Half-Bridge Voltage Source Inverter Control Development Using HIL System

Andrej Brandis
Department of Electromechanical Engineering
Faculty of Electricity Engineering, Computer Science and Information Technology Osijek
Osijek, Croatia
abrandis@ferit.hr

Denis Pelin
Department of Electromechanical Engineering
Faculty of Electricity Engineering, Computer Science and Information Technology Osijek
Osijek, Croatia
dpelin@ferit.hr

Danijel Topić
Department of Power Engineering
Faculty of Electricity Engineering, Computer Science and Information Technology Osijek
Osijek, Croatia
dtopic@ferit.hr

Goran Knežević
Department of Power Engineering
Faculty of Electricity Engineering, Computer Science and Information Technology Osijek
Osijek, Croatia
gknezevic@ferit.hr

***Abstract*— The paper deals with a HIL (Hardware In the Loop) system or so-called Rapid Control Prototyping (RCP) tools, used in a single-phase half-bridge Voltage Source Inverter (VSI) control development. The control algorithm is developed for the ATMEL ATmega2560 microcontroller. The code is implemented to the microcontroller via Arduino development board. A control board mock-up is assembled for control algorithm validation. HIL system is used as an RCP tool for testing. Furthermore, the VSI model is done in a HIL Schematic Editor and HIL SCADA software environment. The system functionality is tested for two selected operation points. Waveforms are recorded, and the voltage spectrum analysis is performed via external low-voltage spectrum analyzer. The shortcomings of developed system are addressed and the solution for them is proposed.**

Keywords*— *hardware in the loop, rapid control prototyping, voltage source inverter, control algorithm, development board

I. Introduction

Due to nowadays trending in the electrical energy power production and conversion, the Power Electronic (PE) converters became an indispensable part of every such industry [1]. A great share of PE converters consist of VSIs which are used mainly in renewables and industries with AC motor drive systems, including the automotive [2], [3]. Due to new and ongoing applications, and the demand for constant improvements (better efficiency, increasing compactness, etc.), the VSIs control algorithms need continual development [4]. Despite the possibility of simulated control algorithm development, sooner or later the control algorithm needs to be used on a real system. In order to avoid a potential VSI hardware damage due to control algorithm flaws, the algorithm can be safely developed and tested using HIL system [5]. Such system enables the real hardware control circuits (Microcontrollers, DSPs, FPGAs, etc.) to be connected to a virtual VSI modeled in software. In this way, the control algorithm can easily be tested and debugged, and finally translated to a real control hardware. The method of such control developing is known as a Rapid Control Prototyping or RCP for short [6], and in recent years this term has become synonymous for fast and safe PE converters control development, as shown e.g. in [7]–[11]. This method is also used in this paper and will be explained later.

The common VSI topology is a three-phase VSI in bridge configuration as shown in Fig. 1.

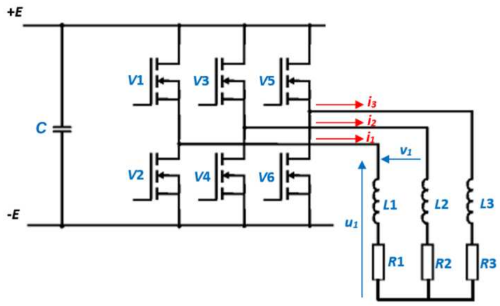

Fig. 1. Schematic of three-phase VSI in bridge configuration

This topology (Fig. 1) contains of six switching components (usually MOSFETs or IGBTs). Regarding switching technology, it is worth to mention that in recent years among researchers, the main subjects are the GaN and SiC switching technologies [12], [13]. Nevertheless, a standard three-phase VSI can be assembled in three different ways: by using discrete components as presented in [14], by using three single leg modules as shown in [15], or by using a full bridge module with nine embedded switches as presented in [16].

This paper deals with a single IGBT leg module option since the aim of this research is to develop a control algorithm for the real assembled 1 kW system using an RCP development method. This converter is designed as a single phase VSI in a half-bridge configuration (Fig. 2).

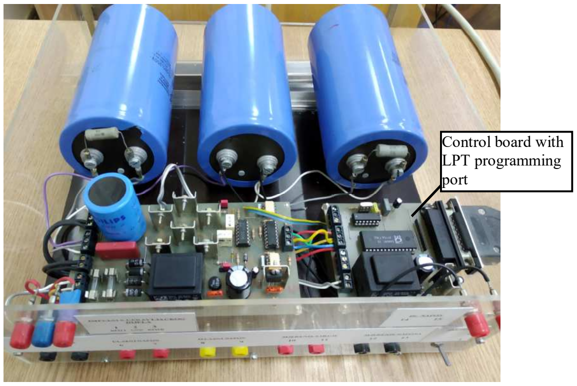
Control board with LPT programming port

Fig. 2. Single-phase 1 kW half-bridge VSI

978-1-6654-0187-6/21/$31.00 ©2021

The presented VSI's control is based on a bipolar PWM [17]. The choice of modulation method is limited due to a half-bridge topology configuration which specifically requires two DC link capacitors to create a zero-voltage node (Fig. 5). A control board mock-up with a development board and the belonging periphery is assembled and prepared for converter (Fig. 3b).

The converter given in Fig. 2 is also modeled in a Typhoon HIL software (Fig. 5), which is used later for control algorithm testing. The VSI from Fig. 2 is driven by the SA8281 chip and it is programmed by an obsolete system using an LPT port (Fig. 2) and DOS system. The final goal is to upgrade this VSI with a new control board system. Hence, the control board operation needs evaluation before the installation. This paper deals with the first step of the VSI (Fig. 2) upgrading process, i.e., the control algorithm development.

The rest of the paper is organized in four chapters. The second chapter deals with the control algorithm and the control board mock-up. The third chapter presents the VSI software and hardware modeling tools. The second to last chapter presents the measurement results for two different operation points. At the final chapter, the conclusion is made summarizing the most important points of presented paper.

II. Half-Bridge VSI Control

The main motive of replacing the old SA8281 control (Fig. 2) with the up-to-date microcontroller is done for practical reasons – to reduce size and complexity of the whole VSI system.

A. Hardware

Since the VSI from Fig. 2 needs +5 V digital signal port and since the peripheral communication with the user is mandatory, a suitable development board type is chosen. The main parts of the control system are development board, display and keypad (Fig. 3). The Arduino board with ATmega2560 processor is utilized, primarily due to 8 kB of SRAM memory. The display is chosen to be the ST7920 type, due to the possibility of displaying a minimum of six lines simultaneously. The keypad is a standard 4x4 matrix format keypad which allows user a prompt input of desired parameters. The connection diagram of the control system is shown in Fig. 3a. The assembled control board mock-up is given in Fig. 3b.

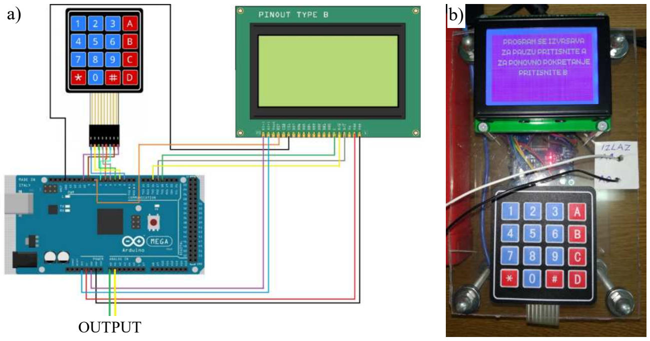

Fig. 3. The connection diagram (a) and assembled control board mock-up (b) for VSI control

B. Software

The control algorithm is written in Arduino software environment, which flow diagram is shown in Fig. 4.

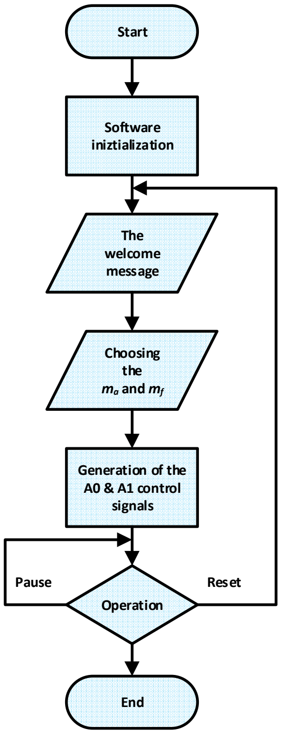

Fig. 4. Control algorithm flow diagram

The control algorithm begins with the periphery initialization. The first initialized periphery is the ST7920 type display with 128x64 resolution, for which a standard library is used for programming. Second initialized is the keypad, where the key matrix and the pinouts are defined for the communication with the development board. Inside of the setup function, the functions needed for the operation are initialized and activated. Since the VSI control demands synchronous switching of both transistors, the control output is set via PORTF register function which comprising the A0 – A7 pins, out of which only A0 and A1 are utilized for the IGBTs control purpose.

After periphery initialization, the functions for user input are defined. Next, the welcome message is displayed, afterward the user chooses the amplitude modulation index m_a parameter. Then, the frequency modulation index m_f parameter is chosen. After pressing the confirmation key on the keypad, the program runs. The display shows chosen parameters and additional options such as resetting, stopping or pausing the process (Fig. 4).

The transistors control logic is derived via the two Look Up Tables (LUTs); one for the sine wave and one for the triangle wave, both fragmented into a 1000 equal pieces (ranges 0-1000). Every element of the sine LUT is compared with the chosen m_f-th value of the triangle LUT. With this method, a triangle waveform signal which is repeated m_f times is obtained.

If the value of the sine LUT is greater than triangle LUT, the output gives the result field of 1, otherwise it gives a 0. The result field is then repeated in the loop until interrupted by the user. The full code is available as supplementary material in back matter of the paper.

III. Voltage Source Inverter Model

Although HIL systems can be used for many purposes, in the strict sense related to this paper, it can be described as a testing environment for developing and debugging the VSI's control algorithm. The HIL software simulates the actual external VSI hardware, receiving the input control signals through the HIL interface and processes them in the same way as they would be processed within the actual VSI hardware.

A. *Software*

The VSI schematic model (Fig. 5) is created in a HIL Schematic Editor. Since this model simulates real system for which the control is developed, the components parameters are chosen according to the real VSI from Fig. 2.

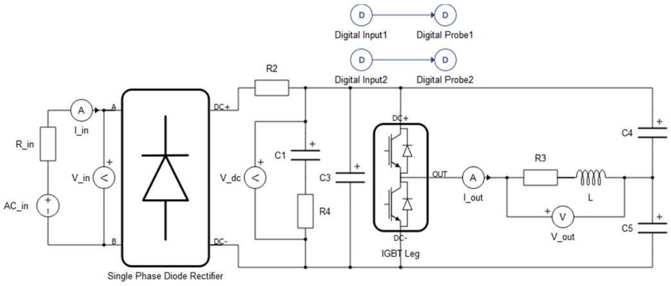

Fig. 5. VSI HIL Schematic Editor model

The external signals A0 and A1 sent from the controller board mock-up are linked to the simulation model through the HIL hardware and are set up in the software (*Digital Input1* and *Digital Input2*; Fig. 5). The usual process of HIL simulation model development is given in Fig. 6.

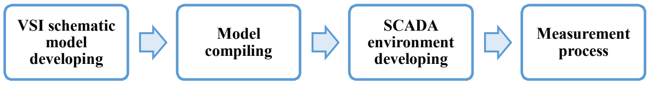

Fig. 6. HIL modeling process

After the VSI simulation model is created, a debugging process is performed, until compiling is successfully done. In this stage this means that the HIL software translated the created schematic into a machine language successfully, and that the model is implemented into the HIL's FPGA processor – ready for use.

After previous step, the provided HIL SCADA software is utilized. This piece of software serves as a virtual SCADA environment for monitoring all HIL's inputs and outputs, including both – control and power part of the VSI. It is worth to mention that the input and output hardware pins can be assigned to the created schematics in SCADA environment (this can be done in a HIL Schematic Editor also). Furthermore, simulation start and stop functions can be performed in the SCADA. After the SCADA is created, the user can start with the measurements. It is worth to mention that HIL software can work on-line (with the hardware) and off-line (without the hardware).

B. *Hardware*

The full hardware setup used in paper can be seen in Fig. 7. A brief explanation follows. The control board mock-up (mark 1) is connected via USB cable to the PC (mark 5). This cable serves as a programming cable for the microcontroller, as well as the power supply for the mock-up. The Typhoon HIL hardware is marked with 2. This hardware has its own power supply and the USB cable which is connected to the PC for real-time simulation via provided software. On the front of the HIL device a HIL DSP interface is attached (mark 3). This interface has a slot for TI C2000 DSP type MCU (for DSP control programming and testing), and the pinouts for analog inputs and outputs (16 analog inputs + 16 analog outputs and 32 digital inputs and 32 digital outputs). A low-voltage spectrum analyzer software is marked with 4. This is a *UniTrain SO4204* interface which has dedicated software for low voltage (0 – 40 V) spectrum analysis. This interface is connected also to the PC via USB cable.

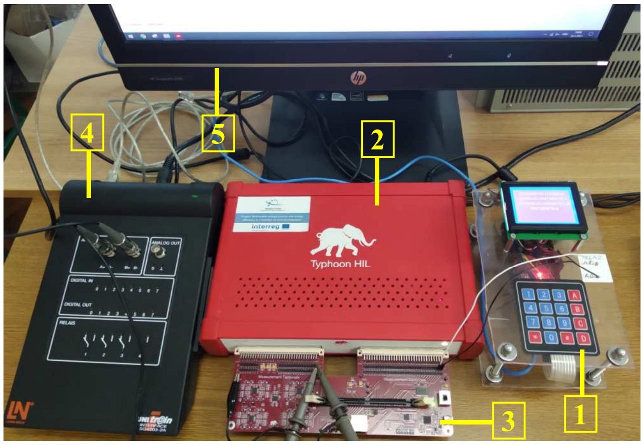

Fig. 7. Rapid control prototyping workplace; 1 – control board mock-up, 2 – HIL hardware, 3 – HIL DSP interface, 4 –spectrum analyzer interface, 5 – PC with required software

After everything is connected and checked, a measurement process follows.

IV. Measurement Results

Measurements are done for many different operations points out of which four are presented in this chapter. Selected parameters are shown in Tab. 1.

TABLE I. Selected Measurement Parameters

Set	m_a	m_f	E [V]	f [Hz]	R [Ω]	L [mH]
1.	1	12	230	50	800	330
2.	1	48				
3.	1	24	230	50	800	330
4.	0,8	24				

The idea here is to show how control works with different amplitude modulation indexes m_a, as well as with different frequency modulation indexes m_f, to detect possible flaws in control algorithm. The measurement results for set 1 – set 4 (Tab. 1) are shown in Fig. 8 – Fig. 11.

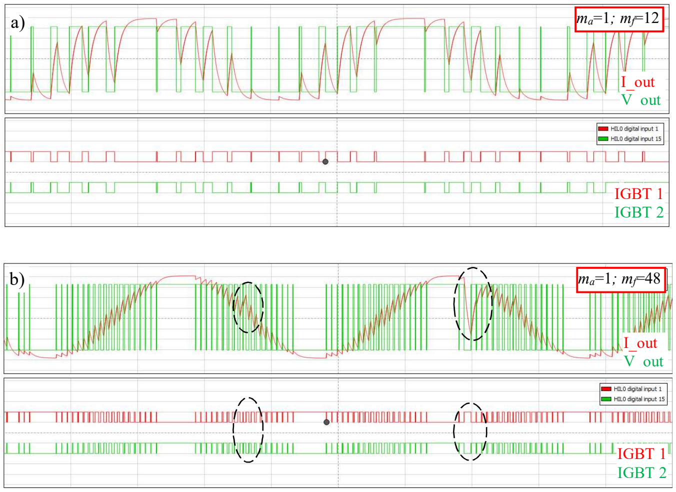

Fig. 8. The output voltage and current waveforms for the parameters from set 1 (a) and set 2 (b)

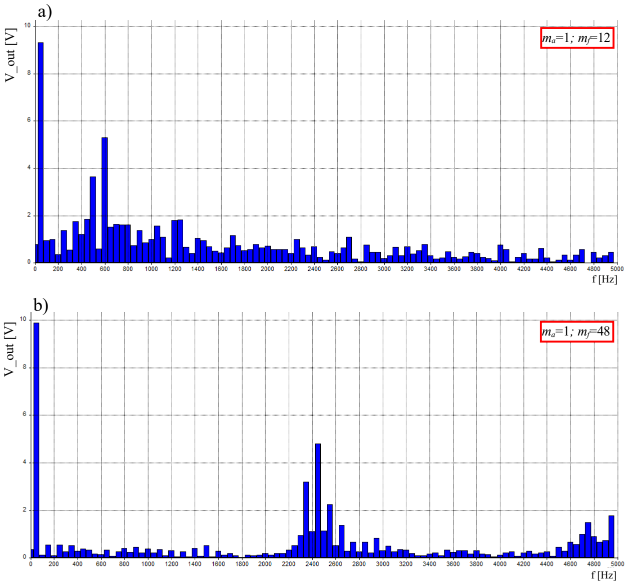

Fig. 9. The spectrum analysis of the output voltage for the parameters from set 1 (a) and set 2 (b)

It can be seen from waveforms on Fig. 8 that higher the frequency modulation index m_f is, the current is more sinusoidal which implies lower THD. The waveforms also show some anomalies as marked with the dashed line on Fig. 8b. Lower switching frequency with the m_f=12 (Fig. 8a) is not so sensitive to this problem as the higher switching frequency at m_f=48 (Fig. 8b). At this point, the algorithm clearly needs some improvements. With more detailed analysis of presented irregularities, we concluded that one of the problems is in the main loop code, and more specifically, this problem occurs due to constant keypad operation checking. Since the whole code is in the mail loop, the algorithm checks if anything is pressed on keypad after every forwarded control signal. This operation takes up a certain amount of the CPU time and extends affected IGBT control signal, which then causes the anomaly in the voltage and at the end in the current waveform.

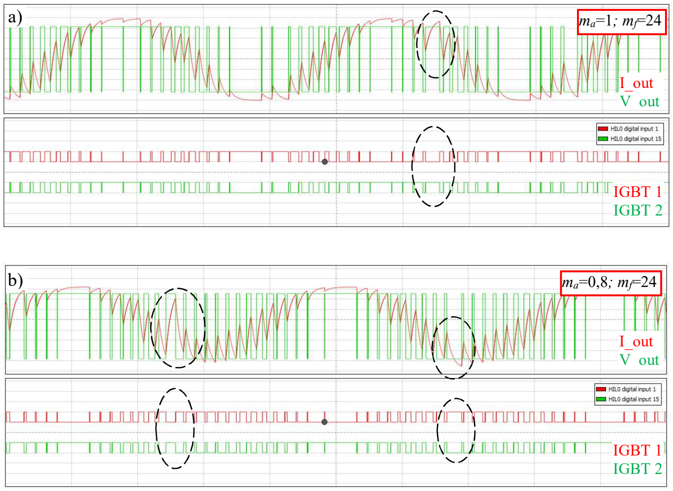

Fig. 10. The output voltage and current waveforms for the parameters from set 3 (a) and set 4 (b)

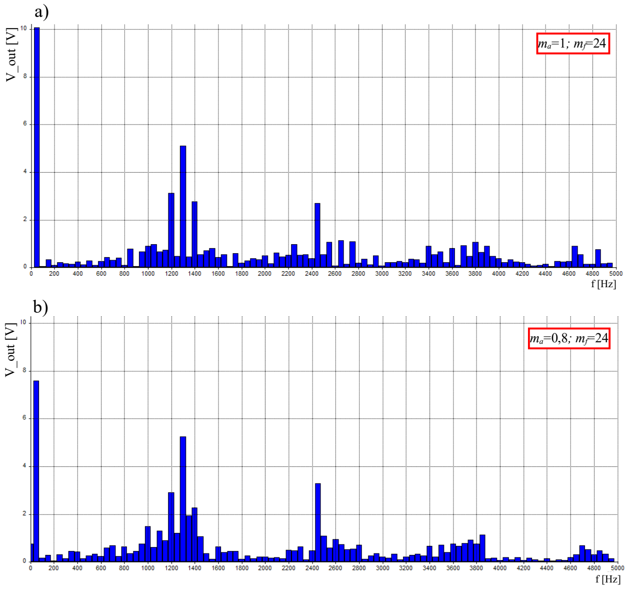

Fig. 11. The spectrum analysis of the output voltage for the parameters from set 3 (a) and set 4 (b)

In the case of set 3 and set 4 measurements, the frequency modulation index m_f is constant, and the amplitude modulation index m_a is changed. Changing the m_a (with keeping the m_f constant) yields with different current amplitudes i.e., lowering the m_a would give lower current amplitude and vice versa This is expected. The measurement results from Fig. 10 shows similar problems as the waveforms from Fig. 8. The conclusion is that at higher switching frequencies this problem is more pronounced due to shorter switching period, appearing as a longer duration of affected signal in relation to the switching period.

The spectral analysis for every set of measurements (Tab. 1) is given in Fig. 9 and Fig. 11. As expected, the higher harmonics appears at the switching frequencies and its multiplies, which is in direct correlation with the frequency modulation index m_f (taking into consideration fixed output frequency of 50 Hz, increasing the m_f would increase the switching frequency and vice versa).

It is visible from conducted spectral analysis that the spectrum is not clear (as ideally should be) in between the fundamental harmonic and in between the m_f-th harmonic regions. Referring to the voltage waveforms from Fig. 8 and Fig. 10 this can be expected. Namely, due to presented shortcomings in switching algorithm, the asymmetry of the output voltage waveform is produced. This produces all kinds of higher harmonics, including an even-order harmonics as seen in Fig. 8 and Fig. 10. Although the inverter technically works, all the problems from above impairing the modulation quality, hence, the control algorithm cannot be implemented on the VSI (Fig. 2) yet, until this is solved.

A. Proposed solution for addressed problems

Fig. 12 shows the output current and voltage waveforms for the arbitrary parameters set, with (Fig. 12a) and without (Fig. 12b) keypad attached to the control board mock-up (Fig. 3b). This clearly shows the influence of the keypad on control algorithm. The addressed problem therefore can be solved by changing the algorithm fundamentally, i.e., to use a completely different programming logic regarding time commands.

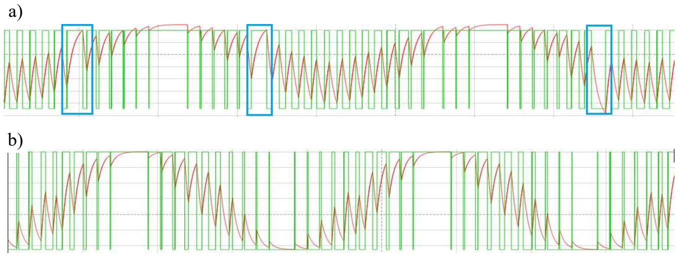

Fig. 12. The voltage and current waveforms with (a) and without (b) keypad attached

Presented control algorithm uses a *delayMicroseconds* function. This function as every delay function in Arduino takes up a processing time of the microcontroller. Due to unpredictable appearing of irregular control signals for IGBTs, the proposal is to use *Interrupt* functions instead of *delayMicroseconds*. *Interrupt* functions do not take up processing time, rather, they execute certain commands when are called. Hence, when processor execute some algorithm and if the *Interrupt* is used, the processor switches to the *Interrupt* commands and executes them, and when finished it continues with the algorithm. This logic is essentially different than usual Arduino *loop* kind of programming.

Another problem found in testing is a 160 V DC voltage appearance at the output (load) voltage when a pause or stop command is used on the keypad. This appearance could be potentially dangerous in the real converter and this phenomenon needs also to be fixed before the implementation to the real converter.

Since this paper is not aimed on the modulation quality but on the control algorithm shortcomings detection via RCP developing methods, the solution to the addressed problems is not carried at this point. Rather, the solution is given as a proposal for the future upgrade.

V. Conclusion

The paper presents the RCP method in developing of control algorithm of single-phase VSI. A special attention is given to the software control debugging using the RCP method. The measurement results for chosen operation points show that developed ATmega2560 control works on the HIL model, but not perfect. Thus, the algorithm needs some clearly stated improvements before it can be implemented on real system. To overcome addressed problems, a different approach in algorithm logic is proposed (using *Interrupt* instead of *delayMicroseconds* functions in Arduino programming). In the future work, all remaining flaws of presented control algorithm will be corrected, and when ready, the control algorithm will be translated to the real 1 kW single-phase VSI.

The advantages of using the RCP method in control algorithm developing using HIL systems are obvious as presented in the paper; debugging and upgrading the control algorithm using an RCP and HIL is safe, with no need for the real VSI hardware. This allows the hardware and software of any converter to be developed independently, which can save many resources. While using only simulation does not give a real specific microcontroller behavior, applying the control algorithm directly to the converter increases the probability of damaging something in the systems circuit. Therefore, the HIL system can be described as the bridge between the two extremes in PE converters control developing. The only real drawback of such systems is a relatively high price (starting from 20.000,00 Eur and above).

Supplementary Materials

The code used in this paper is available in publicly accessible repository at: https://doi.org/10.6084/m9.figshare.14701218.

Acknowledgement

This work was funded by the European Union through the European Regional Development Fund Operational Programme Competitiveness and Cohesion 2014 – 2020 of the Republic of Croatia under project KK.01.1.1.04.0034 „Connected Stationary Battery Energy Storage”

References

[1] M.-K. Nguyen, “Power Converters in Power Electronics: Current Research Trends,” *Electronics*, vol. 9, no. 4, p. 654, Apr. 2020, doi: 10.3390/electronics9040654.

[2] L. Meegahapola, P. Mancarella, D. Flynn, and R. Moreno, “Power system stability in the transition to a low carbon grid: A techno-economic perspective on challenges and opportunities,” *WILEY PERIODICALS, INC*, Apr. 2021, doi: 10.1002/wene.399.

[3] S. R. Khasim and D. C., “Selection parameters and synthesis of multi-input converters for electric vehicles: An overview,” *PERGAMON-*

ELSEVIER SCIENCE LTD, vol. 141, May 2021, doi: 10.1016/j.rser.2021.110804.

[4] S. Xu, Y. Xue, and L. Chang, “Review of Power System Support Functions for Inverter-Based Distributed Energy Resources- Standards, Control Algorithms, and Trends,” *IEEE Open J. Power Electron.*, vol. 2, pp. 88–105, 2021, doi: 10.1109/OJPEL.2021.3056627.

[5] “Typhoon Hardware In the Loop,” May 24, 2021. https://www.typhoon-hil.com/hil-hardware/ (accessed May 24, 2021).

[6] J. Martins *et al.*, “Test Platform for Rapid Prototyping of Digital Control for Power Electronic Converters,” in *IECON 2019 - 45th Annual Conference of the IEEE Industrial Electronics Society*, Lisbon, Portugal, Oct. 2019, pp. 2056–2061. doi: 10.1109/IECON.2019.8927836.

[7] B. Chandan, P. Dwivedi, and S. Bose, “Closed Loop Control of SEPIC DC-DC Converter Using Loop Shaping Control Technique,” in *2019 IEEE 10th Control and System Graduate Research Colloquium (ICSGRC)*, Shah Alam, Malaysia, Aug. 2019, pp. 20–25. doi: 10.1109/ICSGRC.2019.8837093.

[8] P. N. Singh, S. Hajari, O. Ray, and S. R. Samantaray, “Development of Controller HIL Test-bed for Solar-battery integration,” in *2021 1st International Conference on Power Electronics and Energy (ICPEE)*, Bhubaneswar, India, Jan. 2021, pp. 1–6. doi: 10.1109/ICPEE50452.2021.9358471.

[9] R. Collin, M. Stephens, and A. von Jouanne, “Development of SiC-Based Motor Drive Using Typhoon HIL 402 as System-Level Controller,” in *2020 IEEE Energy Conversion Congress and Exposition (ECCE)*, Detroit, MI, USA, Oct. 2020, pp. 2689–2695. doi: 10.1109/ECCE44975.2020.9236201.

[10] K. Nisha and D. N. Gaonkar, “Predictive Control of Three Level Bidirectional Converter in Bipolar DC Microgrid for EV Charging Stations,” in *2020 IEEE International Conference on Power Electronics, Smart Grid and Renewable Energy (PESGRE2020)*, Cochin, India, Jan. 2020, pp. 1–6. doi: 10.1109/PESGRE45664.2020.9070356.

[11] C. D. Kumar, B. S. Shiva, and V. Verma, “Vector Control of PMSM Drive with Single Current Sensor,” in *2020 IEEE Students Conference on Engineering & Systems (SCES)*, Prayagraj, India, Jul. 2020, pp. 1–6. doi: 10.1109/SCES50439.2020.9236774.

[12] R. Letor, F. Scrimizzi, G. Longo, F. Iucolano, and M. Moschetti, “Compact design of DC/DC converter with new STi ^{2} GaN solution,” in *2020 AEIT International Conference of Electrical and Electronic Technologies for Automotive (AEIT AUTOMOTIVE)*, Turin, Italy, Nov. 2020, pp. 1–5. doi: 10.23919/AEITAUTOMOTIVE50086.2020.9307412.

[13] H. Li, Y. Ma, R. Ren, and F. Wang, “SiC Impact On Grid Power Electronics Converters,” in *2020 IEEE Power & Energy Society General Meeting (PESGM)*, Montreal, QC, Canada, Aug. 2020, pp. 1–5. doi: 10.1109/PESGM41954.2020.9281404.

[14] S. Selvaperumal, M. S. S. Sundari, and K. Siva Subramanian, “Hardware Implementation of Asymmetrical Cascaded H-Bridge 41 Level Inverter using PIC Microcontroller,” in *2018 International Conference on Smart Systems and Inventive Technology (ICSSIT)*, Tirunelveli, India, Dec. 2018, pp. 415–418. doi: 10.1109/ICSSIT.2018.8748839.

[15] Y. Xiong, A. Oyane, T. Ou, S. Thilak, J. Imaoka, and M. Yamamoto, “Comparison of switching performance between GaN and SiC MOSFET via 13. 56MHz Half-bridge Inverter,” in *2020 IEEE 29th International Symposium on Industrial Electronics (ISIE)*, Delft, Netherlands, Jun. 2020, pp. 672–676. doi: 10.1109/ISIE45063.2020.9152253.

[16] M. Calabretta, A. Sitta, S. M. Oliveri, and G. Sequenzia, “Silicon carbide multi-chip power module for traction inverter applications: thermal characterization and modeling,” *IEEE Access*, pp. 1–1, 2021, doi: 10.1109/ACCESS.2021.3080505.

[17] V. Trifa, Gh. Brezeanu, and E. Ceuca, “Unipolar PWM vs Bipolar PWM in Three Phase Block Commutation,” in *2020 International Semiconductor Conference (CAS)*, Sinaia, Romania, Oct. 2020, pp. 169–172. doi: 10.1109/CAS50358.2020.9267979.