A Direct Modulation for Matrix Converters Based on the One-Cycle Atomic Operation Developed in Verilog HDL

Pawel Szczepankowski, *Member, IEEE*, Wojciech Sleszynski, *Member, IEEE*, and Tomasz Bajdecki

***Abstract*—This article presents a fast direct pulsewidth modulation (PWM) algorithm for the conventional matrix converters developed in Verilog hardware description language. All PWM duty cycle calculations are performed in one cycle by an atomic operation designed as a digital module using field-programmable gate array basic blocks. The algorithm can be extended to any number of output phases. The improved version of the discontinuous direct analytic voltage PWM (DAV-PWM) method is proposed, in which the use of trigonometry, angles, and program loops has been eliminated. The proposed DAV-PWM is equivalent to the space vector modulation; it can be applied during input asymmetry and also allows for the control of the displacement input angle. The proposal has been verified using the circuit simulation in PSIM, digital structure modeling in ModelSim, and finally through an experiment.**

***Index Terms*—AC–AC converters, field-programmable gate array (FPGA) device, matrix converters, pulsewidth modulation (PWM).**

NOMENCLATURE

T	Transposition of the matrix.
$\boldsymbol{v}_{\mathrm{i}}$	Measured input voltages $[v_{\mathrm{i}1}, v_{\mathrm{i}2}, v_{\mathrm{i}3}]^T$.
$\boldsymbol{v}_{\mathrm{o}}$	Averaged output voltages $[v_{\mathrm{o}1}, v_{\mathrm{o}2}, v_{\mathrm{o}3}]^T$.
$\boldsymbol{i}_{\mathrm{i}}$	Averaged input currents $[i_{\mathrm{i}1}, i_{\mathrm{i}2}, i_{\mathrm{i}3}]^T$.
$\boldsymbol{i}_{\mathrm{o}}$	Measured output currents $[i_{\mathrm{o}1}, i_{\mathrm{o}2}, i_{\mathrm{o}3}]^T$.
$\boldsymbol{D}$	PWM duty cycle matrix with size 3×3.
x	Real signal component.
y	Imaginary signal component.
$\boldsymbol{v}_{\mathrm{ix}}$	Matrix of $\mathbf{v}_{\mathrm{i}}$ in-phase components.
$\boldsymbol{v}_{iy}$	Matrix of $\mathbf{v}_i$ quadrature components.
$\boldsymbol{v}_{ox}$	Real parts of reference voltages $[v_{o1x}, v_{o2x}, v_{o3x}]^T$.
$\boldsymbol{v}_{oy}$	Imaginary parts of reference voltages $[v_{o1y}, v_{o2y}, v_{o3y}]^T$.
ϕ_i	Input displacement angle.
s_i, s_o	Input and output voltage sectors.
ω_i	$= 2\pi f_i$, where f_i is the input frequency.
ω_o	$= 2\pi f_o$, where f_o is the output frequency.
q	$= V_o/V_i$, Voltage transfer ratio.
$q_{\max}$	$= \sqrt{3}/2$, Maximum value of q.
T_{PWM}	Modulation period.
f_{s}	$= 1/T_{\mathrm{PWM}}$ sampling frequency.

Manuscript received August 5, 2020; revised December 23, 2020 and February 25, 2021; accepted April 19, 2021. Date of publication May 5, 2021; date of current version December 20, 2021. This work was supported by the Linte laboratory. *(Corresponding author: Pawel Szczepankowski.)*

Pawel Szczepankowski and Wojciech Sleszynski are with the Gdańsk University of Technology, 80-233 Gdańsk, Poland (e-mail: pawel.szczepankowski@pg.edu.pl; wojciech.sleszynski@pg.edu.pl).

Tomasz Bajdecki is with the Institute of Power Engineering Gdańsk, 80-870 Gdańsk, Poland (e-mail: t.bajdecki@ien.gda.pl).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TIE.2021.3076703.

Digital Object Identifier 10.1109/TIE.2021.3076703

I. INTRODUCTION

A CONVENTIONAL matrix converter (CMC), shown in Fig. 1, contains semiconductor switches arranged into a matrix configuration divided into three cells: $\{h_{11}, h_{21}, h_{31}\}$, $\{h_{12}, h_{22}, h_{32}\}$, and $\{h_{13}, h_{23}, h_{33}\}$. Compared to ac–dc–ac back-to-back converters with large capacitors in the dc link, the CMC allows for direct ac–ac conversion using the small input filter, which is an advantage of these solutions [1]–[3]. The general motor drive application scheme with the matrix converter is illustrated in Fig. 1. Such a topology permits for regenerative power from the electrical motor M with negligible input grid current harmonic content. An important feature of the CMC control is the possibility to adjust the input displacement angle to zero [4]–[6]. The single switch h can be built from two transistors with two diodes or two reverse-blocking insulated-gate bipolar transistor (RB-IGBT) devices [1], [7].

The single CMC's cell is properly controlled to prevent line-to-line short circuits and to maintain a continuous waveform of the load current. Due to the safe commutation process requirement, the dead-time mechanism should be applied during the generation of switch control signals. The overvoltage upon the switch, caused by an interruption of the inductive current, has to be absorbed by a clamp circuit [8], [9]. The reduction of the scale of both the input filters and the clamp circuit can be reached by applying faster semiconductors with a high operation frequency; thus, a panel size of the CMC can be significantly reduced [10]. Attempts to integrate this panel with the electric motor have already taken place [11]. Gallium nitride (GaN) and silicon carbide (SiC) semiconductors offer fundamental advantages over silicon solutions [12]–[15]. The switch's operation frequency can be very high compared to the silicon counterparts, which makes these devices great for high-frequency applications, which also include high-speed drives in compressors or high-speed generators in gas turbines [16]–[18]. Such applications require complex calculations within a short period of time.

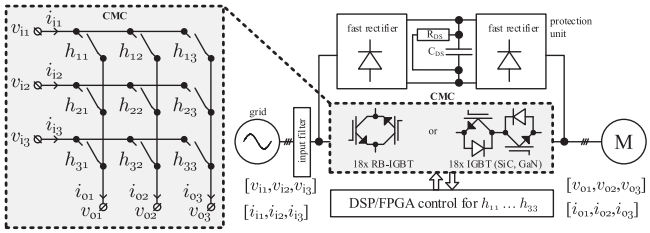

Fig. 1. Classic matrix converter and its typical application scheme.

This article presents an approach to computations performed in a field-programmable gate array (FPGA) in one step, during one cycle, without trigonometry and angles. Thus, the execution time of the PWM duty cycle calculation is atomic and limited only by the critical timing constraints of the FPGA device. The development of a digital structure that performs such an algorithm, in one clock cycle, requires the use of an appropriate modulation method that uses only simple arithmetic operations supported by Verilog HDL. Moreover, the proposed solution should allow us to obtain the maximum voltage transfer ratio and to adjust the angle at the input of the system, which is a characteristic feature of the PWM for the CMC [1].

PWM strategies for the CMC have been widely reported in the literature, such as the direct control by the Venturini approach [3], [19], [20], scalar control realized according to the Roy method [21], and the space vector modulation (SVM) [1], [22]. Other control methods, such as hysteresis and direct torque control, are interesting alternatives [23], [24]. However, the torque ripple in the low-speed region or switching frequency variations according to the change of the motor speed are drawbacks of these approaches. The aforementioned methods are not suitable for developing the simplest solution, which should be built from basic digital elements such as multiplexes, adders, and multipliers. A simplified carrier-based modulator based on the concept of a virtual matrix converter was presented in [25]. The proposed algorithm allows for obtaining the same instantaneous matrix states as the SVM method with a smaller number of calculations. Although the solution represents a more synthetic and systematic approach, this original approach has a major disadvantage. An input displacement angle is permanently equal to zero and cannot be adjusted.

The concept of simplifying and generalizing the PWM algorithm is also presented in [26] and [27]. As with the previous method, the proposal makes it possible to calculate PWM duty cycles without trigonometry with minimal computational effort. Moreover, the proposed computation scheme allows for the application of various methods of power factor control. Among the described modulation methods, the authors indicated that the direct analytic voltage pulsewidth modulation (DAV-PWM) is optimal because it allows reaching the maximum voltage transfer ratio equal to 0.866 with the same switch state collection as the SVM modulation. This algorithm has been developed as a sequential code, which contains a program loop for preselection input voltage vectors. Such a program loop has to be eliminated within the atomic and concurrent implementation based on an FPGA device. This article mainly addresses this issue and proposes certain improvements, which are clarified in Section II. The theoretical basis of an improved DAV-PWM algorithm, ready for the hardware description language (HDL) conversion, has been presented in the next section. Power electronics simulation, modeling based on the HDL, and hardware-in-the-loop (HIL) verification have been presented in Section IV. The conducted research path is summarized in Table I. Experimental results are presented in Section V. The Verilog HDL code of the improved DAV-PWM algorithm is available in an Appendix after the conclusion section.

TABLE I
SUMMARY OF THE CONDUCTED RESEARCH PATH

formal analysis	Conducted using Matlab R2020a software, containing symbolic variables, matrix operations and graphical visualization of the vectors arrangement.	
	ANSI C	Verilog HDL
code developing	The algorithm code written in ANSI C/C++ was developed in Visual Studio 2019 as a function of the user DLL block for power electronic circuit modelling software – PSIM 11.	Equivalently, the same functionality has been developed using Verilog HDL language using ModelSim - Intel Fpga Starter Edition 10.5b.
simulation and validation	Simulation has been performed using the complete electrical scheme of CMC and RL load using PSIM11 software. The control board was emulated using the DLL user block.	For early validation of the Verilog HDL synthesizability, the Quartus 18.1 has been used with SignalTap Logic Analyzer with conjunction with DE10–Lite evaluation board.
experiment	A 5kW CMC with control board based on multicore ADSP–SC589 DSP from Analog Devices and MAX10 Intel FPGA device. The bidirectional power switch was built using two SiC transistors C3M0075120D.	

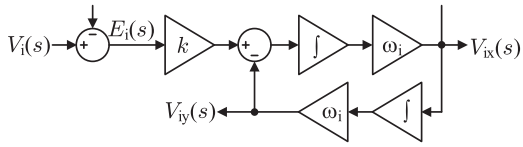

Fig. 2. DSOGI-OSG structure in continuous time domain.

II. Proposal for Changes in the DAV-PWM

The use of an FPGA chip offers new possibilities to design the computation structure with a short execution time, provided that these algorithms do not use advanced mathematical functions, such as trigonometric functions, and do not contain the program loops. Therefore, the computation scheme of the DAV-PWM should be optimized. The most important modifications of the DAV-PWM algorithm are explained in the following subsections.

A. Quadrature Component Generation of an Input Voltage

Each input voltage with the pulsation, ω_i can be expressed as a rotating vector, as shown in Fig. 3. For pure sinusoidal input voltages with an amplitude V, the imaginary coordinates are just

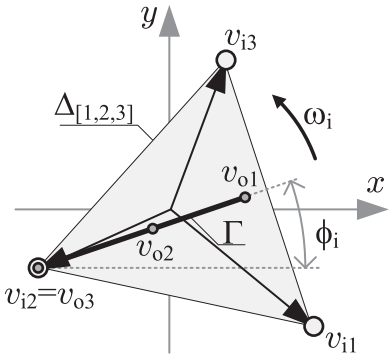

Fig. 3. Regulation of an input power factor in CMC by tilting the trajectory Γ.

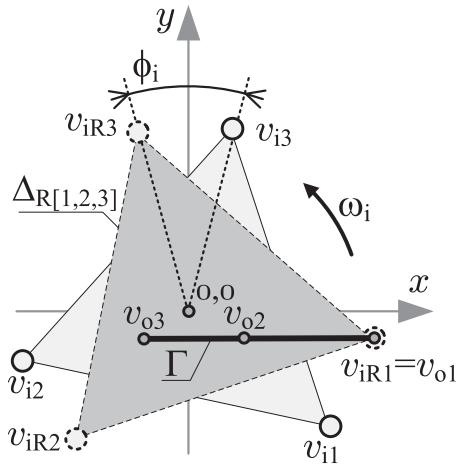

Fig. 4. Schematic diagram of the improved DAV-PWM algorithm.

quadrature components. Thus, the analytic signals corresponding to input voltages can be expressed as follows:

$$\mathbf{v_i} = V \begin{bmatrix} \cos(\omega_i t) & \sin(\omega_i t) \\ \cos(\omega_i t - 2\pi/3) & \sin(\omega_i t - 2\pi/3) \\ \cos(\omega_i t + 2\pi/3) & \sin(\omega_i t + 2\pi/3) \end{bmatrix}. \quad (1)$$

If the input voltages are not perfectly sinusoidal, all coordinates should be determined using Hilbert filter or fast-Fourier-transform/discrete-Fourier-transform-based operation [28]–[30], which cannot be easily implemented in the FPGA without using an advanced intellectual property core. However, calculations using Clarke's triple transforms, although simple, in the case of asymmetry cause a distortion of the input current [26]. In practice, error signals in the form of dc offsets, glitches, and momentary voltage sags may occur in measurements. Therefore, coordinates can be computed by double second-order generalized integrator with loop feedback extension functioning as orthogonal signal generator (DSOGI-OSG), shown in Fig. 2, which in the OSG part prevents unexpected resonance and variable overflow [31]–[33]. If processed signal frequency does not have an exact value, another extension of the second-order generalized integrator structure, called the frequency-locked loop (FLL), may be applied [34]–[36].

B. Simpler Approach to the Input Displacement Angle Regulation

According to the concept proposed in [26] and [27], an input angle displacement regulation is realized by tilting the trajectory Γ by the desired displacement angle, exactly equal to ϕ_i, as illustrated in Fig. 3. The modification of the Γ trajectory results in decreasing the voltage transfer ratio q. Thus, reference output voltages can be represented by the following formula:

$$\mathbf{v_o} = q \cdot \cos(\phi_i) \begin{bmatrix} \cos(\omega_o) + v_{cm} \\ \cos(\omega_o - 2\pi) + v_{cm} \\ \cos(\omega_o + 2\pi) + v_{cm} \end{bmatrix} \begin{bmatrix} 1 & \tan(\phi_i) \end{bmatrix} \quad (2)$$

where the common-mode signal v_{cm} is expressed as follows:

$$v_{cm} = -0.5(\max_o + \min_o)$$

$$\max_o = \mathrm{MAX}\{\cos(\omega_o), \cos(\omega_o - 2\pi/3), \cos(\omega_o + 2\pi/3)\}$$

$$\min_o = \mathrm{MIN}\{\cos(\omega_o), \cos(\omega_o - 2\pi/3), \cos(\omega_o + 2\pi/3)\}. \quad (3)$$

To achieve the maximum voltage transfer ratio, the trajectory Γ should to be shifted to the nearest vertex of the triangle $\Delta_{[1,2,3]}$. According to the original DAV-PWM algorithm, coordinates of the shift vector are designated by the program loop routine, in which the algorithm selects the best candidate among the input voltage vector set [26]. This solution can also be improved to meet the optimization requirements. The Gamma Γ modification in (2) can be replaced by a formula containing a rotation matrix. Thus, the relation between input and output voltages in the CMC may be written in a following general form:

$$\mathbf{v_o} \cdot \boldsymbol{R} = \mathbf{D} \cdot \mathbf{v_i} \quad (4)$$

where

$$\boldsymbol{R} = \begin{bmatrix} \cos(\phi_i) & -\sin(\phi_i) \\ \sin(\phi_i) & \cos(\phi_i) \end{bmatrix} \quad (5)$$

and $\mathbf{D}$ is a square matrix that contains all PWM duty cycles

$$\mathbf{D} = \begin{bmatrix} d_{11} & d_{12} & d_{13} \\ d_{21} & d_{22} & d_{23} \\ d_{31} & d_{32} & d_{33} \end{bmatrix} \quad (6)$$

for switches h_{11}–h_{33}. Taking into account the properties of the $\boldsymbol{R}$ matrix, (4) can be finally rewritten as

$$\mathbf{v_o} = \mathbf{D} \cdot (\mathbf{v_i} \cdot \boldsymbol{R}^{-1}) = \mathbf{D} \cdot \mathbf{v_{iR}}. \quad (7)$$

Now, the desired angle of displacement ϕ_i can be achieved by the angular displacement of the input vector collection (1). This result has a significant impact on the optimization of the original DAV-PWM algorithm because all calculations can be performed for reference voltage, which always has a zero imaginary component. Hence, the selection of the shift coordinates is simplified and is free from an undesired program loop. Reference to the new synthesis field $\Delta_{R[1,2,3]}$ shown in Fig. 4, the shift vector always corresponds to an intermediate vertex between the top and the bottom vertex, which can be immediately selected in

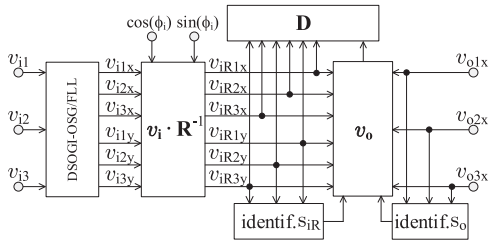

Fig. 5. Identifying the sector of the rotated input vectors using three comparators during an amplitude asymmetry.

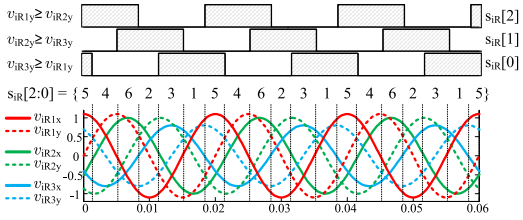

Fig. 6. Three selected source voltage conditions during operation with a unity power factor at the input—simulation using PSIM11 software.

a much simpler way using comparators instead of the program loop.

C. Simplification of the Reference Output Voltage Generation

The trajectory shift operation eliminates the common voltage from (2). Therefore, this component can be deleted in the proposed version of the DAV-PWM algorithm. As a consequence, only the sinusoidal voltage references can be applied in the algorithm.

III. Improved DAV-PWM Algorithm

A schematic diagram of the improved DAV-PWM algorithm is presented in Fig. 5. The input voltage $\{v_{i1}, v_{i2}, v_{i3}\}$ from measurements is converted into analytic signal pairs using the DSOGI-OSG or DSOGI-FLL structure. Next, all input vectors are multiplied by the rotation matrix $\boldsymbol{R}$ (5), which takes arguments—$\cos\phi_i$ and $\sin\phi_i$—from the input power factor control routine. Three coordinates pairs for modulating signals

$$\mathbf{v_o} = \begin{bmatrix} v_{o1x} + v_{sx} & v_{sy} \\ v_{o2x} + v_{sx} & v_{sy} \\ v_{o3x} + v_{sx} & v_{sy} \end{bmatrix} \tag{8}$$

are selected according to the input sector s_{iR} and the output sector s_o. Both sectors are directly identified using comparators, as shown in Figs. 6 and 7.

Dependencies of the shift vector coordinates on the input and output sectors are referred to Table II. The collection of PWM duty cycles for switches h_{11}, h_{21}, and h_{31} can be calculated without trigonometry and angles using the following formulas:

$$d_{11} = \xi \cdot \left| \det \begin{bmatrix} v_{iR2x} - \mathbf{v_o}(1,1) & v_{iR2y} - \mathbf{v_o}(1,2) \\ v_{iR3x} - \mathbf{v_o}(1,1) & v_{iR3y} - \mathbf{v_o}(1,2) \end{bmatrix} \right| \tag{9}$$

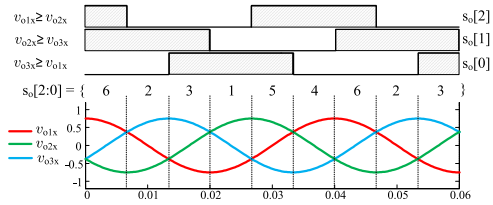

Fig. 7. Two periods of the load voltage for the ten-switch double-sided modulator for the conventional SVM method.

TABLE II
Dependence of the Shift Vector Coordinates v_s, $\max_o$, and $\min_o$ on Input and Output Sector Number

s_{iR} or s_o	v_{sx}	v_{sy}	$\max_o$	$\min_o$
1	$v_{iR2x} - \min_o$	v_{iR2y}	v_{o3x}	v_{o1x}
2	$v_{iR1x} - \min_o$	v_{iR1y}	v_{o2x}	v_{o3x}
3	$v_{iR3x} - \max_o$	v_{iR3y}	v_{o2x}	v_{o1x}
4	$v_{iR3x} - \min_o$	v_{iR3y}	v_{o1x}	v_{o2x}
5	$v_{iR1x} - \max_o$	v_{iR1y}	v_{o3x}	v_{o2x}
6	$v_{iR2x} - \max_o$	v_{iR2y}	v_{o1x}	v_{o3x}

TABLE III
PWM Duty Cycle Matrices $\mathbf{D}$ Within an Input and Output Voltage Sectors in DAV-PWM

s_{iR} \ s_o	4,6 (1,3)	2,3 (4,5)	1,5 (2,6)
2(5)	$\begin{bmatrix} 1 & d_{12} & d_{13} \\ 0 & d_{22} & d_{23} \\ 0 & d_{32} & d_{33} \end{bmatrix}$	$\begin{bmatrix} d_{11} & 1 & d_{13} \\ d_{12} & 0 & d_{23} \\ d_{13} & 0 & d_{33} \end{bmatrix}$	$\begin{bmatrix} d_{11} & d_{12} & 1 \\ d_{12} & d_{22} & 0 \\ d_{13} & d_{32} & 0 \end{bmatrix}$
1(6)	$\begin{bmatrix} 0 & d_{12} & d_{13} \\ 1 & d_{22} & d_{23} \\ 0 & d_{32} & d_{33} \end{bmatrix}$	$\begin{bmatrix} d_{11} & 0 & d_{13} \\ d_{21} & 1 & d_{23} \\ d_{31} & 0 & d_{33} \end{bmatrix}$	$\begin{bmatrix} d_{11} & d_{12} & 0 \\ d_{21} & d_{22} & 1 \\ d_{31} & d_{32} & 0 \end{bmatrix}$
3(4)	$\begin{bmatrix} 0 & d_{12} & d_{13} \\ 0 & d_{22} & d_{23} \\ 1 & d_{32} & d_{33} \end{bmatrix}$	$\begin{bmatrix} d_{11} & 0 & d_{13} \\ d_{21} & 0 & d_{23} \\ d_{31} & 1 & d_{33} \end{bmatrix}$	$\begin{bmatrix} d_{11} & d_{12} & 0 \\ d_{21} & d_{22} & 0 \\ d_{31} & d_{32} & 1 \end{bmatrix}$

$$d_{21} = \xi \cdot \left| \det \begin{bmatrix} v_{iR1x} - \mathbf{v_o}(1,1) & v_{iR1y} - \mathbf{v_o}(1,2) \\ v_{iR3x} - \mathbf{v_o}(1,1) & v_{iR3y} - \mathbf{v_o}(1,2) \end{bmatrix} \right| \tag{10}$$

$$d_{31} = 1 - d_{11} - d_{21} \tag{11}$$

where det means the determinant of the 2×2 matrix, and

$$\xi = \left| \det \begin{bmatrix} v_{iR2x} - v_{iR1x} & v_{iR2y} - v_{iR1y} \\ v_{iR3x} - v_{iR1x} & v_{iR3y} - v_{iR1y} \end{bmatrix} \right|^{-1}. \tag{12}$$

Other PWM duty cycles, in the second and third rows of (8), can be computed analogously. The PWM duty cycle matrices $\mathbf{D}$ within an input s_{iR} and output s_o voltage sectors in DAV-PWM are summarized in Table III. The $\mathbf{D}$ matrix consists of nine duty cycles d_{11}–d_{33}, which are transformed into the sequence of logical signals for switch state control. These sequences are usually generated by the specialized digital structure based on counters and comparators according to the selected commutation strategy in the CMC [37]–[41]. The cyclic Venturini approach and the four-step commutation strategy have been chosen [5].

IV. Power Electronics Simulation, HDL Modeling, and HIL Verification

This section presents results obtained during the functional simulation in PSIM11 software, behavioral modeling of the HDL using ModelSim Intel FPGA environment, and the HIL

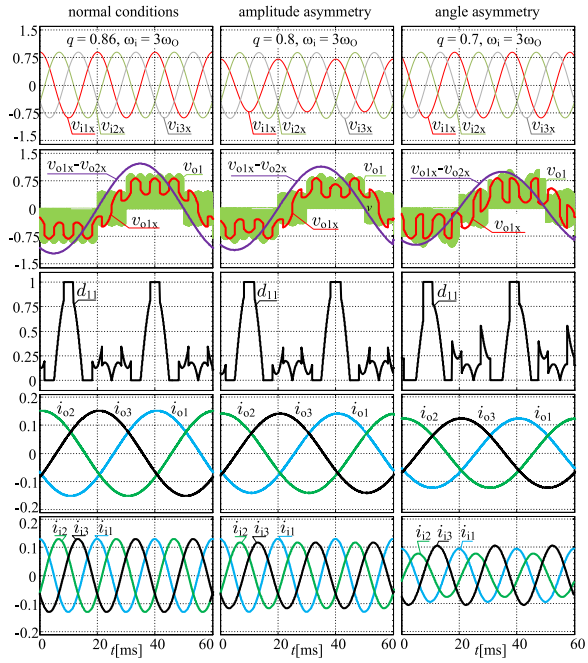

Fig. 8. ModelSim simplified simulation diagram.

verification using Quartus Intel FPGA software with the signal-tap logic analyzer (STLA) tool.

A. Simulation and Switch State Sequences Comparison for DAV-PWM and SVM

The PSIM environment was used to simulate CMC control and verify the algorithm compiled as the user DLL block, which is written in C language. The proposed improved algorithm maintains the important properties of DAV-PWM proposed in [26]. The obtained sequences of switch states, including the waveforms of line-to-line voltages, remain unchanged. An input voltage amplitude asymmetry or phase angle disturbance change the shape and area of the synthesis field limiting the value of the voltage transfer ratio q. However, during operation with a unity power factor, currents on both sides of the converter are sinusoidal, as can be seen in Fig. 8, where three selected input conditions for RL load type are presented. This article proposes a direct modulation algorithm, in which the switch states are not explicitly declared as in the conventional SVM modulation method. Table IV presents all switch states of the CMC. Black dots represent the active switches in the matrix panel.

Several modulation techniques are compared in [42] and analyzed in [43]. Among the described switch state sequences, two of them can be distinguished: ten-switch double-sided sequence shown in Fig. 9 and low-distortion eight-switch sequence illustrated in Fig. 10. The switch states for DAV-PWM can be derived using the switch state parser connected with all nine control signals h_{11}–h_{33}. The state parser is a MATLAB script, which operates on the PSIM data saved in the CSV file format. Decoded SVM switching sequences for DAV-PWM with the cyclic Venturini scheme are collected in Table V.

TABLE IV
SWITCH STATE COLLECTION FOR THE SVM METHOD

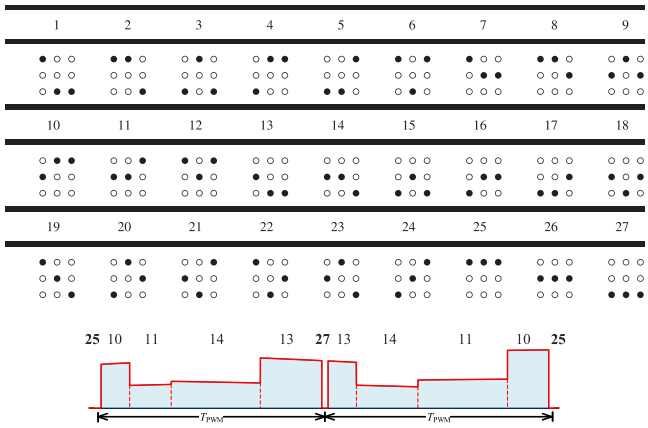

Fig. 9. Waveforms in the analog form obtained by modeling of the DAV-PWM algorithm written in Verilog HDL using ModelSim environment. (a) $q = 0.86$, $\phi_i = 0$, and $\omega_i = 2\omega_o$. (b) $q = 0.75$, $\phi_i = -\pi/6$, and $\omega_i = 2\omega_o$.

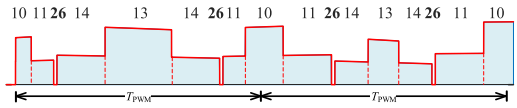

Fig. 10. Early validation scheme of the improved DAV-PWM algorithm developed in Verilog HDL.

TABLE V
DECODED SVM SWITCHING SEQUENCES FOR DAV-PWM WITH THE CYCLIC VENTURINI SCHEME

s_{iR}	s_o 6(1)	2(5)	3(4)
5(2)	25,02,01,02,25,08,07,08,25	25,02,03,02,25,08,09,08,25	25,10,09,10,25,04,03,04,25
4(3)	27,01,02,01,27,13,14,13,27	27,03,02,03,27,15,14,15,27	27,03,04,03,27,15,16,15,27
6(1)	26,11,10,11,26,14,13,14,26	26,11,12,11,26,14,15,14,26	26,07,12,07,26,16,15,16,26
2(5)	25,04,05,04,25,10,11,10,25	25,12,11,12,25,06,05,06,25	25,06,01,06,25,12,07,12,25
3(4)	27,05,04,05,27,17,16,17,27	27,05,06,05,27,17,18,17,27	27,01,06,01,27,13,18,13,27
1(6)	26,07,08,07,26,16,17,16,26	26,18,17,18,26,09,08,09,26	26,18,13,18,26,09,10,09,26

The following major conclusions can be formulated based on the switch state sequence comparison.

1) Both the SVM method and DAV-PWM use the same active vectors in their input and output voltage sectors.
2) The construction of the switch state sequences in the DAV-PWM method is identical to that of sequences in SVM shown in Fig. 10.
3) The proposed DAV-PWM allows for reducing the harmonic distortion by having two zero vectors per period, as reported in [42], but only the approach illustrated in Fig. 9 permits for reduction of switch operation frequency.
4) Tabularizing the switch state sequences is not necessary for a carrier-based modulation such as DAV-PWM.

B. HDL Compilation

The ModelSim software was applied for modeling the digital module of the PWM duty cycle computation, in which the behavioral equivalent of the C-language-developed DAV-PWM

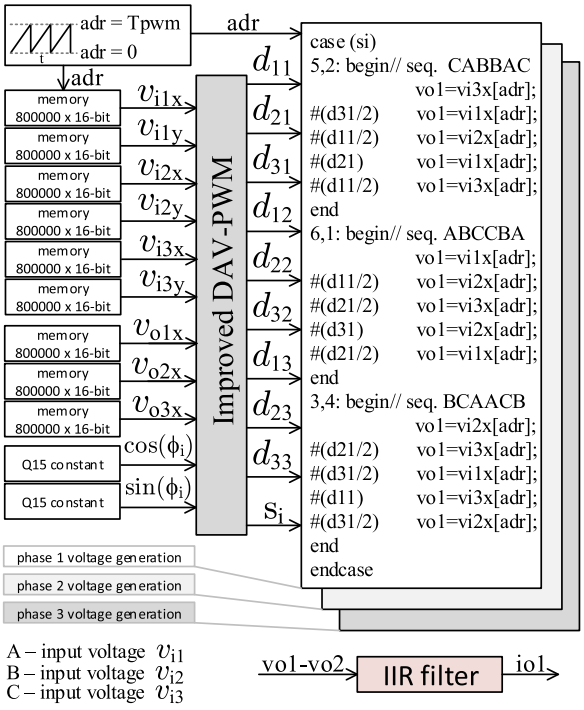

Fig. 11. 5-kW matrix converter with the DSP–FPGA control board.

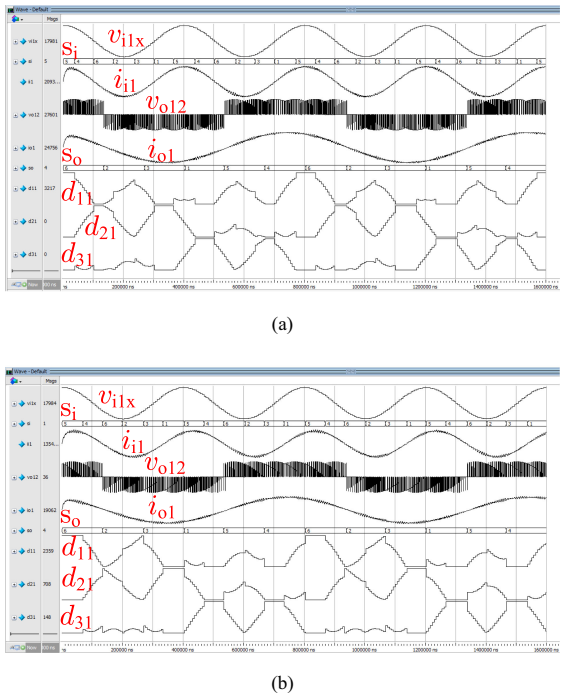

Fig. 12. Schematic diagram of the experiment configuration.

had been coded using Verilog HDL. The ModelSim-simplified simulation diagram is shown in Fig. 11. This software is usually dedicated for digital core or module simulation, but, here, that software has been used for developing the matrix converter control module. The fixed-point Q15 format arithmetic, in comparison with the single-precision format, often permits for developing very fast algorithms without pipelines and recursive operations. Continuous signals, such as input voltages, can be represented by the large tables of 16-bit values. The discrete sampling time was generated by the counter with auto-reload. An input displacement angle has been expressed as a sin–cos pair expressed by constant Q15 values. For simulation purpose only, the cyclic Venturini switching strategy was developed in Verilog HDL using nonsynthesizable modeling based on the signal delay command. This Verilog HDL piece of code is intentionally depicted in the drawing. Selected electrical waveforms, sectors, and PWM duty cycles in the analog form, modeled using ModelSim simulation environment, are shown in Fig. 12. The load current was modeled using the first-order infinite impulse response (IIR) filter. All line-to-line load voltages were generated using the signal switching approach controlled by the input sector number.

C. HIL Verification

For early validation of the Verilog HDL project file synthesizability, the Quartus 18.1 with the STLA was used. An evaluation board DE10-Lite was used during the HIL test. Two debugging tools were used during the validation stage. As shown in Fig. 13, the In-System Sources and Probes tool was used to give phase and frequency of signals generated by CORDIC numerically controlled oscillators, while the STLA tool allowed

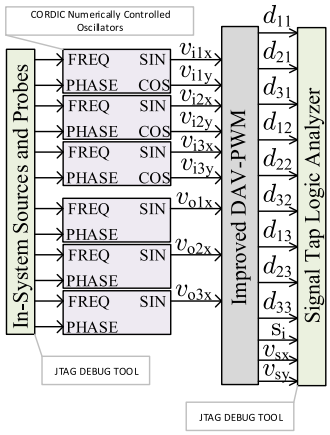

Fig. 13. Block diagram of computation performed by the FPGA device during an experiment.

TABLE VI
IMPROVED DAV-PWM DUTY CYCLE COMPUTATION MODULE RESOURCE UTILIZATION

resource name	utilization
Logic Cells	1454
Dedicated Logic Register	211
DSP Elements	40
DSP 18x18 multipliers	20

to visualize and record the selected signals in real time. Fig. 14 shows improved DAV-PWM algorithm signals during the debug session using the STLA for $\omega_i/\omega_o = 2.67$, $f_s = 1$ MHz, and $q = 0.8$. The proposed PWM duty cycle computation module resource utilization is presented in Table VI.

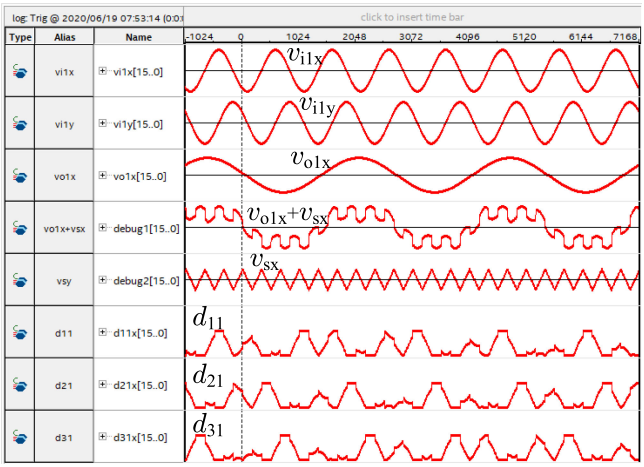

Fig. 14. Experimental results for $\omega_o/\omega_i = 0.5$ and symmetric input amplitudes for (a) unity power factor ($q = 0.86$ and $\phi_i = 0$) and (b) reactive power generation ($q = 0.6$ and $\phi_i = -\pi/4$).

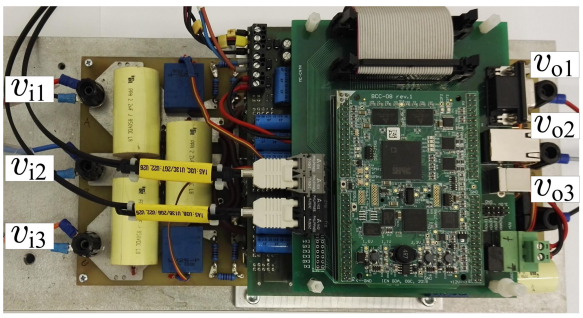

Fig. 15. Output and input currents for asymmetrical input voltage source for DAV-PWM: $q = 0.55$ and $\omega_o/\omega_i = 0.5$.

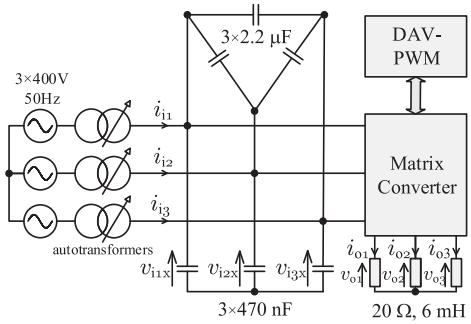

Fig. 16. The schematic diagram of the experiment configuration.

V. Experimental Results

A 5-kW CMC with the control board, shown in Fig. 15, based on the multicore ADSP-SC589 DSP from Analog Devices and the MAX10 Intel FPGA device was used during the experimental stage. The bidirectional power switch was build using two SiC transistors C3M0075120D. The schematic diagram of the experiment configuration is shown in Fig. 16. The block diagram of computation performed by the FPGA device during an experiment is presented in Fig. 17. The Q15 symbol indicates digital structures based on fixed-point arithmetic, while SP means the floating-point IP core used for time cycle scaling. This element

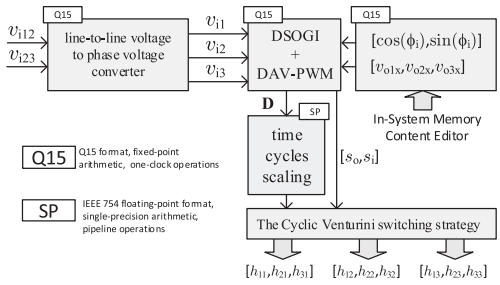

Fig. 17. The block diagram of performed computation by FPGA device during an experiment.

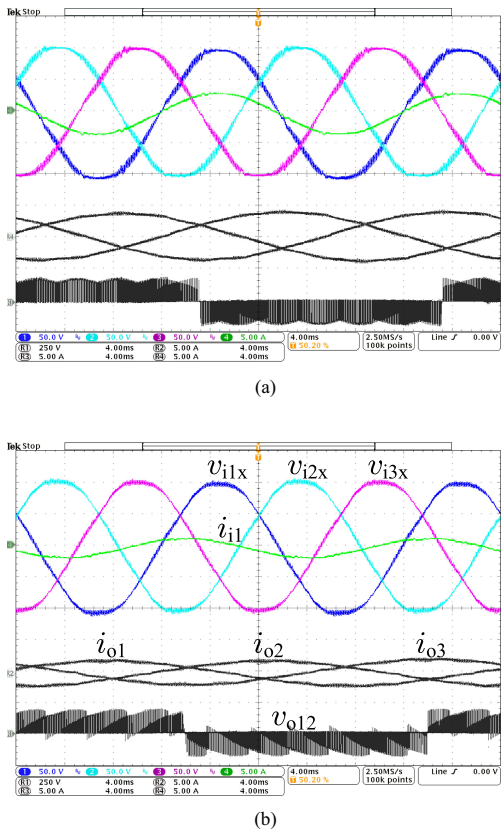

Fig. 18. Experimental results for $\omega_o/\omega_i = 0.5$ and symmetric input amplitudes for: (a) unity power factor, (b) reactive power generation.

is needed to preserve constant modulation frequency. The modulation period was 100 μs, while algorithm computation in Q15 blocks was accomplished through 100-ns positive clock pulse.

The waveforms for the normal operation with a zero displacement angle value ϕ_i and modulation with $\phi_i = -\pi/4$ are shown in Fig. 18. The proposed modulation method can be used in the case of supply voltage asymmetry. Despite such conditions, the input currents in each phase are sinusoidal. Experimental results for a significant asymmetry of input voltages ($V_{i1} = 75$ V, $V_{i2} = 100$ V, and $V_{i3} = 125$ V) are shown in Fig. 19.

VI. Conclusion

The proposed computation scheme of PWM duty cycles did not require trigonometry operation and angles resulting in a

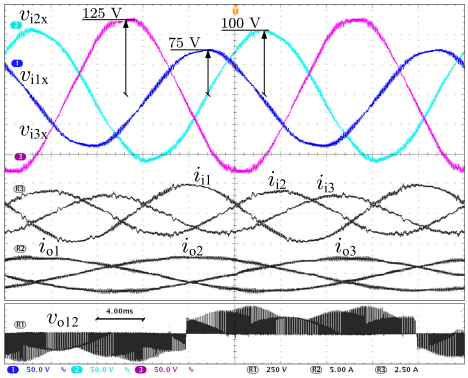

Fig. 19. An output and input currents for asymmetrical input voltage source for DAV-PWM modulation: $q = 0.55, \omega_o/\omega_i = 0.5$.

TABLE VII
COMPARISON OF THE SVM AND THE IMPROVED DAV-PWM

	SVM	DAV–PWM
maximum voltage transfer ratio q if $\cos(\phi_i) = 1$	0.866	0.866
an input angle control	yes	yes
the q dependency from the input angle ϕ_i	$q = 0.866 \cdot \cos(\phi_i)$	$q = 0.866 \cdot \cos(\phi_i)$
trigonometric functions in the formulas for the PWM duty cycles	yes	no
implementation in FPGA using HDL	difficult	simple
execution time of the PWM duty cycle	several hundred clock cycles	one clock
switch states table	switch state sequences are pre–selected and placed in the look–up table	tabularizing of the switch state sequences is not needed
ready for hybrid modulation	no – this algorithm is hermetic and the modulation can not be modified in the fly during the device is running	yes – it is enough to modify the reference voltage waveform
ready for multiphase output (5 or more terminals)	no – both, the optimal switch state sequences and the PWM duty cycles formulas must be elaborated again	yes – a new output phase is represented by the same commutation cell, an only new double multiplier block is required for the PWM duty cycles computing
PLL requirements	yes – for the indication of the output voltage sectors and also the input current sectors	no – there is no PLL needed

simplistic matrix converter control algorithm. All calculation was based on basic arithmetic operations that can be easily implemented in the FPGA within one clock tact. The proposed general direct modulation was an SVM equivalent when the low-distortion eight-switch double-sided modulator was applied. However, compared with SVM, the proposed DAV-PWM algorithm was much simpler. The comparison of the SVM and the improved DAV-PWM was presented in Table VII. With regard to the solution presented in [26], the following benefits were obtained, as presented in Table VIII.

The discussed approach was suitable and more robust for FPGA implementation than a conventional approach such as SVM. For a better understanding of the solution developing process, this article also briefly described the HDL project

TABLE VIII
COMPARISON OF THE PREVIOUS [26] AND THE IMPROVED DAV-PWM ALGORITHMS

	previous	present
maximum voltage transfer ratio q if $\cos(\phi_i) = 1$	0.866	0.866
an input angle control method	the reference voltages' trajectory Γ slope	based on the input voltage rotation matrix $\mathbf{R}$
the extraction method of the quadrature components of the input voltages	the Clarke transformation on each natural axis a–b–c	using the DSOGI–OSG structure
the occurrence of program loops in the algorithm	yes – for selection of the best reference voltage trajectory position inside the synthesis field	no program loop
ready to use in	DSP and processors only using the C-code	C-code and HDL-code allow the use in both, DSP and FGPA based application
shortening the calculation time	—	$\approx 30\%$ in DSP, one–cycle operation in the FPGA device

compilation stage and HIL verification. Moreover, an early functional simulation of the CMC control can be performed using the ModelSim software. Finally, the solution can be represented only by one Verilog HDL file; thus, an export to another FPGA vendor platform is not complicated, and it is not necessary to explicitly predefine the semiconductor switch states as in the SVM methods. Each converter cell is controlled independently in a direct way. This is especially convenient when the number of outputs is greater than the standard three. Simulation files are provided in the IEEE DataPort portal to increase understanding of this article. The first file is a script for the MATLAB environment, which is a presentation of the proposed modulation. The second file contains the testbench of the proposed Verilog HDL modulator [44].

APPENDIX

```
{The Verilog HDL code of the improved DAV-PWM algorithm.}
//module definition
module ImprovedDAVPWM #(parameter MSB=33,LSB=18)
(
 input iCLK,
 input signed [15:0] vi1x,vi2x,vi3x,vi1y,vi2y,vi3y,
 input signed [15:0] vo1x,vo2x,vo3x,R_cos,R_sin,
 output reg[15:0] d11,d21,d31,d12,d22,d32,d13,d23,d33,sum,
 output reg [2:0] si,so
);
//x-coordinate difference
reg signed [16:0] vi1x_vo1x,vi2x_vo1x,vi3x_vo1x;
reg signed [16:0] vi1x_vo2x,vi2x_vo2x,vi3x_vo2x;
reg signed [16:0] vi1x_vo3x,vi2x_vo3x,vi3x_vo3x;
//y-coordinate difference
reg signed [16:0] vi1y_py,vi2y_py,vi3y_py;
//the real line-to-line voltage
reg signed [16:0] vi1x_vi2x,vi2x_vi3x,vi3x_vi1x;
//the imaginary line-to-line voltage
reg signed [16:0] vi1y_vi2y,vi2y_vi3y,vi3y_vi1y;
//minimum and maximum output voltage
reg signed [15:0] min_vox,max_vox;
//shift vector coordinates
reg signed [15:0] vsx,vsy;
//input vector coordinates after rotation
reg signed [31:0] viR1xx,viR2xx,viR3xx,viR1yy,viR2yy,viR3yy;
//rescaled input vector coordinates
reg signed [15:0] viR1x,viR2x,viR3x,viR1y,viR2y,viR3y;
//shifted real coordinates of output voltages
reg signed [15:0] vo1xx,vo2xx,vo3xx;
//rational functions for matrix D calc.
reg signed [33:0] d11w,d21w,d31w,d12w,d22w,d32w,d13w,d23w,d33w,d00w;
//absolute rational functions for matrix D calc.
reg signed [33:0] d11ww,d21ww,d31ww,d12ww,d22ww,d32ww,d13ww,d23ww,d33ww,d00ww;

//behavioral description
always @(posedge iCLK)
```

```
begin
//output voltage sector calculation
so[2]=(vo1x >= vo2x)? 1:0; so[1]=(vo2x >= vo3x)? 1:0; so[0]=(vo3x >=
 vo1x)? 1:0;
//maximum output voltage
case(so)
1: max_vox = vo3x;2: max_vox = vo2x;3: max_vox = vo2x;
4: max_vox = vo1x;5: max_vox = vo3x;6: max_vox = vo1x;
endcase
//minimum output voltage
case(so)
1: min_vox = vo1x;2: min_vox = vo3x;3: min_vox = vo1x;
4: min_vox = vo2x;5: min_vox = vo2x;6: min_vox = vo3x;
endcase
//rotation of the input vectors
viR1xx = vi1x * R_cos - vi1y * R_sin; viR1yy = vi1x * R_sin + vi1y * R_cos;
viR2xx = vi2x * R_cos - vi2y * R_sin; viR2yy = vi2x * R_sin + vi2y * R_cos;
viR3xx = vi3x * R_cos - vi3y * R_sin; viR3yy = vi3x * R_sin + vi3y * R_cos;
//scaling of the rotated input vectors
viR1x=viR1xx[30:15]; viR1y=viR1yy[30:15]; viR2x=viR2xx[30:15];
viR2y=viR2yy[30:15]; viR3x=viR3xx[30:15]; viR3y=viR3yy[30:15];
//input voltage sector calculation
si[2]=(viR1y>=viR2y)? 1:0; si[1]=(viR2y>=viR3y)? 1:0; si[0]=(viR3y>=
 viR1y)? 1:0;
//y-coordinate of shift vector
case(si)
5,2: vsy=viR1y; 4,3: vsy=viR3y; 6,1: vsy=viR2y;
endcase
//x-coordinate of shift vector
case(si)
6: vsx=viR2x-max_vox; 1: vsx=viR2x-min_vox; 3: vsx=viR3x-max_vox;
4: vsx=viR3x-min_vox; 5: vsx=viR1x-max_vox; 2: vsx=viR1x-min_vox;
endcase
//the real line-to-line input voltage
vi1x_vi2x=viR1x - viR2x; vi2x_vi3x=viR2x - viR3x; vi3x_vi1x=viR3x - viR1x;
//the imaginary real line-to-line input voltage
vi1y_vi2y=viR1y - viR2y; vi2y_vi3y=viR2y - viR3y; vi3y_vi1y=viR3y - viR1y;
//shifted the real coordinates of output voltages
vo1xx=vo1x+vsx; vo2xx=vo2x+vsx; vo3xx=vo3x+vsx;
//x-coordinate difference
vi1x_vo1x=viR1x - vo1xx; vi2x_vo1x=viR2x - vo1xx; vi3x_vo1x=viR3x - vo1xx;
vi1x_vo2x=viR1x - vo2xx; vi2x_vo2x=viR2x - vo2xx; vi3x_vo2x=viR3x - vo2xx;
vi1x_vo3x=viR1x - vo3xx; vi2x_vo3x=viR2x - vo3xx; vi3x_vo3x=viR3x - vo3xx;
//y-coordinate difference
vi1y_py=viR1y - vsy; vi2y_py=viR2y - vsy; vi3y_py=viR3y - vsy;
//rational functions
d11w=(vi2x_vo1x*vi3y_py)-(vi3x_vo1x*vi2y_py);
d21w=(vi3x_vo1x*vi1y_py)-(vi1x_vo1x*vi3y_py);
d31w=(vi1x_vo1x*vi2y_py)-(vi2x_vo1x*vi1y_py);
d12w=(vi2x_vo2x*vi3y_py)-(vi3x_vo2x*vi2y_py);
d22w=(vi3x_vo2x*vi1y_py)-(vi1x_vo2x*vi3y_py);
d32w=(vi1x_vo2x*vi2y_py)-(vi2x_vo2x*vi1y_py);
d13w=(vi2x_vo3x*vi3y_py)-(vi3x_vo3x*vi2y_py);
d23w=(vi3x_vo3x*vi1y_py)-(vi1x_vo3x*vi3y_py);
d33w=(vi1x_vo3x*vi2y_py)-(vi2x_vo3x*vi1y_py);
d00w=(vi1x_vi2x*vi3y_vi1y)-(vi1y_vi2y*vi3x_vi1x);
//their absolute value
d00ww = (d00 w<0) ? -d00 w : d00 w; d11ww = (d11 w<0) ? -d11 w : d11 w;
d21ww = (d21 w<0) ? -d21 w : d21 w; d31ww = (d31 w<0) ? -d31 w : d31 w;
d12ww = (d12 w<0) ? -d12 w : d12 w; d22ww = (d22 w<0) ? -d22 w : d22 w;
d32ww = (d32 w<0) ? -d32 w : d32 w; d13ww = (d13 w<0) ? -d13 w : d13 w;
d23ww = (d23 w<0) ? -d23 w : d23 w; d33ww = (d33 w<0) ? -d33 w : d33 w;
 //PWM duty cycles update
sum = d00ww[MSB:LSB];
d11 = d11ww[MSB:LSB]; d21 = d21ww[MSB:LSB]; d31 = d31ww[MSB:LSB];
d12 = d12ww[MSB:LSB]; d22 = d22ww[MSB:LSB]; d32 = d32ww[MSB:LSB];
d13 = d13ww[MSB:LSB]; d23 = d23ww[MSB:LSB]; d33 = d33ww[MSB:LSB];
end
endmodule
```

REFERENCES

[1] J. Rodriguez, M. Rivera, J. Kolar, and P. Wheeler, "A review of control and modulation methods for matrix converters," *IEEE Trans. Ind. Electron.*, vol. 59, no. 1, pp. 58–70, Jan. 2012, doi: 10.1109/TIE.2011.2165310.

[2] L. Helle, K. Larsen, A. Jorgensen, S. Munk-Nielsen, and F. Blaabjerg, "Evaluation of modulation schemes for three-phase to three-phase matrix converters," *IEEE Trans. Ind. Electron.*, vol. 51, no. 1, pp. 158–171, Feb. 2004, doi: 10.1109/TIE.2003.821900.

[3] T. Friedli and J. Kolar, "Milestones in matrix converter research," *IEEJ J. Ind. Appl.*, vol. 1, no. 1, pp. 2–14, Jul. 2012, doi: 10.1541/ieejjia.1.2.

[4] H. M. Nguyen, H.-H. Lee, and T.-W. Chun, "Input power factor compensation algorithms using a new direct-SVM method for matrix converter," *IEEE Trans. Ind. Electron.*, vol. 58, no. 1, pp. 232–243, Jan. 2011, doi: 10.1109/TIE.2010.2044736.

[5] M. Apap, J. Clare, P. Wheeler, and K. Bradley, "Analysis and comparison of ac-ac matrix converter control strategies," in *Proc. IEEE 34th Annu. Conf. Power Electron. Spec.*, Jun. 2003, pp. 1287–1292, doi: 10.1109/PESC.2003.1216774.

[6] W. Xiong *et al.*, "A cost-effective and low-complexity predictive control for matrix converters under unbalanced grid voltage conditions," *IEEE Access*, vol. 7, pp. 43 895–43905, 2019, doi: 10.1109/ACCESS.2019.2908446.

[7] S. Mori, M. Aketa, T. Sakaguchi, H. Asahara, T. Nakamura, and T. Kimoto, "Suppression of punch-through current in 3 kV 4h-SiC reverse-blocking MOSFET by using highly doped drift layer," *IEEE J. Electron Devices Soc.*, vol. 6, pp. 449–453, 2018, doi: 10.1109/JEDS.2018.2819681.

[8] S. Tammaruckwattana, C. Yue, Y. Ikeda, and K. Ohyama, "Comparison of switching losses of matrix converters for commutation methods," in *Proc. 16th Eur. Conf. Power Electron. Appl.*, Aug. 2014, pp. 1–10, doi: 10.1109/EPE.2014.6910995.

[9] Y. Guo, Y. Guo, W. Deng, J. Zhu, and F. Blaabjerg, "An improved 4-step commutation method application for matrix converter," in *Proc. 17th Int. Conf. Elect. Machines Syst.*, Oct. 2014, pp. 3590–3593, doi: 10.1109/ICEMS.2014.7014112.

[10] O. Simon, J. Mahlein, M. N. Muenzer, and M. Bruckmarm, "Modern solutions for industrial matrix-converter applications," *IEEE Trans. Ind. Electron.*, vol. 49, no. 2, pp. 401–406, Apr. 2002, doi: 10.1109/41.9932.

[11] P. W. Wheeler *et al.*, "An integrated 30kw matrix converter based induction motor drive," in *Proc. IEEE 36th Power Electron. Spec. Conf.*, Jun. 2005, pp. 2390–2395, doi: 10.1109/PESC.2005.1581967.

[12] A. Hassan, Y. Savaria, and M. Sawan, "GaN integration technology, an ideal candidate for high-temperature applications: A review," *IEEE Access*, vol. 6, pp. 78 790–78802, 2018, doi: 10.1109/ACCESS.2018.2885285.

[13] M. Ishida, T. Ueda, T. Tanaka, and D. Ueda, "GaN on Si technologies for power switching devices," *IEEE Trans. Electron Devices*, vol. 60, no. 10, pp. 3053–3059, Oct. 2013, doi: 10.1109/TED. 2013.2268577.

[14] D. Lan, P. Das, and S. K. Sahoo, "A high-frequency link matrix rectifier with a pure capacitive output filter in a discontinuous conduction mode," *IEEE Trans. Ind. Electron.*, vol. 67, no. 1, pp. 4–15, Jan. 2020, doi: 10.1109/TIE.2019.2893838.

[15] R. J. Kaplar *et al.*, "Characterization and reliability of SIC- and GaN-based power transistors for renewable energy applications," in Proc.*IEEE Energytech*, May 2012, pp. 1–6, doi: 10.1109/EnergyTech.2012.6304627.

[16] C. Bailey, D. M. Saban, and P. Guedes-Pinto, "Design of high-speed direct-connected permanent-magnet motors and generators for the petrochemical industry," *IEEE Trans. Ind. Appl.*, vol. 45, no. 3, pp. 1159–1165, May/Jun. 2009.

[17] J. Benzaquen, M. B. Shadmand, and B. Mirafzal, "Ultrafast rectifier for variable-frequency applications," *IEEE Access*, vol. 7, pp. 9903–9911, 2019, doi: 10.1109/ACCESS.2019.2891351.

[18] Q. Wu, M. Wang, W. Zhou, X. Wang, G. Liu, and C. You, "Analytical switching model of a 1200 V SiC MOSFET in a high-frequency series resonant pulsed power converter for plasma generation," *IEEE Access*, vol. 7, pp. 99 622–99632, 2019, doi: 10.1109/ACCESS.2019.2930535.

[19] M. Venturini and A. Alesina, "The generalised transformer: A new bidirectional, sinusoidal waveform frequency converter with continuously adjustable input power factor," in *Proc. IEEE Power Electron. Spec. Conf.*, 1980, pp. 242–252.

[20] A. Alesina and M. G. B. Venturini, "Analysis and design of optimum-amplitude nine-switch direct AC-AC converters," *IEEE Trans. Power Electron.*, vol. 4, no. 1, pp. 101–112, Jan. 1989.

[21] A. K. Dey, G. Mohapatra, T. K. Mohapatra, and R. Sharma, "A modified Venturini PWM scheme for matrix converters," in *Proc. IEEE Int. Conf. Sustain. Energy Technol. Syst.*, Feb. 2019, pp. 13–18, doi: 10.1109/ICSETS.2019.8745014.

[22] Z. Malekjamshidi, M. Jafari, and J. Zhu, "Analysis and comparison of direct matrix converters controlled by space vector and venturini modulations," in *Proc. IEEE 11th Int. Conf. Power Electron. Drive Syst.*, Jun. 2015, pp. 635–639, doi: 10.1109/PEDS.2015.7203550.

[23] C. Ortega, A. Arias, C. Caruana, J. Balcells, and G. M. Asher, "Improved waveform quality in the direct torque control of matrix-converter-fed PMSM drives," *IEEE Trans. Ind. Electron.*, vol. 57, no. 6, pp. 2101–2110, Jun. 2010.

[24] M. Rivera *et al.*, "A comparative assessment of model predictive current control and space vector modulation in a direct matrix converter," *IEEE Trans. Ind. Electron.*, vol. 60, no. 2, pp. 578–588, Feb. 2013.

[25] F. Gruson, P. Le Moigne, P. Delarue, A. Videt, X. Cimetiere, and M. Arpilliere, "A simple carrier-based modulation for the SVM of the matrix converter," *IEEE Trans. Ind. Informat.*, vol. 9, no. 2, pp. 947–956, May 2013.

[26] P. Szczepankowski, P. Wheeler, and T. Bajdecki, "Application of analytic signal and smooth interpolation in pulse width modulation for conventional matrix converters," *IEEE Trans. Ind. Electron.*, vol. 67, no. 12, pp. 10011–10023, Dec. 2020, doi: 10.1109/TIE.2019.2956391.

[27] P. Szczepankowski, T. Bajdecki, and R. Strzelecki, "Direct modulation for conventional matrix converters using analytical signals and barycentric coordinates," *IEEE Access*, vol. 8, pp. 22 592–22616, 2020.
[28] A. Reilly, G. Frazer, and B. Boashash, "Analytic signal generation-tips and traps," *IEEE Trans. Signal Process.*, vol. 42, no. 11, pp. 3241–3245, Nov. 1994, doi: 10.1109/78.330385.
[29] L. Marple, "Computing the discrete-time "analytic" signal via FFT," *IEEE Trans. Signal Process.*, vol. 47, no. 9, pp. 2600–2603, Sep. 1999, doi: 10.1109/78.782222.
[30] L. Asiminoael, F. Blaabjerg, and S. Hansen, "Computing the discrete-time "analytic" signal via FFT," *IEEE Ind. Appl. Mag.*, vol. 13, no. 4, pp. 22–33, Jul. 2007, doi: 10.1109/MIA.2007.4283506.
[31] Q. Yan, R. Zhao, X. Yuan, W. Ma, and J. He, "A DSOGI-FLL-based dead-time elimination PWM for three-phase power converters," *IEEE Trans. Power Electron.*, vol. 34, no. 3, pp. 2805–2818, Mar. 2019.
[32] Z. Xin, R. Zhao, F. Blaabjerg, L. Zhang, and P. C. Loh, "An improved flux observer for field-oriented control of induction motors based on dual second-order generalized integrator frequency-locked loop," *IEEE Trans. Emerg. Sel. Topics Power Electron.*, vol. 5, no. 1, pp. 513–525, Mar. 2017.
[33] N. Hoffmann, R. Lohde, M. Fischer, F. W. Fuchs, L. Asiminoaei, and P. B. Thøgersen, "A review on fundamental grid-voltage detection methods under highly distorted conditions in distributed power-generation networks," in *Proc. IEEE Energy Convers. Congr. Expo.*, 201, pp. 3045–3052.
[34] P. Rodriguez, A. Luna, M. Ciobotaru, R. Teodorescu, and F. Blaabjerg, "Advanced grid synchronization system for power converters under unbalanced and distorted operating conditions," in *Proc. 32nd Annu. Conf. IEEE Ind. Electron.*, 2006, pp. 5173–5178.
[35] K. R. Patil and H. H. Patel, "Modified dual second-order generalised integrator FLL for synchronization of a distributed generator to a weak grid," in *Proc. IEEE 16th Int. Conf. Environ. Elect. Eng.*, 2016, pp. 1–5.
[36] X. He, H. Geng, and G. Yang, "A generalized design framework of notch filter based frequency-locked loop for three-phase grid voltage," *IEEE Trans. Ind. Electron.*, vol. 65, no. 9, pp. 7072–7084, Sep. 2018.
[37] P. W. Wheeler, J. Clare, and L. Empringham, "Enhancement of matrix converter output waveform quality using minimized commutation times," *IEEE Trans. Ind. Electron.*, vol. 51, no. 1, pp. 240–244, Feb. 2004.
[38] H. Nguyen and H. Lee, "A modulation scheme for matrix converters with perfect zero common-mode voltage," *IEEE Trans. Power Electron.*, vol. 31, no. 8, pp. 5411–5422, Aug. 2016.
[39] H. She, H. Lin, B. He, X. Wang, L. Yue, and X. An, "Implementation of voltage-based commutation in space-vector-modulated matrix converter," *IEEE Trans. Ind. Electron.*, vol. 59, no. 1, pp. 154–166, Jan. 2012.
[40] H. Hara, E. Yamamoto, Jun-Koo Kang, and T. Kume, "Improvement of output voltage control performance for low-speed operation of matrix converter," *IEEE Trans. Power Electron.*, vol. 20, no. 6, pp. 1372–1378, Nov. 2005.
[41] J. Mahlein, J. Igney, J. Weigold, M. Braun, and O. Simon, "Matrix converter commutation strategies with and without explicit input voltage sign measurement," *IEEE Trans. Ind. Electron.*, vol. 49, no. 2, pp. 407–414, Apr. 2002.
[42] K. B. Larsen, A. H. Jorgensen, L. Helle, and F. Blaabjerg, "Analysis of symmetrical pulse width modulation strategies for matrix converters," in *Proc. IEEE 33rd Annu. IEEE Power Electron. Spec. Conf.*, 2002, vol. 2, pp. 899–904.
[43] L. Helle and S. Munk-Nielsen, "A novel loss reduced modulation strategy for matrix converters," in *Proc. IEEE 32nd Annu. Power Electron. Spec. Conf.*, 2001, vol. 2, pp. 1102–1107.
[44] P. Szczepankowski, "Direct analytic voltage pulse width modulation using FPGA and Verilog HDL," 2020. [Online]. Available: https://dx.doi.org/10.21227/8xgm-6h16

Pawel Szczepankowski (Member, IEEE) received the Ph.D. degree in electrical engineering in 2009 and the D.Sc. degree in power electronics, in 2020 both from the Gdańsk University of Technology, Gdańsk, Poland.

He authored or coauthored more than 50 scientific and technical papers. He deals with designing and programming microprocessor control systems with digital signal processors and field-programmable gate array devices. His research interests include design, control, diagnostics, modeling and simulation of power electronic converters, including multilevel, matrix, and modular topologies.

Wojciech Sleszynski (Member, IEEE) received the M.Sc. degree in control engineering and the Ph.D. degree in electrical engineering from the Gdańsk University of Technology, Gdańsk, Poland, in 2000 and 2007, respectively.

Since 2001, he has been with the Faculty of Electrical and Control Engineering, Gdańsk University of Technology, where he is currently an Assistant Professor. His research and teaching interests include power electronics, digital signal processing, semiconductor switch diagnostic, and control of the energy conversion.

Tomasz Bajdecki received the M.S. degree in electrical engineering from the Czestochowa University of Technology, Czstochowa, Poland, in 1992, and the Ph.D. degree in electrotechnical from the Gdańsk University of Technology for dissertation "Control strategy for the matrix converter," Gdańsk, Poland, in 2003.

He is currently a Research Staff Member with the Institute of Power Engineering Gdańsk, Gdańsk. His main research interests include control of the high-power converters.