[27] R. J. LeVeque, *Finite Difference Methods for Ordinary and Partial Differential Equations*, Society for Industrial and Applied Mathematics (SIAM), 2007.
[28] Ó. Jiménez, Ó. Lucía, I. Urriza, L. A. Barragan, D. Navarro, and V. Dinavahi, "Implementation of an FPGA-based online hardware-in-the-loop emulator using high-level synthesis tools for resonant power converters applied to induction heating appliances," *IEEE Trans. Ind. Electron.*, Vol. 62, No. 4, pp. 2206-2214, Apr. 2015.
[29] 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.
[30] A. Visioli, *Practical PID control*, 1st ed., Springer-Verlag, London, pp. 42-43, 2006.

András Futó was born in Budapest, Hungary. He received his B.S. degree in Electrical Engineering at the Florida Institute of Technology, Melbourne, FL, USA, in 2010; and his M.S. degree at the Budapest University of Technology and Economics (BME), Budapest, Hungary, in 2012, where he is presently working towards his Ph.D. degree. In 2015, he worked on a HIL simulation system in the Hyundai Technology Center Hungary Kft., Budapest, Hungary. In 2016 and 2017, he worked on the battery charging system of an electrical aircraft at Siemens - Hungary, Budapest, Hungary. His current research interest include HIL, the control of power converters, and decreasing the harmonic emissions of power converters.

Tamás Kökényesi was born in Gyöngyös, Hungary, in 1986. He received his B.S. and M.S. degrees in Electrical Engineering from the Budapest University of Technology and Economics (BME), Budapest, Hungary, in 2009 and 2011, respectively. He is presently working as an Assistant Lecturer in the Department of Automation and Applied Informatics, BME. Since 2005, he has been developing HIL simulators at Siemens Zrt., Budapest, Hungary. His current research interests include the FPGA based HIL simulation of power electronics and drive systems.

István Varjasi was born in Soltvadkert, Hungary, in 1956. He received his M.S. and Ph.D. degrees in Electrical Engineering from the Budapest University of Technology and Economics (BME), Budapest, Hungary, in 1980 and 1997, respectively. He is presently working as an Associate Professor in the Department of Automation and Applied Informatics at BME. His current research interests include the control of power electronics and electrical drives, and the real-time simulation of power electronics. He has more than 100 publications.

Zoltán Sütő was born in Pécs, Hungary, in 1969. He received his M.S. and Ph.D. degrees in Electrical Engineering from the Budapest University of Technology and Economics (BME), Budapest, Hungary, in 1993 and 2008, respectively. He is presently working as an Associate Professor in the Department of Automation and Applied Informatics at BME. His current research interests include nonlinear dynamics, chaos and bifurcations in power electronics, model-based design, and rapid prototyping in the control of power electronics. He has more than 50 publications.

István Vajk received his M.S. degree in Electrical Engineering from the Budapest University of Technology and Economics (BME), Budapest, Hungary, in 1975, where he was a Postgraduate Student and obtained his Ph.D. degree. Since 1976, he has been with the Faculty of Electrical Engineering and Informatics in the Department of Automation and Applied Informatics. He obtained his Doctor of Sciences degree from the Hungarian Academy of Sciences (MTA), Budapest, Hungary, in 2007. From 1994 to 2016, he was the Head of the Department. His current research interests include the theory and application of control systems, especially adaptive systems and system identification, as well as real-time software engineering.

Attila Balogh was born in Pápa, Hungary, in 1982. He received his M.S. and Ph.D. degrees in Electrical Engineering from the Budapest University of Technology and Economics (BME), Budapest, Hungary, in 2006 and 2011, respectively. He is presently working as an Associate Professor in the Department of Automation and Applied Informatics at BME. His current research interests include power control, power circuit design, converter optimization techniques, power quality improvement methods for sea vessels, and rapid prototyping in the control of power electronics. He has more than 20 publications.

Gergely György Balázs received his M.S. degree in Electrical Engineering at the Budapest University of Technology and Economics (BME), Budapest, Hungary, in 2007; and his Engineering-Economist degree at Corvinus University, Budapest, Hungary, in 2012. He obtained his Ph.D. degree at BME, in 2013. Since 2010, he has been a Lecturer in the Department of Electric Power Engineering at BME, where he presently holds the rank of Senior Lecturer. Since 2014, he has been the Head of the R&D department at Siemens Zrt., Budapest, Hungary. His current research interests include battery management systems and electric dive system controls for special applications.