Implementation of Hardware Architecture for SVPWM with Arbitrary Parameters

LUIGI DI BENEDETTO1, (Senior Membership), ANDREA DONISI1, (Student Membership), GIAN DOMENICO LICCIARDO1, (Senior Membership), ROSALBA LIGUORI1, (Membership), EDUARDO PICCIRILLI2, EMILIO LANZOTTI2, ALFREDO RUBINO1, (Membership)

1Department of Industrial Engineering, University of Salerno, Fisciano (SA), 84084, Italy (e-mail: ldibenedetto@unisa.it, adonisi@unisa.it, gdlicciardo@unisa.it, rliguori@unisa.it, arubino@unisa.it).
2DACC Generic Product and Wayside Devices Hitachi Rail STS, Napoli, 80010, Italy (e-mail: eduardo.piccirilli@hitachirail.com, emilio.lanzotti@hitachirail.com).

Corresponding author: A. Donisi (e-mail: adonisi@unisa.it).

ABSTRACT A novel hardware digital architecture for the Space Vector Pulse Width Modulation technique is proposed. Its characteristics are the reduced demand for resources and the possibility to change in real-time the values of the carrier and switching frequencies, of the phase and of the amplitude of the three-phase output voltages without external reference signals or processor. The basic idea is to pre-calculate a set of normalized dwell-time for only one sixth of the $\alpha - \beta$-plane and, then, to reconstruct through our architecture the effective dwell-times and the right inverter configurations by an optimized management of the memory. The architecture is implemented in a Field Programmable Gate Array Cyclone V using the 6% and the 1.01%, respectively, of the Look-Up Tables and of the Flip Flops and the experimental measurements show the goodness of the generated waveforms and the high numbers of degrees of freedom without the demand of Digital Signal Processor or Personal Computer.

INDEX TERMS Digital controller, Field Programmable Gate Array, Space Vector Pulse Width Modulation, Three-phase DC/AC power converter.

I. INTRODUCTION

The modern power converters are complex electronics systems where Analog/Digital Converters, decision logic circuits, auxiliary supply circuits, communication circuits are part of a single unit together with the power electronic circuits. Digital controllers can manage them and also perform the mere control of the power converter but requires a high degree of complexity in terms of calculations and of resources. For example, in some applications a multi stage of power converters, like insulated DC/DC converter followed by DC/AC inverter [1], is frequently used and a customized digital controller can drive both converters, differently from the past where a single controller is dedicated to each single power stage. Programmable Logic Devices, PLD, like micro controller [2], Digital Signal Processor, DSP, [3] or Field Programmable Gate Array, FPGA, [4], are good candidate for such kind of applications thanks to the good performances of elaborations and to the ease of use [5]. In the mentioned scenario, even if the elaboration speed of the new processors increases, it also needs the capability of parallel processing for the management of the auxiliary circuits as well as for the overall computing load. Therefore, the design of the new HardWare, HW, architectures focuses on reducing the use of advanced processors, like DSPs, in order to keep them for other features, as well as the resource utilization and the power dissipation. Moreover, in the scenario of the obsolescence of integrated circuits that is relevant for all those industrial applications where the product life is longer than ten years [6], Application Specific Integration Circuits, ASICs, are more expensive and difficult for firmware updating than PLD, like FPGAs. Moreover, the recent introduction of the wide-bandgap power devices, like Silicon Carbide DMOSFETs [7]–[9] and Gallium Nitride HEMTs [10]–[12], permits to increase the switching frequency over hundreds of kHz and new HW architecture of the PLDs are required in order to optimize their resources. Indeed, stand alone systems needs a single PLD able to manage both power electronics and the auxiliary circuits and, also, their large scale produc-
The paper is organized in this way: in Section II the basic theory of the SVPWM control technique is reported, whereas the theory and the implementation of our HW architecture are, respectively, in Section III and IV; in Section V we implement our proposal and compare the results from the state-of-art; Section VI reports the conclusions.

II. BASIC THEORY OF SVPWM

Let us consider the three-phase inverter of Fig. 1.a where \( P_{A+}, P_{B+}, P_{C+} \) are the top transistors and \( P_{A-}, P_{B-}, P_{C-} \) are the bottom transistors. \( V_{DC} \) is the total DC input voltage and \( v_A, v_B, v_C \) are the phase voltages refer to ground. We can generate an output three-phase sinusoidal waveform at a desired carrier frequency, \( f_C \), and amplitude, \( V_M \), using a modulation signal at a switching frequency, \( f_{SW} \), if the gate driver signals of the transistors are opportunely timed. Although there are several techniques to generate the gate drive signals, the SVPWM has better performance in terms of stator current quality and harmonic wave generation [27]. An example of temporal sequence of gate signals of the top transistors in a switching period is reported in Fig 1.b: the interval times of each configuration of the inverter, \( S_i \), are called dwell-times, i.e. \( T_0, T_A \) and \( T_B \). It is worth to note that 1 means that the top transistor is \( ON \) and the bottom one is \( OFF \), instead 0 is the opposite configuration.

The SVPWM is based on the representation of the three-phase voltage system through the reference vector, \( \overline{V}_{REF} \), in the \( \alpha – \beta \) plane of Fig.2 obtained from the following two-phase transformation:

\[
\overline{V}_{REF} = \begin{bmatrix} V_A \\ V_B \end{bmatrix} = \begin{bmatrix} \frac{2}{\sqrt{3}} & -\frac{1}{\sqrt{3}} \\ 0 & \frac{\sqrt{3}}{3} \end{bmatrix} \begin{bmatrix} v_A \\ v_B \end{bmatrix}
\]

that can be also expressed in the vector form with amplitude \( V_{REF} = \sqrt{V_A^2 + V_B^2} \) and angle respect to the \( \alpha \)-axis \( \theta = \arctan(V_B/V_A) \), as shown in Fig.2.

Once \( \overline{V}_{REF} \) is defined in the \( \alpha – \beta \) plane, it is projected along the two closest vectors, \( \overline{V}_A \) and \( \overline{V}_B \): indeed, observing

![FIGURE 1](image)

(a) Basic three-phase power inverter. Temporal sequence of the top transistors in a switching period (b) for a odd sector and (c) for a even sector where \( T_A = T_A' + T_A'', T_B = T_B' + T_B'', \) and \( T_0 = T_0' + T_0'' + T_f \).
where $T$ and $\theta$ are, respectively, $(000)$ and $(111)$. The decomposition of the $V_{REF}$ is expressed as follows:

$$\int_{0}^{T_{SW}} V_{REF} \, dt = \int_{0}^{T_{0}} V_{0/\tau} \, dt + \int_{0}^{T_{A} + T_{0}} V_{A} \, dt + \int_{T_{B} + T_{A} + T_{0}}^{T_{SB} + T_{A} + T_{0}} V_{B} \, dt \quad (2)$$

where $T_{SW} = f_{SW}^{-1}$. Solving (2) we have:

$$V_{REF} = f_{SW}T_{0}V_{0/\tau} + f_{SW}T_{A}V_{A} + f_{SW}T_{B}V_{B} \quad (3)$$

where $V_{0}$ and $V_{7}$ are assumed null vectors. Taking into account that the six vectors $V_{i}$ are arranged to each other of an angle $\pi/3$, the well-known expressions of the dwell-times are obtained [24]:

$$T_{A}(i) = \frac{m_{a}\sqrt{3}}{2f_{SW}} \sin \left[ S_{i} \frac{\pi}{3} - \theta(i) \right] \quad (4a)$$

$$T_{B}(i) = \frac{m_{a}\sqrt{3}}{2f_{SW}} \sin \left[ \theta(i) - (S_{i} - 1) \frac{\pi}{3} \right] \quad (4b)$$

$$T_{0}(i) = T_{SW} - T_{A}(i) - T_{B}(i) \quad (4c)$$

where $i$ represents the $i$-th $T_{SW}$-instant and $m_{a} = 2V_{M}/V_{DC} \leq 2/\sqrt{3}$ the modulation index. It is worth to note that, for a given sector, $V_{A}$ and $V_{B}$ refer to the vectors, respectively, with the smallest and the largest angle, i.e., for $S_{i} = 1$, $V_{A} = V_{2}$ and $V_{B} = V_{2}$, or for $S_{i} = 4$, $V_{A} = V_{2}$ and $V_{B} = V_{5}$. Moreover, the discretization of the circumference depends on $f_{SW}$ and $f_{C}$ and the smallest angle, $\Delta \theta$, is equal to $2\pi f_{C}/f_{SW}$. The SVPWM technique uses (4) to evaluate the dwell-times and defines their sequence for a single $T_{SW}$. The complexity to implement SVPWM in a digital controller is precisely the calculation of trigonometric functions and the sequence of the dwell-times, i.e., Fig.1b or c, in order to have an output signal with low harmonic distortion and low device power dissipation. In this paper we propose a HW architecture for the implementation of the SVPWM in a digital controller that solves such limitations.

III. THEORY OF THE PROPOSED HW ARCHITECTURE

Generally, the SVPWM are performed by using an external processor, like a DSP or a PC, together with a controller, like microcontroller or FPGA. The external processor needs to generate the three-phase reference voltage with the desired $V_{M}$ and $f_{C}$ that is transformed through (1) in the vector form to calculate $V_{REF}$ and $\theta$. Then, the digital controller evaluates the dwell-times and the inverter configurations. Such approach is unwanted when one needs a stand alone or an independent controller. On the other hand, other solutions use an internal memory [25] where the values of the dwell-times are stored for a fixed $f_{C}$ and $f_{SW}$, but without the possibility to vary them in real-time. We overcome such limits and, although it is based on the approach to store the dwell-times in an internal memory of the controller, our HW architecture satisfies the following points:

- all the system is integrated in the digital controller;
- the external processor is avoided or is just used to configure the system;
- $V_{M}$, $f_{C}$, $f_{SW}$ and $\theta$ can be varied in real-times;
- the HW resources are reduced thanks to an optimization of the required memory.

The procedure of our proposal can be summarized in the following steps:

i the pre-calculation of the normalized dwell-time in terms of the minimum carrier frequency, $f_{Cn}$, of the maximum switching frequency, $f_{SWn}$, and of the maximum modulation index, $m_{a}$;

ii the storing of the pre-calculated data in the internal Look-Up Table, LUT, of the digital controller;

iii the setting in real-time of the $f_{C}$, $f_{SW}$, $\theta$ and of the percentage of the modulation index, $m$, which are the input of the controller;

iv the calculation of the effective dwell-times through the HW architecture;

v the generation of the six gate drive signals including the dead-time for the power transistors.

In the next subsections, the modeling of the proposed SVPWM algorithm is reported and, in particular, we show the evaluation of the normalized dwell-time and the managing of the internal memory in order to obtain the desired dwell-times.

A. NORMALIZED DWELL-TIME

Observing the vector representation of Fig. 2, the six sectors are equally divided in angles of $\pi/3$ and, when $V_{REF}$ is in a generic sector $S_{i}$, we can calculate from (4) the dwell-times referring either to the angle $\theta$ or to the angle $\phi$, i.e., $\phi = \theta - (S_{i} - 1)\pi/3$. Moreover, in each sector the projections of $V_{REF}$ are symmetric around the angle $(S_{i} - 0.5)\pi/3$, for example for $S_{i} = 2$ at $\theta = \pi/2$ we have $T_{A} = T_{B}$ or we obtain $T_{A}(\theta = \pi/3) = T_{B}(\theta = 2\pi/3)$. Taking into account such considerations and from (4a) and (4b), the normalized
dwell-time, $T_N$, is expressed in the following equation:

$$T_N(i) = \frac{m_n\sqrt{3}}{2f_{SW_n}} \sin[\phi(i)] \frac{f_{CLK}}{100}$$  \hspace{1cm} (5)$$

where $f_{CLK}$ is the clock of the controller, 100 is for the variation of $m_n$ in real-time from 0% to 100% of its value, $l$ is a constant. The variable is $\phi$ and its values are from 0 to $\pi/3$ with an incremental step of $\Delta \phi_n = 2\pi f_{Cn}/f_{SW_n}$, whereas the variation of the sector is given by the right selection of the inverter configuration. Indeed, once the starting angle is located and, hence, also the starting sector, the sequence of the sectors is always the same and in succession. The constant $l$ is introduced in order to have integer numbers of the normalized dwell-times so that the counter of the HW architecture can easily perform the counting of the times. Its value is $l = 2^4$ where:

$$L = \text{int} \left[ \log_2 \left( \text{int} \left[ \frac{200}{m_n\sqrt{3}} \sin (\Delta \phi_n) f_{CLK} \right] \right) \right] + 1$$  \hspace{1cm} (6)$$

and it has been calculated by equaling the smallest value of $T_N$ to one, i.e. using $\phi(i) = \Delta \phi_n$ and placing (5) equal to 1.

**B. MANAGING OF THE LUT**

The size of the memory is reduced because only one sixth of the circumference is stored. Indeed, once $T_N$ are evaluated from (5), they are stored in a single LUT of the digital controller. The size of LUT and of its cells are equal, respectively, to:

$$N_{LUT} = \text{int} \left[ \frac{\pi/3}{\Delta \phi_n} \right] + 1 = \text{int} \left[ \frac{f_{SW_n}}{6f_{Cn}} \right] + 1$$  \hspace{1cm} (7a)$$

$$N_{CELL} = \text{int} \log_2 \left( \text{int} \left[ \frac{3m_n f_{CLK} \sin (\Delta \phi_n) f_{SW_n}}{400} \right] \right) + 1$$  \hspace{1cm} (7b)$$

where (7b) is calculated by placing $\phi(i) = \pi/3$ in (5). Moreover, we obtain a further reduction of the resource occupation because the reconstruction of the two dwell-times is performed by a double indexing of the memory: as shown in Fig. 3.a, the LUT memory is pointed by two indexes, $ind_A$ and $ind_B$, from which the data for the calculation of $T_A$ and $T_B$, respectively, are extracted. In particular, for each sector $s_i$, $ind_A$ increases from 1 to $N_{LUT} - 1$, whereas $ind_B$ decreases from $N_{LUT}$ to 2. When $\theta$ is defined, we can find the starting cells of the LUT in two sequential steps:

1. the identification of $s_i$, is done through a cascading comparators having $\theta$ as inputs:

$$S_i = \begin{cases} 
1 & \text{if } 0 \leq \theta < \pi/3 \\
2 & \text{if } \pi/3 \leq \theta < 2\pi/3 \\
3 & \text{if } 2\pi/3 \leq \theta < \pi \\
4 & \text{if } \pi \leq \theta < 4\pi/3 \\
5 & \text{if } 4\pi/3 \leq \theta < 5\pi/3 \\
6 & \text{if } 5\pi/3 \leq \theta < 2\pi 
\end{cases}$$ \hspace{1cm} (8)$$

2. the evaluation of $ind_A$ and $ind_B$ is as follows:

**C. EVALUATION OF THE EFFECTIVE DWELL-TIMES**

Once the $T_N$-values are stored into the internal LUT, our HW architecture calculates the effective dwell-times, which are expressed in times of clock system pulses, through the following operations:

$$T_{A/B}(i) = \frac{n_{SW} m_n}{l} T_N(ind_{A/B})$$  \hspace{1cm} (12a)$$

$$T_0(i) = n_{SW} \text{int} \left[ \frac{f_{CLK}}{f_{SW}} \right] - T_A(i) - T_B(i)$$  \hspace{1cm} (12b)$$

For example, when $\theta = \pi/3$ the starting sector is 2 and $ind_A = 1$ and, then, $ind_A$ increases of 4 when $n_{SW} = 4$ and $n_C = 1$ (see Fig.3.b), of 3 when $n_{SW} = 1$ and $n_C = 3$ (see Fig.3.c), and of 6 when $n_{SW} = 2$ and $n_C = 3$ (see Fig.3.d).

The values of the minimum $f_{SW}$ and of the maximum $f_C$ are defined by the Nyquist-Shannon sampling theorem [28], from which, defined the accuracy of the reconstruction through the integer number $s$, i.e. $f_{SW} \geq s f_C$, we can find the respective values of $n_{SW}$ and $n_C$ from (7a) and (10) as follows:

$$n_{SW} n_C \leq \frac{f_{SW}}{s f_C} = 6 \frac{N_{LUT} - 1}{s}$$  \hspace{1cm} (11)$$
In Fig. 4 we report the effective dwell-times evaluated with our approach through the software MATLAB [29]. First of all, we defined $f_{CLK} = 50\, MHZ$, $f_{CN} = 10\, Hz$, $f_{SW} = 100k\, Hz$, and $m_a = 1.06$: obtaining $l = 2^9 = 512$ from (6), a LUT size of $N_{LUT} = 1667$ cells from (7a), a cell size of $N_{CELL} = 11$ bits from (7b), and a limit of the sampling of $n_{CNSW} \leq 99.9$ from (11) for $s = 100$, which depends on the user degree of accuracy. After that, we used (12) and the sequential access to the LUT (see Section III-B) to calculate the effective dwell-times. In Fig. 4.a and b the three effective dwell-times are reported for $m = 100$ and $n_{SW} = 1$, instead the carrier frequency has been changed from $10\, Hz$ to $50\, Hz$, i.e. $n_C = 1$ (see Fig. 4.a) to $n_C = 5$ (see Fig. 4.b); although the values of the effective dwell-times are the same due to the same multiplication factor in (12), they show a different $f_C$. Moreover, six intervals divide each carrier period and correspond to the six sectors of Fig. 2. In Fig. 4.b and c $f_{SW}$ has been changed from $100k\, Hz$ to $20k\, Hz$, i.e. $n_{SW} = 1$ (see Fig. 4.b) to $n_{SW} = 5$ (see Fig. 4.c), whereas the amplitude changed from $100\%$ to $69\%$ of $m_a$ (see Fig. 4.c). First of all, we can observe the increase of the values of the effective dwell-times when $n_{SW} = 5$ because there is a direct dependency on it in (12); then, for the same $f_{SW}$ the reduction of $m$ induces a decrease of $T_A$ and $T_B$ and an increase of $T_0$, because the reduction of the output three-phase voltage amplitudes can be achieved setting the inverter in the null configuration for more time. For all the cases the symmetry of the effective dwell-times in each sector interval time is evident.

IV. HW ARCHITECTURE

The functional block diagram of Fig. 5 shows the HW implementation of our system, which we called SVPWM BLOCK. The DATA IN are $\theta$, $n_C$, $n_{SW}$, $m$, and $L/R$ and can be changed in real-time during the controller operation. The first four inputs have the same meaning of the Section III instead the signal $L/R$ defines the phase of the three-phase output voltages and it can be useful, for example, to rotate clockwise and counterclockwise the rotor of an asynchronous motor [30]. The output signals are the six gate driver signals and are towards the power converter. The main blocks of the system can be arranged in three subsystems and each of them is described in the following subsections.

A. LUT AND COUNTERS

The first subsystem is formed by the internal memory LUT and by the two counters, POINTERS and COUNTER. The LUT is a memory block in which $T_N$, evaluated from (5), is stored. The sizes of the LUT and of its cells are, respectively, $N_{LUT}$ from (7a) and $N_{CELL}$ from (7b). The two indexes of the LUT, $ind_A$ and $ind_B$, point to two different positions in order to give the two normalized dwell-times, $T_{AN}$ and $T_{BN}$, that are sent to the next subsystem (see Section IV-B) for the evaluation of the effective dwell-times. The POINTERS block manages the indexes and its inputs are $\theta$, $n_C$ and $n_{SW}$. During the compilation process of the digital controller, POINTERS needs the value of $N_{LUT}$ so that the reset of the counter can be set. Furthermore, POINTERS keeps track of $S_i$ through ACTUAL SECTOR signal controlling the SECTOR SELECT block: the starting sector is defined from a cascading comparators and the starting indexes are from (9). Then, the POINTERS increases and decreases, respectively, $ind_A$ and $ind_B$ of the quantity equal

![FIGURE 4. Dwell-times values for a) $n_C = 1$ $n_{SW} = 1$ $m = 100$, for b) $n_C = 5$ $n_{SW} = 1$ $m = 100$, and for c) $n_C = 5$ $n_{SW} = 10$ with $m = 69$ and 100. $f_{SW} = 100k\, Hz$, $f_{CN} = 10\, Hz$, $m_a = 1.06$, and $f_{CLK} = 50\, MHz$.](image)

![FIGURE 5. Functional block diagram of the proposed HW architecture.](image)
to $n_{CnSW}$ and, when $ind_A \geq N_{LUT}$ or, similarly, $ind_B \leq 1$, it resets its counters and increases $ACTUAL\ SECTOR$ by one up to 6, after that restarts to 1.

POINTER operations are done when the $ENABLE$ signal from the $COUNTER$ block is high, whereas, once they are completed, the $DONE$ signal is sent to the $COUNTER$ that ensures the right inverter configuration with the needed dwell-times. In Fig.6 the flow chart of the POINTER operation is shown.

The $COUNTER$ block has seven input signals, that are the effective dwell-times for each sub-interval time of the switching periods (see Fig. 1.b and c), and the $DONE$ input signal in addition to the LSB of $ACTUAL\ SECTOR$ signal. The flux diagram of Fig.7 is performed by the $COUNTER$ and described as follows:

- IDLE: in this state the $COUNTER$ sets the multiplexer to "00", the $ENABLE$ signal to "1", the $START$ signal to "0", and the internal variable $J$ to 0. It remains in this state until the $DONE$ signal changes to "1";
- Set-Up: once $DONE = 1$, the counting preparation for a single $T_{SW}$ is performed by setting $ENABLE = 0$ and by organizing the sequences of the effective interval times from the $TIME\ DIVISOR$ block with the $MUX$ signal as reported in Tab.1 through a check of the LSB of the $ACTUAL\ SECTOR$ signal. Then, $START$ is set to "1";
- CONFIG: the condition of the counting stop as well as the relative configuration of the multiplexer depend on the value of $J$ and in this state they are imposed together with $COUNT = 1$. Then, each time the $COUNT$ becomes "0", $J$ increases by one up to reach 8 where the $COUNTER$ block enters in IDLE state;
- COUNT: it is a simple count state from 0 to the value setting by CONFIG state.

In Tab.1 the two sequences of the dwell-times and of the related inverter configurations give a reduced harmonics distortion and a lower transistor power dissipation, because in each transition only one transistor switches.

### B. CALCULATION OF THE EFFECTIVE DWELL-TIMES

This sub-system is composed by the DENORMALIZATION, $T_0$ – CALCULATION, and $TIME\ DIVISOR$ blocks of Fig.5 and calculates the seven sub-interval times for a single $T_{SW}$ from the values of $T_{AN}$ and $T_{BN}$. The functional block diagrams for each of them are shown in Fig.8.

Normalized dwell-times together with $n_{SW}$ and $m$ are the inputs of the DENORMALIZATION block that perform the calculation of $T_A$ and $T_B$ by using (12a). Taking into account that for multiplication operation the bits length of the product is the sum of the bits lengths of the factors, we
performs (12a) in three steps without loss in accuracy, as reported in Fig.8.a: firstly, the multiplication of $T_{AN/BN}$ and $m$ is done; secondly, a left bit shifting is executed for $L$ bit positions; finally, this last is multiplied by $n_{SW}$. Once $T_A$ and $T_B$ are calculated, the effective dwell-time $T_0$ is evaluated in the $T_0 - CALCULATION$ block that performs (12b) using as inputs $n_{SW}$, $T_A$ and $T_B$ and the functional block diagram is in Fig.8.b. It is worth to note that $L$, evaluated from (6), and $\text{int}[f_{CLK}/f_{SW}]$ are constant.

Before sending the effective dwell-times to the $COUNTER$ block, the $TIME DIVISOR$ block (see Fig.8.c) divides the evaluated $T_A$, $T_B$ and $T_0$ in the seven sub-intervals of $T_{SW}$ as reported in Fig.1.b and c. To perform it, a sub-block $DIV$ makes the following operations:

$$T' = \begin{cases} \frac{T}{2} & \text{if } T \text{ is even} \\ \frac{T-1}{2} & \text{otherwise} \end{cases}$$

$$T'' = \begin{cases} \frac{T}{2} & \text{if } T \text{ is even} \\ \frac{T+1}{2} & \text{otherwise} \end{cases}$$

where the check of the LSB of the input $T$ defines whether it is even, i.e. $LSB = 0$, or odd, i.e. $LSB = 1$, and its functional block diagram is reported in Fig.8.d. It is worth to note that the operation is a single left bit shifting and requires only a small amount of resources.

C. SELECTION OF THE INVERTER CONFIGURATION

The last sub-system consists of the $SECTOR SELECT$ and $DEAD TIME$ blocks, of a multiplexer, and of a crossover switch and its functional block diagram is reported in Fig.9. The $SECTOR SELECT$ block supplies the four 3-bit string, i.e. $C_0$, $C_7$, $C_A$, and $C_B$, to the multiplexer used to define the inverter configuration during a dwell-time. These four signals depend on the wanted sector, which is selected through the $ACTUAL SECTOR$ input signal. Then, as reported in Tab. 2 the $MUX$ signal from the $COUNTER$ block selects the right inverter configuration.

Before sending the three signals to the $DEAD TIME$ block, two of the three signals go through the crossover switch, that is controlled by $L/R$ input signal. In this way, the gate signals between the leg $B$, i.e. $P_{B+}$ and $P_{B-}$, and the leg $C$, i.e. $P_{C+}$ and $P_{C-}$, of the power converter of Fig.1.a are switched one to each other introducing a phase shift among the relative phase voltages, i.e. $v_B$ and $v_C$: it means inducing a rotating magnetic field in clockwise or counterclockwise directions.

The $DEAD TIME$ block generates the six signals for all the power transistors of the inverter including the dead-time. A counter performs the delay of integer multiple of $1/f_{CLK}$ with a minimum value of $1/f_{CLK}$ and a maximum of $(2^{10} - 1)/f_{CLK}$. It is worth to note that such block has been introduced in our HW architecture for completeness being fundamental for the correct operation of the power converter. It can be removed if analogue dead-time circuit is used or substituted by another implementation of the $DEAD TIME$ block. However, in the next Section the estimated resources for our architecture also consider the $DEAD TIME$ block.

D. FLOW CHART OF THE PROPOSAL

The flow chart of the proposed architecture is reported in Fig.10 and describes the main steps and conditions of the $SV PWM$ block. The elaboration blocks of Fig.10 have the same colours of those of Fig.5 in order to correlate them. Taking into account also the flow chart of the $POINTER$
block of Fig. 6, we can extract the complexity of the algorithm considering the McCabe’s cyclomatic number and nesting complexity metric, which are, respectively, 10 and 12 following the procedure reported in [31].

V. FPGA IMPLEMENTATION

The prototype is a three-phase DC/AC power converter, used to drive a 750 W – 380 VAC – 50 Hz three-phase asynchronous motor without neutral point, and is shown in Fig. 11 together with the motor and the point machine which are used in the railway network. The DC-input voltage is $V_{DC} = 600$ V with an input capacitor of 250 $\mu$F, whereas the power transistors are STMicroelectronics IGBT STGW40M120DF3 that are driven by a maximum $f_{SW}$ of 10 kHz. The digital controller is an Altera CycloneV 5CEBA4F17C17 [32] and it is programmed with $f_{Cn} = 10 \text{ Hz}$, $f_{SWn} = 100 \text{ kHz}$, $m_a = 1.06$, and $f_{CLK} = 50 \text{ MHz}$, that are equal to those of Section III-C.

In Fig. 12.a the waveforms of the gate drive signals for the top transistors show the sequence of the inverter configuration in $S_i = 3$ at $f_{SW} = 10 \text{ kHz}$, being $n_{SW} = 10$, instead in Fig. 12.b the top and bottom gate driver signals are compared in a single switching period in order to show the dead-time of 500 ns as expected.

In Fig. 13.a and b the phase and the line-to-line output voltages are reported, respectively, by using $n_C = 5$, $n_{SW} = 10$ and $m = 69$, which gives a peak of the line-to-line voltage of:

$$V_{AB} = m_a \frac{V_{DC}}{2} \frac{m}{100} \sqrt{3} = 380 \text{V}$$

Moreover, the third-harmonic injection of the phase voltages
is also evident, which is typical in the SVPWM modulation [33], as well as the good phase displacement of $2\pi/3$ of the three voltages. The possibility to change in real-time the amplitude of the output voltage permits to perform several control, like the ramp up of voltage at the start-up to mitigate the overcurrent into the motor: as shown in Fig. 14, the line-to-line voltage and phase current with a start-up ramp of 1.7 s are performed and overcurrent is avoided. It is worth to note that the time of the ramp and the increasing step of $m$ have been imposed in real-time from the external TLC interface, although they can be generated by FPGA. Moreover, the experimental measurements are obtained using the point machine of Fig.11 as load of the power converter.

In Fig.15 we show the results of our system in several configurations of the signals input and, in particular, we vary $n_C$ from 5 to 2 and then to 6 (see Fig.15.a) and the values of $m$ and $n_C$ with a constant ratio of 7, i.e. $m = \text{const} \cdot n_C$ (see Fig.15.b), which can be used as $V/f$ control, and the value of $\theta$ to $2\pi/3$ and 0 (see Fig.15.c). In all the cases the change of the system configuration is sudden and distortions of the waveforms are not evident. Moreover, the changes are performed in real-time with the external TLC interface, although they can be generated by FPGA.

In Fig. 16 the comparison of the Fast Fourier Transformation of a gate drive signal is for $n_C = 9$ and $n_{SW} = 10$ (see Fig. 16.a) and for $n_C = 3$ and $n_{SW} = 6$ (see Fig. 16.b), whereas in the inset of Fig.16.a, the measured $f_{SW}$ as function of $n_{SW}$ are compared with the theoretical ones of (10b).

In Tab.3 we report the comparison based on the required

![Figure 13](image1.png)

**FIGURE 13.** Experimental waveforms of a) phase and b) line-to-line output three-phase voltages. Set-up is $f_{CN} = 10$ Hz, $f_{SWn} = 100$ kHz, $m_a = 1.06$, $f_{CLK} = 50$ MHz, $m_{MAX} = 69$, $n_C = 5$, $n_{SW} = 10$, $V_{DC} = 600$ V. Phase output voltages are between output phase potential and negative DC input.

![Figure 14](image2.png)

**FIGURE 14.** Experimental waveforms of line-to-line voltage and current of leg A during the start-up $T_{RAMP} = 1.7s$. Set-up is $f_{CN} = 10$ Hz, $f_{SWn} = 100$ kHz, $m_a = 1.06$, $f_{CLK} = 50$ MHz, $m_{MAX} = 69$, $n_C = 5$, $n_{SW} = 10$, $V_{DC} = 600$ V. Current probe: $100$ mV/A as function of $n_{SW}$ are compared with the theoretical ones of (10b).

![Figure 15](image3.png)

**FIGURE 15.** Experimental line-to-line voltages from gate signals for different changes a) of $n_C$, b) of $m$ and $n_C$ at constant ratio and c) of $\theta$. Set-up is $f_{CN} = 10$ Hz, $f_{SWn} = 100$ kHz, $m_a = 1.06$, $f_{CLK} = 50$ MHz, $m = 69$, $n_{SW} = 10$, $V_{DC} = 600$ V.
FPGA resources between our HW architecture and the state-of-art. A general FPGA architecture is based on three modules: Input/Output blocks, Interconnection Wires and Configurable Logic Blocks, CLB. These last are used to implement user logic and are disposed in a two dimensional matrix where the user can also design their interconnection. CLBs contain LUT, FF and multiplexers and mainly define the resource of the device. Hence, our metric of comparison is expressly focused on, as follows: [38]:

- **Look-Up-Table (LUT)** is generally a n-bit combinational structure and stores the truth table of an user-defined function. Moreover, it can be used as a memory block where the stored value of the single cell is given as output for each value of the input, namely index;
- **Flip Flop (FF)** is the smallest memory unit and able to store one bit either at the falling or at the rising edge of the clock signal;

Moreover, we consider the below metrics for a complete comparison:

- **Block Random Access Memory (BRAM)** is used to store data into a FPGA and are preferred to LUT when a large amount of data needs;
- **Digital Signal Processor (DSP)** is an optimized processor dedicated to the signal processing;
- **Dynamic Power (Dyn.Pow.)** is the power consumption related to the dynamic component of the overall dissipation and is evaluated through Quartus II [39] for [32] and Vivado Design Suite [40] for [34].

From the comparison with the state-of-art of Tab.3 we can state that our architecture avoids the use of the external signal references to generate the three-phase voltages before the Clarke transformation [35]- [36], of less dynamic power [35], of other resources, like DSP [37] or BRAM [17]. For such differences, our purpose is a good candidate for stand alone systems, in which the external controllers are absent or with limited functionalities. Or, for those applications where the diversity of FPGA needs to satisfy the requirements to have different controller suppliers both for the industrial production and for the safety conditions. For example, when a high Safe Integrity Level equal to 4 is needed like in railways applications, a 2oo2 architecture of digital controllers is used to reduce the risk of dangerous failure below a tolerable probability and such functionality is achieved by connecting in parallel two different processors, performing the same tasks, so that the problem of spurious trips, which induce undesired shut-down of the overall system, is reduced.

In order to show the generality of our HW architecture, we also implemented it in a low performance FPGA like Xilinx Artix 7 [34] whose 4.75% and 0.75% of LUT and FF are used. Indeed, although our system needs more resources than other like [25] even to the +2.35 and +1.27 times the required LUT and FF, respectively, or like [17] even to the +1.21 and +1.66 times the required LUT and FF, respectively, the possibility of setting several configurations of the output three-phase voltages in real-time justifies such increase. Moreover, the executive time results equal to three system clock periods and it is due to the presence of FFs between the combinational blocks of the system for the data consistency. It is worth to note that the maximum clock frequency of the system is 100 MHz, this data is been calculated using the timing analysis of [40].

Finally, the accuracy of the architecture refers on the capacity to describe the trajectory and the amplitude of $V_{REF}$ in the $\alpha - \beta$ plane of Fig.2 and is reflected on the values of the dwell-times. The trajectory of $V_{REF}$ is imposed by the incremental step of the angle, i.e. $\Delta \theta$, and is equal to $2\pi f_C / f_{SW}$. It means that it depends on the power converter, i.e. $f_{SW}$, and on the applications, i.e. $f_C$. Hence, if the conditions reported in Section III on how to size LUT and its cells, are respected, the LUT construction reflects the accuracy imposed by $\Delta \theta$. About the output amplitude accuracy, it depends on how much we can change $m_a$, whose maximum value is defined during the LUT construction implementation and corresponds to the maximum values achievable by the power converter in our experimental case. In order to change

| TABLE 3. Synthesis of the Proposed Architecture and its Comparison with the State-of-Art |
|---------------------------------------------------------------|
| **LUT** | **BRAM** | **FF** | **DSP** | **Dyn.Pow.** | **Config.** |
|----------------|----------|-------|---------|-------------|------------|
| This work      | [32]     | 1087  | 0       | 372         | 0          | 1.30       | int.      |
| Cyclone II     | [35]     | 987   | 0       | 312         | 0          | 0.51       | int.      |
| Cyclone IV     | [36]     | 780   | -       | 88          | -          | 28         | ext.      |
| Spartan III    | [37]     | 903   | -       | 18          | -          | -          | ext.      |
| Artix 7        | [25]     | 161   | -       | 77          | 11         | -          | ext.      |
| Spartan III    | [17]     | 420   | 0       | 245         | 0          | 0.233      | no        |
|                |          | 810   | 3       | 188         | 0          | -          | int.      |
it in real-time, we introduced the variable \( m \) varying the amplitude of \( V_{\text{ref}} \) with a precision of 1% of the maximum \( m_o \) value.

### A. COMPARISON WITH CORDIC ARCHITECTURES

CORDIC algorithm is generally used to evaluate trigonometric functions as for example those in (4) and in this subsection we compare our HW architecture with CORDIC ones. In particular, we replaced the blocks \texttt{POINTER}, \texttt{LUT}, and \texttt{DENORMALIZATION} of Fig.5 with CORDIC-based hardware: those three blocks are used to evaluate the dwell-times, instead the others are to manage the switching sequences of power converter. In this way, we can better compare the HW architectures. To design CORDIC architecture, we need to define the number of iterations of the algorithm: it depends on the minimum angle \( \theta \) and, hence, on the incremental step \( \Delta \phi_o \). In our application it is \( 2\pi 10^{-4} \) and imposes 11 bits for the decimal part, that means 11 iterations for the required convergence. Then, we can define two kinds of architectures, namely \texttt{CORDIC cascade} and \texttt{CORDIC iterative}. Both of them are based on the same evaluation layer, but they differ on how to achieve the results of the trigonometric functions: \texttt{CORDIC cascade} is a cascade of 11 single layers and the calculation is done in one system clock; instead \texttt{CORDIC iterative} is a single layer that is iteratively invoked for 11 times, which clearly requires approximately 11 system clock periods. In Tab.4 the results of the synthesis implemented in Xilinx Artix 7 [34] are reported.

Although our HW architecture requires a LUT memory to store the data, our optimization procedure keeps the required resources reduced, at the most equal to those of \texttt{CORDIC iterative}. However, the latter needs a more complex architecture to work, due to the 11 clock period lag between the arrival of the new angle and the correct output. Moreover, the 11 clock period delay may limit the switching frequency whenever \( f_{\text{CLK}} \) is close to \( f_{\text{SW}} \), that is possible when low cost digital controllers are used for high switching converters, i.e. around MHz. On the other side, \texttt{CORDIC cascade} needs more resources and has a higher dynamic power dissipation, making it the worst choice.

### VI. CONCLUSION

The proposed HW architecture permits to evaluate the dwell-times of the gate driver signals used in a SVPWM without the requirement of external reference three-phase signals and with the possibility to change in real-time the values of \( f_C \), \( f_{SW} \), \( \theta \) and \( V_M \). Indeed, they are the input signals of our system and can be both internal to the digital controller and external from another controller in order to perform more complex control techniques. It has been possible because trigonometric functions are avoided and an excellent management of the required information to reconstruct the dwell-times has been proposed. Moreover, the implementation in a FPGA digital controller has shown its compactness in terms of needed resources, which is useful in stand alone system as well as for low cost controller. Such benefits have been also highlighted from a comparison with the well-known CORDIC algorithm. When this architecture is used in a closed-loop system, our block can be seen as a sub-block, which calculates the dwell times and generates the drive signals of the power transistors, and the controller has to define its inputs, namely \texttt{DATA IN}. It is worth to note that, for the case of the \( d–q \) representation, \( m \) and \( \theta \) signals of \texttt{DATA IN} are, respectively, evaluated through the modulus and the phase of \( V_d \) and \( V_q \), performed by the controller.

Finally, our HW architecture can be also designed for microcontrollers and results useful in all those applications where open-loop or electrical closed-loop control are required due to the lacks of information about the motor motion.

### REFERENCES

[1] L. Di Benedetto, G. D.Licciardo, A. Rubino, E. Lanzotti, and E. Piccirilli, “Analysis of the performances of a fully 4h-sic insulated dc/ac converters,” in 2017 IEEE International Conference on Environment and Electrical Engineering and 2017 IEEE Industrial and Commercial Power Systems Europe (IEEEIC/EPS Europe), pp. 1–4. IEEE, 2017.

[2] G. E. M. Ruiz, N. Muñoz, and J. B. Cano, “Design methodologies and programmable devices used in power electronic converters—a survey,” in 2015 IEEE Workshop on Power Electronics and Power Quality Applications (PEPQA), pp. 1–6. IEEE, 2015.

[3] O. Pop, G. Chindris, and A. Dulf, “Using dsp technology for true sine pwm generators for power inverters,” in 27th International Spring Seminar on Electronics Technology: Meeting the Challenges of Electronics Technology Progress, 2004., vol. 1, pp. 141–146. IEEE, 2004.

[4] E. Monmasson, L. Idkhajine, I. Bahri, L. Charaabi et al., “Design methodology and fpga-based controllers for power electronics and drive applications,” in 2010 5th IEEE Conference on Industrial Electronics and Applications, pp. 2328–2338. IEEE, 2010.

[5] M. H. Rashid, Power electronics handbook: devices, circuits and applications. Elsevier, 2010.

[6] H. Guzman-Miranda, L. Sterpone, M. Violante, M. A. Aguirre, and M. Gutierrez-Rizo, “Coping with the obsolescence of safety- or mission-critical embedded systems using fpgas,” IEEE Transactions on Industrial Electronics, vol. 58, no. 3, pp. 814–821, 2010.

[7] L. Di Benedetto, G. Licciardo, T. Erlenbach, A. Bauer, and A. Rubino, “Optimized design for 4h-sic power monofet,” IEEE Electron Device Letters, vol. 37, no. 11, pp. 1454–1457, 2016.

[8] Y. Li, Y. Zhang, X. Yuan, L. Zhang, F. Ye, Z. Li, Y. Li, Y. Xu, and Z. Wang, "A 500kw forced-air-cooled silicon carbide (sic) 3-phase dc/ac converter with a power density of 1.246 mw/m3 and efficiency> 98.5%," in 2020 IEEE Energy Conversion Congress and Exposition (ECCE), pp. 209–216. IEEE, 2020.

[9] L. Di Benedetto, G. D. Licciardo, T. Erlenbach, A. J. Bauer, R. Liguori, and A. Rubino, “A model of electric field distribution in gate oxide and
L. Di Benedetto et al.: Implementation of Hardware Architecture for SVPWM with Arbitrary Parameters

MATLAB, version 9.9.0 (R2020b). Natick, Massachusetts: The Math-Works Inc., 2020.

N. Mohan, T. M. Undeland, and W. P. Robbins, Power Electronics. Converters, Applications and Design, 3rd ed. John Wiley and Sons, Inc, 2003.

E. Y. Li, “A measure of program nesting complexity,” in National Computer Conference, San Luis Obispo, California, pp. 531–538, 1987.

Cyclon V, Intel, 2019, available from: https://www.intel.com/content/www/us/en/programmable/documentation/mcn/122497163812.html

S. R. Bowes and Y-S. Lai, “The relationship between space-vector modulation and regular-sampled pwm,” IEEE Transactions on Industrial Electronics, vol. 44, no. 5, pp. 670–679, 1997.

Artix 7, Xilinx inc., 2018, rev. 2018.02.27. Available from: https://www.xilinx.com/products/silicon-devices/fpga/artix-7.html

A. Belkheiri, S. Aoughellanet, and M. Belkheiri, “Fpga implementation of a space vector pulse width modulation technique for a two-level inverter,” Elektrotechnik Vestnik, vol. 85, no. 3, pp. 73–88, 2013.

B. Rashidi and M. Sabahi, “High performance fpga based digital space vector pvm three phase voltage source inverter,” International Journal of Modern Education and Computer Science, vol. 5, no. 1, pp. 62, 2013.

S. Boukaka, H. Chaoui, and P. Sicard, “Fpga implementation of svpwm,” in 2013 IEEE 11th International New Circuits and Systems Conference (NEWCAS), http://dx.doi.org/10.1109/NEWCAS.2013.6573656DOI 10.1109/NEWCAS.2013.6573656, pp. 1–4, 2013.

U. Farooq, Z. Marrakchi, and H. Mehrez, Tree-based Heterogeneous FPGA Architectures: Application Specific Exploration and Optimization, pp. 7–48, Springer New York, 2012.

Quartus II Handbooks Volume 1: Design and Synthesis, Intel, 2014, qisv1 (2015.05.04). Available from https://www.intel.com/content/dam/altera-www/global/en_US/pdfs/literature/hb/qts/archives/quartusii_handbook_archive-15.0.pdf

Vivado Design Suite Properties Reference Guide, Xilinx inc., 2018, u9G12 (2018.2). Available from: https://www.xilinx.com/support.html/documentation.
GIAN DOMENICO LICCIARDO (S’03-M’06-SM’17) received the Electronic Engineering degree from the University of Naples Federico II in 2002 and the Ph.D. degree in information engineering from the University of Salerno, Italy, in 2006. Since 2007, he has been an Assistant Professor of Electronics with the Department of Industrial Engineering, University of Salerno. His main research interests include the modeling, simulation, and characterization of electron devices and the design of VLSI systems.

ROSALBA LIGUORI (M’17) received the B. Sc., M.Sc. and Ph.D. degrees in electronic engineering from the University of Salerno, Fisciano, Italy, in 2006, 2010 and 2014, respectively. Since 2014, she has been a Research Fellow with the Department of Industrial Engineering, University of Salerno. Her current research interests include modeling, simulation, and development of electronic devices based on novel semiconductors and design of very large-scale integration systems.

EDUARDO PICCIRILLI received the B.Sc. and the M.Sc. degrees (cum laude) in electronic engineering from the University of Salerno, Fisciano, Italy, in 2007 and 2010, respectively. In 2011 he joined Hitachi Rails STS and his main activities are aimed in the designing of communication and signalling control systems with inherent failsafe and reactive fail-safe architectures, railways condition monitoring, power and energy systems, vital interlocking data acquisition and analysis and systems engineering.

EMILIO LANZOTTI received the B.Sc. and the M.Sc. degree (cum laude) in telecommunication engineering from the University of Sannio, Benevento, Italy, in 2004 and 2007, respectively. After a short experience in Ericsson, in 2009 he joined Hitachi Rail STS with the main activities on IXL Wayside platform development. In particular, he is a product line engineering of Wayside platform regarding the peripheral post Technology.

ALFREDO RUBINO (M’17) received the Laurea degree in physics from the University of Naples Federico II, Naples, Italy, in 1988. In 1989, he was with the Italian National Agency for New Technologies, Energy and the Environment, Portici, Italy. Since 2005, he has been an Associate Professor of Electronics with the University of Salerno, Salerno, Italy. His current research interests include electronics technology and organic electronics.

***