

## Article

# Student Project-Based Space Vector Modulation Technique for Power Electronics Laboratory

Lutfu Saribulut <sup>1</sup> and Arman Ameen <sup>2,\*</sup><sup>1</sup> Department of Electrical-Electronics Engineering, Adana Alparslan Turkes Science and Technology University, Adana 01250, Turkey<sup>2</sup> Department of Building Engineering, Energy Systems and Sustainability Science, University of Gävle, 801 76 Gävle, Sweden

\* Correspondence: arman.ameen@hig.se

**Abstract:** Two-level DC/AC inverter topologies are widely used for low voltage and high voltage applications in power systems and industrial areas. Space Vector Modulation (SVM) is a popular Pulse-Width Modulation technique used for controlling the inverters and providing the efficient energy conversion from DC sources. However, applications of SVM-based studies are limited in the Power Electronics Laboratory (PEL) due to the vital risks associated with high voltage applications, and it is not easily learned through mathematical analysis and visual learning without implementation by undergraduate students. A simulation and experimental setup of an SVM-controlled two-level, three-phase inverter was presented in this study for undergraduate students to learn its basics in the PEL. Several programs were used to simulate the inverter in the classroom environment and to design a power circuit and microcontroller-based printed circuit board of the inverter for PEL experiments. The two case studies were given. In the case results, the output voltage waveforms of simulation and experimental inverters were compared to show the validation of simulation results. With this study, the students' experience is enhanced in electronic circuit design, programming, coordination with hardware and software development activities, self-learning, and teamwork. Additionally, practical applications increase undergraduate students' interest in Power Electronics Courses and reinforce their knowledge from lecture and laboratory studies.



**Citation:** Saribulut, L.; Ameen, A. Student Project-Based Space Vector Modulation Technique for Power Electronics Laboratory. *Electronics* **2023**, *12*, 2714. <https://doi.org/10.3390/electronics12122714>

Academic Editor:  
Ahmed Abu-Siada

Received: 10 May 2023  
Revised: 12 June 2023  
Accepted: 15 June 2023  
Published: 17 June 2023



**Copyright:** © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

**Keywords:** power electronics laboratory; two-level inverters; space vector modulation technique; microcontroller; USB application; PSCAD

## 1. Introduction

The increasing demand for electrical power in industrial and residential areas necessitates more efficient utilization of existing transmission and distribution systems by incorporating new power electronic devices. These devices have a wide range of applications, including DC-DC/DC-AC power conversion, industrial motor drives, and grid connection of renewable energy resources. The field of power electronics is multidisciplinary, encompassing various subjects such as circuit theory, signal analysis, electronics, control systems, and semiconductor devices [1]. Therefore, there is a need for skilled professionals who can undertake roles such as researchers, resource planners, engineers, and technicians in this area [2]. Consequently, it has become increasingly important for undergraduate students to learn the concepts of power electronic devices in the lectures of the “Power Electronics Course (PEC)” and “Power Electronics Laboratory (PEL)”.

During the lectures of the PEC, undergraduate students are introduced to theoretical concepts, analysis techniques, and the importance of experimental learning with the support of this course. In addition to providing theoretical knowledge, the PEL, which usually accompanies the PEC, allows students to gain hands-on experience through the use of industrial-grade software/hardware tools [3]. It is important to note that an effective

power electronic curriculum should provide engineers with hands-on experience [1]. Due to the high costs and safety issues of power system applications, several companies have developed virtual laboratories that simulate these applications in a computer environment. MATLAB and Power System Computer-Aided Design (PSCAD) are the most preferred virtual laboratories used in PECs and PELs [4,5].

Although simulation and virtual laboratories offer valuable supplements to formal lectures, they do not emphasize hardware implementation and physical aspects of systems [1]. It can be said that the PEL is fundamental in aiding undergraduate students' understanding of the PEC. This trend has resulted in an increased interest in the PEC at the senior undergraduate and graduate levels in various institutions [6]. Generally, the PEC covers topics such as DC-DC converters, DC-AC inverters, Pulse-Width Modulation (PWM) techniques, and motor drivers. To improve the controllability of laboratory equipment and power electronic devices such as inverters, converters, and drivers, microcontrollers and Digital Signal Processors (DSPs) are used with appropriate software in the PEL. Microcontrollers and DSPs can be connected to a PC via a Universal Serial Bus (USB), and a series of commands can be sent to these devices using any terminal emulator program. During the connection of these devices to a PC, students can change the frequency of clock signals, read or write to any RAM module and peripherals, configure a debugging module, and monitor universal bus states [3,7].

The need to support theory with practice in power electronic education, particularly in control education, was supported by numerous papers listed in [8]. An educational PEL primarily developed to reinforce students' understanding of fundamental concepts experimentally was presented in [1]. This laboratory combined theoretical design, simulation studies, digital control, fabrication, and verification of power electronic circuits using a set of hardware and software tools. A 2 kVA prototype of a power electronic and drives experimental bench was developed and tested for various modes of operation [2]. This kit included reconfigurable hardware modules that could be interconnected to achieve over 14 different circuit topologies. A pedagogical approach to teach behavioral modeling of switch-mode electronic systems was presented using electronic circuit simulators [3].

The methodology was geared towards electrical engineering students enrolled in the PEC, Industrial Electronic Course, or similar programs at the undergraduate level. Switched-Mode Power Supplies (SMPS) were modeled using MATLAB programming [6]. The proposed practical SMPS was then designed in the laboratory environment. A low-cost experiment for a control systems laboratory module was presented [8]. The experiments were organized around the microcontroller-based control of a permanent magnet DC motor. A web server interface design was proposed for electrical machines, with the illustration of excited DC motor and generator experiments presented separately [9]. Lab VIEW v. 7.10 software supported tools were used to create web-based automation and control of the experiment hardware.

An example of applying project-based lab teaching in the PEC was presented [10]. In this project, students used a microcontroller and an H-bridge inverter to design a control system for real practical applications, such as washing machines, trams, or electric lifts. An implementation of the SVM algorithm using a microcontroller was presented for its educational benefits. It was focused on the experimental survey of the inverter's parameters, analyzing and proposing improved solutions for the inverter to operate optimally. The SVM algorithm was tested on an Arduino Mega 2560 board and small power inverter model [11]. An educational tool was proposed for graduate students who take the "Vector Control of AC Motors course" [12]. The theoretical backgrounds of electrical motors, space vector control methods, speed acquisition, and control algorithms were explained. At the conclusion, case results and how to interpret these results were given.

The design and development of a remote interactive laboratory was proposed for controlling DC machines at the Technical and Vocational University in Mohajer Faculty [13]. An Arduino board was used to make a data acquisition card and actuator controller. Furthermore, LabVIEW was used for the remote control of the experimental setup using

the internet. This experimental board also allowed the students to observe and perform the experiments of the DC machines course anywhere and anytime.

The ability to apply knowledge in real-world situations is one of the most necessary skills for the engineers. Practical Experimental Labs (PELs) offer undergraduate students the opportunity to work on the development of experimental prototypes in software or hardware, providing training in the development of real products and allowing students to confront the problems they will face in their future careers [14]. Learning by doing is the most effective way to ensure permanent learning for students in PELs during their education in the PECs. PECs and PELs complement each other as theoretical and practical aspects, much like the complementary angles in geometry. Therefore, undergraduate students in Electrical/Electrical–Electronics Engineering should take both PECs and PELs during their educational periods.

Undergraduate students in the PECs and PELs often have difficulty understanding the operating principles of DC-AC inverters, such as the two-level DC/AC, three-phase inverter, as well as the types of inverter controllers such as the Space Vector Modulation (SVM) technique, without implementation in a laboratory environment. To simplify the understanding of these inverters and their controllers, a simulation model of SVM for PECs and an experimental procedure of the inverter for PELs were presented in this paper. These models provided step-by-step processes for the students to follow, allowing them to learn the theory behind the SVM technique, simulate the inverter, design the experimental card, control the experimental board, and verify simulation and experimental board results. Additionally, the programming codes of both the microcontroller and SVM controller components, created in PSCAD, were shared to facilitate faster learning of the SVM technique by undergraduate students.

This paper is organized as follows: Section 2 discusses the pedagogical outline of PECs as implemented in the Departments of Electrical/Electrical–Electronics Engineering in Turkey. Section 3 presents the theory of the SVM technique. Section 4 describes the step-by-step design and application processes of the SVM technique, including the simulation and experimental procedures. Section 5 reports the evaluation of the case studies by comparing the simulation and experimental results. Section 6 discusses the learning objectives and measurable outcomes of educational material. Section 7 discusses the educational benefits of this study. Finally, Section 8 concludes the paper.

## 2. Overview and Objectives of PEC and PEL

Power electronics can be defined as the conversion of electrical power from one type to another using power semiconductor devices as switches. In the Electrical/Electronic Engineering programs of universities, the PEC is an elective course for final-year undergraduate students. The PEC comprises a weekly three-hour lecture over 14 weeks of one semester and is supported by an alternating two-hour PEL [10]. The main goal of the PEC is to teach the theoretical operation of switching converter topologies. The main goal of the PEL is to design these topologies as circuits using commercially available PWM methods and gain practical experience by carrying out hands-on experiments [6]. In the theoretical lectures of the PEC, students receive background knowledge on electronic power converters used to convert voltage and current from one form to another using power electronic converters. These converters can be classified as:

- Rectifiers that are converting an AC voltage to a DC voltage;
- Inverters that are converting a DC voltage to an AC voltage with controlled frequency;
- Switch-Mode Power Supplies that are converting a DC voltage to another DC voltage;
- Cyclo-converters that are converting an AC voltage to another AC voltage with controlled frequency;
- AC/AC voltage controllers that are converting an AC voltage to another AC voltage with the same frequency.

During the theoretical education of the PEC, students can acquire a large amount of knowledge about electronic power converters. Then, in the practical lectures of PELs, they

are offered the opportunity to gain hands-on experience with power electronic circuits, both in software and hardware. The education in PELs is carried out in three stages. The first stage involves simulating an experimental project as a preliminary study, using MATLAB or PSCAD programs. In this stage, students gain practice in simulation programs and learn how to relate mathematical models to circuit simulation. The second stage involves applying the preliminary study in laboratory environments, where students are introduced to the fundamental knowledge of microcontrollers used for controlling power converters. The third stage involves validating the simulation results with the experimental results and discussing these results with the lecturer and other students.

After the final lectures of the PEC and PEL, undergraduate students are expected to further their knowledge on the switching techniques of power electronic converter circuits, the design and control of the system, and efficiency analysis. These gains can also be relevant for engineers who want to improve their skills in designing electronic power converters for transmission and distribution networks, among other applications. The experience gained by the students serves as preparation for bigger challenges in the area of power electronics, especially in the context of renewable energy sources and their importance in achieving sustainable energy worldwide [15].

### 3. Theory of the Space Vector Modulation Technique

Two-level, three-phase DC/AC inverters are widely used for AC motor drives serving as actuators with variable input voltages (controllable for amplitude and frequency). In other words, DC/AC inverters are used for converting a DC power source into AC power applications [16]. There are many methods for controlling the inverters in the literature. The first group which is common among of them includes the PWM, sinusoidal PWM, and multiphase PWM methods. In these methods, the control signals of switches used in the power circuit of inverters are obtained by comparing the AC voltage waveform at the desired frequency and phase with a carrier signal at high frequencies in the triangular waveform. The waveform of inverter output voltage looks like a pulse train. Each pulse has the same amplitude and a different duty cycle which corresponds to the input signal at the sampling instants [3,4].

The second group of most common methods includes 120°, 150°, and 180° conduction modes, respectively. In this group, the carrier signal is not used to obtain the control signals of switches. In contrast, they were defined in a known sequence [17]. In these modes, each switch of the inverter conducts for a specified time period. Hence, the output voltage of the inverter is designed to resemble a sinusoidal waveform. For instance, in the 180° conduction mode, each couple of switches conducts for a 180° time period [18]. The open and closed operations of other couple switches are delayed from the previous couple switching signals by a 180° time period. Hence, all couple switches are opened and closed periodically in the proper sequence to produce the desired output waveform. This mode is generally used in commercial low power inverters [19].

SVM is one of the most popular PWM techniques used in inverter technologies. Among all the switching algorithms proposed in the literature for inverters, SVM appears to be the most promising technique because it offers great flexibility in optimizing switching pattern design, and it is well suited for digital implementation [20]. With the development of microprocessors and DSPs, SVM has become the most widely used PWM technique for three-phase inverters. It uses the space-vector concept to compute the duty cycle of the switches. The notable features of SVM are its aptitude for easy digital implementation, wide linear modulation range for line-to-line output voltages, and suitability for the optimization of switching sequences to improve fast algorithms [20]. Additionally, SVM directly uses the control variable provided by the control system and quickly identifies each switching vector as a point in the complex ( $\alpha$ - $\beta$ ) domain.

The support for the SVM technique in education is evidenced by many previous studies. The two different modulation strategies based on SVM techniques were experimentally verified by utilizing three-phase Neutral Point Clamped (NPC) inverters [21]. A

multilevel converter topology consisting of two insulated DC supplies and a dual two-level inverter was proposed, and the modulation strategy was presented to regulate power sharing between DC sources [22]. The two switching algorithms were proposed to implement SVM-controlled dual-inverter schemes for both simulation and experimental case studies [23]. The proposed algorithms employed the instantaneous phase voltages for the implementation of the SVM technique.

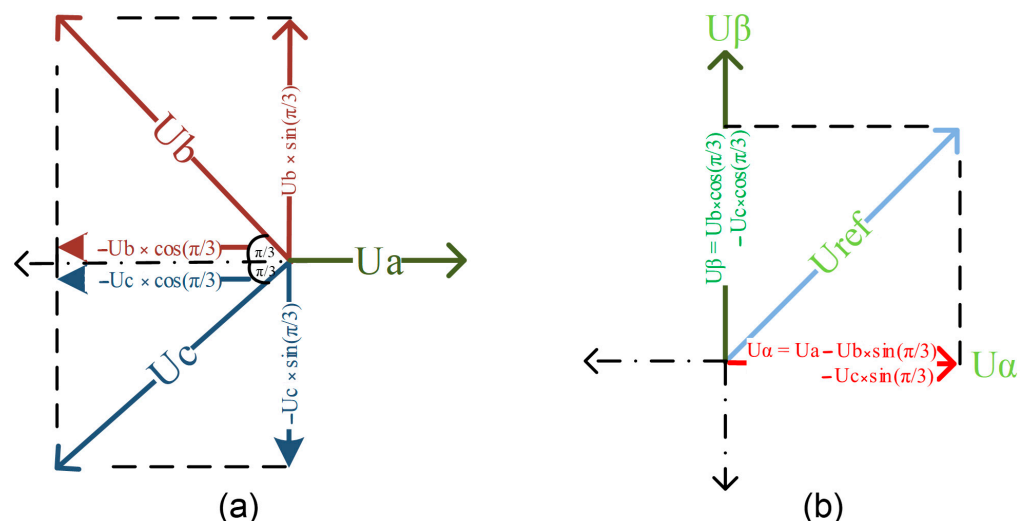
An improvement was described for an existing switching algorithm for the implementation of synchronized SVM for a three-phase, two-level inverter to drive an induction motor with V/f control [24]. The proposed algorithm was experimentally implemented on a DSP platform. A signal generation of SVM for a conventional three-phase inverter was presented by comparing the phase voltages with references, and the switching signal generation and the implementation of SVM were explained using a PIC microcontroller [25]. Finally, the methodology for using SVM in two-level unidirectional PWM rectifiers was presented and applied to three different groups of rectifiers [26]. The proposed modulation technique was validated by simulation and experimental results using a 20 KW prototype.

Any three functions of time, which are always spatially separated by  $120^\circ$ , and their sum that is constantly equal to zero, can be expressed with space vectors. The angle of these vectors increase due to the changing of time. It causes the rotation of each vector with the frequency of those quantities [27]. A three-phase stationary reference frame defined by  $U_a(t)$ ,  $U_b(t)$ , and  $U_c(t)$  can be represented in the following form.

$$\begin{aligned} U_a &= U_m \sin(\omega t) \\ U_b &= U_m \sin(\omega t + 2\pi/3) \\ U_c &= U_m \sin(\omega t + 4\pi/3) \end{aligned} \quad (1)$$

where  $U_m$  is the maximum value of the phase voltage,  $\omega$  is the angular velocity of the phase voltage [27].

In the stationary reference frame, only currents and voltages of two phases are linearly independent, while the quantities of other phases can be expressed as a linear combination of those [28]. By using this property, the voltages of the stationary reference frame can be mapped to the phase voltages of the complex orthogonal ( $\alpha$ - $\beta$ ) frame. It is graphically illustrated in Figure 1.



**Figure 1.** (a) Decomposition of stationary reference frame phase voltages into their  $xy$ -projections; (b) summation of  $xy$ -projections in complex orthogonal frame.

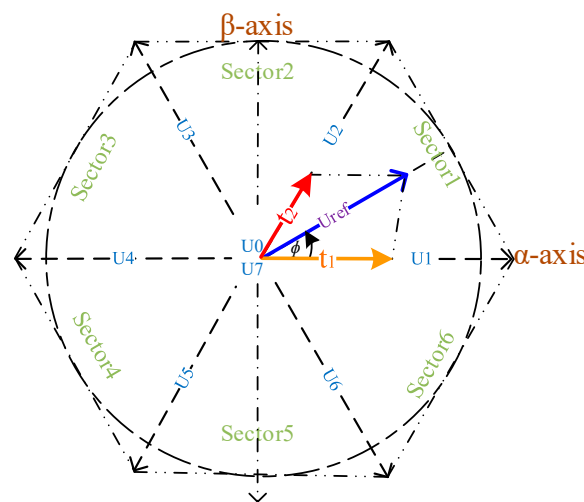


After the decomposition of quantities in Equation (1), they can be redefined into two independent variables ( $U_\alpha$  and  $U_\beta$ ) in the complex orthogonal frame. It is known as the Clark's transformation or  $\alpha$ - $\beta$  transformation. It is formulized as follows [28].

$$\begin{bmatrix} U_\alpha \\ U_\beta \end{bmatrix} = \begin{bmatrix} 1 & -1/2 & -1/2 \\ 0 & \sqrt{3}/2 & -\sqrt{3}/2 \end{bmatrix} \begin{bmatrix} U_a \\ U_b \\ U_c \end{bmatrix} \quad (2)$$

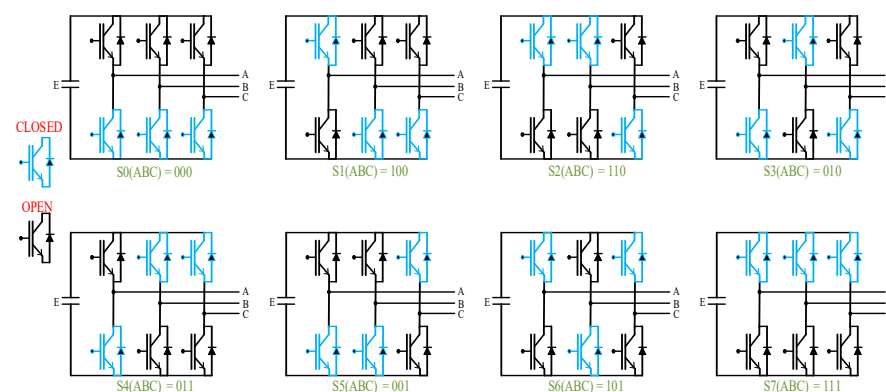
where  $U_\alpha$  is the vector projection of  $U_{ref}$  onto the  $\alpha$ -axis and  $U_\beta$  is the vector projection of  $U_{ref}$  onto the  $\beta$ -axis.

Clark's transformation is generally used for converting the stationary three-phase system to the complex orthogonal system. When applying the transformation to the three-phase system, it is assumed that the a-axis coincides with the  $\alpha$ -axis in the orthogonal  $\alpha$ - $\beta$  coordinate system illustrated in Figure 1. After the derivation of the transformation matrix given in Equation (2), the reference voltage ( $U_{ref}$ ) is represented as a vector in this plane. It is illustrated in Figure 2.



**Figure 2.** Diagram of voltage space vectors for a three-phase, two-level inverter.

The space vector diagram of a three-phase, two-level inverter consists of six sectors (Sector 1 to Sector 6). Each sector has a  $60^\circ$  portion of the space vector diagram. There are also eight switching vectors ( $U_1$  to  $U_7$ ). These vectors and sectors shape a symmetric hexagon.  $U_{ref}$  is used to control the magnitude and frequency of output voltage for the three-phase inverter. It rotates anti-clockwise within the space vector at  $\omega$  velocity. All switching states and sectors of the space vector are given in Figure 2 [29]. These states have also switching positions. They are illustrated with their switching functions in Figure 3.



**Figure 3.** Switching positions of the three-phase, two-level inverter.

The six switching functions (S1(ABC) to S6(ABC)) allow power transfer from the DC-link to the AC side. Hence, they are called the active states. The other two switching functions (S0(ABC) and S7(ABC)) finalize the current circulation at the positive side or at the negative side of the source (E). There is no power flow from the DC-link to the AC side. Hence, they are called the null states [30]. The null vectors are located at the center of the hexagon given in Figure 2.

The eight switching functions should be applied together in one period of reference for producing the desired voltage at the output of the inverter. In real applications, these switching functions are represented as binary combinations of switching functions for the upper switches. As indicated in Figure 3, the switching state '1' for the S1(ABC) function denotes that the upper switch of phase A is closed while the lower switch is open and its output voltage is positive (+E). The switching state '0' for the same function denotes that the upper switch of the same phase is open while the lower switch is closed and its output voltage is positive (−E) [31].

The SVM algorithm is realized by correctly selecting and properly applying the switching functions in each sampling time ( $t_s$ ). Its performance can also be improved by selecting the appropriate switching functions and optimizing the switching sequences [20]. To reduce the number of switching sequences and make full use of the turn-on time of the space vectors,  $U_{ref}$  is commonly split into the two-nearest adjacent vectors and zero vectors in any sector [32]. Depending on its location,  $U_{ref}$  can only be synthesized using switching sequences during each  $t_s$  [33]. For example,  $U_{ref}$  can be expressed as the average of the vectors for any  $t_s$  in sector I using Equation (3):

$$U_{ref}(t) = \left(\frac{t_0}{t_s}\right)U_0 + \left(\frac{t_1}{t_s}\right)U_1 + \left(\frac{t_2}{t_s}\right)U_{21} + \left(\frac{t_7}{t_s}\right)U_7 \quad (3)$$

where  $t_1$  is the switching time of  $U_1$ ,  $t_2$  is the switching time of  $U_2$ , and  $t_0$  and  $t_7$  are the switching times of  $U_0$  and  $U_7$ , respectively.

The switching times of other sectors can be calculated using the equations given in the below:

$$t_1 = \left(\frac{\sqrt{3}}{V_{DC}}\right)U_{ref}t_z \sin\left(z\frac{\pi}{3} - \phi\right) \quad (4)$$

$$t_2 = \left(\frac{\sqrt{3}}{V_{DC}}\right)U_{ref}t_z \sin\left(\phi - \left[(z-1)\frac{\pi}{3}\right]\right) \quad (5)$$

$$t_0 = t_7 = (t_z - t_1 - t_2) \quad (6)$$

where  $\phi$  is the phase angle of  $U_{ref}$ ,  $V_{DC}$  is the capacitor voltage,  $z$  is the sector number, and  $t_z$  is the half of  $t_s$  [31].

For an instance, it is assumed that  $U_{ref}$  is lying in the first sector of the hexagon. Then, the two adjacent active space vectors ( $U_1$  and  $U_2$ ) and zero space vectors ( $U_0$  and  $U_7$ ) are utilized in the construction of  $U_{ref}$  as the average of these vectors using Equation (3) [34]. This is illustrated in Figure 4.

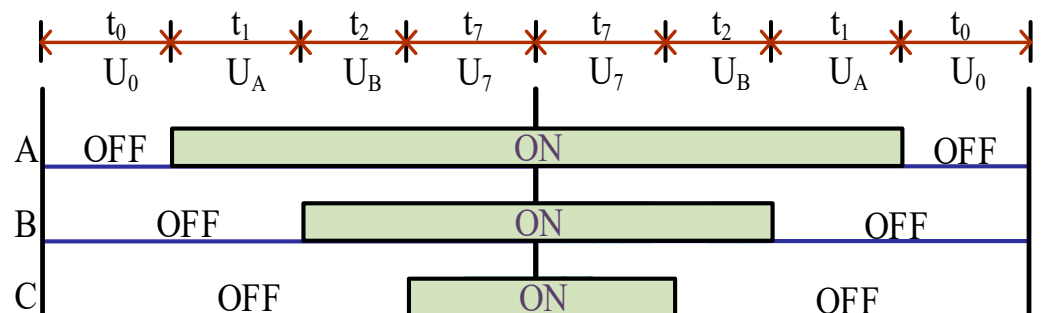


Figure 4. Construction of  $U_{ref}$  as the average of space vectors in each  $t_s$ .

The vectors  $U_A$  and  $U_B$  shown in Figure 4 are the active vectors of any sector in which  $U_{ref}$  lies. For example,  $U_1$  and  $U_2$  vectors are the active vectors of the first sector and are used instead of  $U_A$  and  $U_B$  vectors. Similarly,  $U_3$  and  $U_2$  vectors are the active vectors of the second sector and are used instead of  $U_A$  and  $U_B$  vectors, respectively. This procedure is applied for the other sectors to construct  $U_{ref}$ . It is important to note that the space vectors of each sector should be applied in the correct sequence and at  $t_s$  in order to realize  $U_{ref}$  in the fundamental period. The sequences of the switches for all sectors are given in Table 1 [20].

**Table 1.** Sequences of the switches for the three-phase, two-level inverter.

Sector	Angle Interval	Sequences of switches			
		$t_0$	$t_1$	$t_2$	$t_7$
1	$0 < \phi \leq 60$	S0(ABC)	S1(ABC)	S2(ABC)	S7(ABC)
2	$60 < \phi \leq 120$	S0(ABC)	S3(ABC)	S2(ABC)	S7(ABC)
3	$120 < \phi \leq 180$	S0(ABC)	S3(ABC)	S4(ABC)	S7(ABC)
4	$180 < \phi \leq 240$	S0(ABC)	S5(ABC)	S4(ABC)	S7(ABC)
5	$240 < \phi \leq 300$	S0(ABC)	S5(ABC)	S6(ABC)	S7(ABC)
6	$300 < \phi \leq 360$	S0(ABC)	S1(ABC)	S6(ABC)	S7(ABC)

The simulation procedure and the experimental setup of the SVM technique are presented in detail for the educational purposes in the following sections.

#### 4. Preparation of the Course Materials for SVM Technique

The content of the SVM technique in the PEC is presented to the undergraduate students in two sessions. The first session provides a mathematical explanation of its theory as presented above. The second session focuses on its simulation setup. However, the experimental setup of the SVM technique is given in PELs. These topics are covered in the following subsections.

##### 4.1. Simulation Setup

Firstly, the simulation software used for the PEC should be decided. Generally, MATLAB and PSCAD simulation programs are the most commonly used for engineering education and the professional engineering area. These programs have some advantages and disadvantages according to each other. They are discussed in the following:

- MATLAB includes many components for several engineering areas, but PSCAD is specialized in electrical engineering components.
- MATLAB consists of two parts: simulink and workspace. The simulink is used to simulate any real system for observation and analyzing. The workspace is used to execute the MATLAB codes dynamically. C++ codes can be integrated into MATLAB modules. Integration of these parts can be achieved through special MATLAB codes and simulink components. On the other hand, PSCAD includes the simulink part, and Fortran and C++ codes can be integrated to the simulation using special PSCAD modules.
- In MATLAB, the simulation of any power system can take a long time, and the simulation time increases with the size of the power system. In contrast, PSCAD can simulate the same power system and others in a short time, regardless of the size of the power system.
- The PSCAD library provides electrical components used in power systems that are more accurately imitated than other simulation programs.

By considering the advantages and disadvantages given above, the PSCAD program was preferred for use for the simulation of the SVM-controlled inverter. After the selection

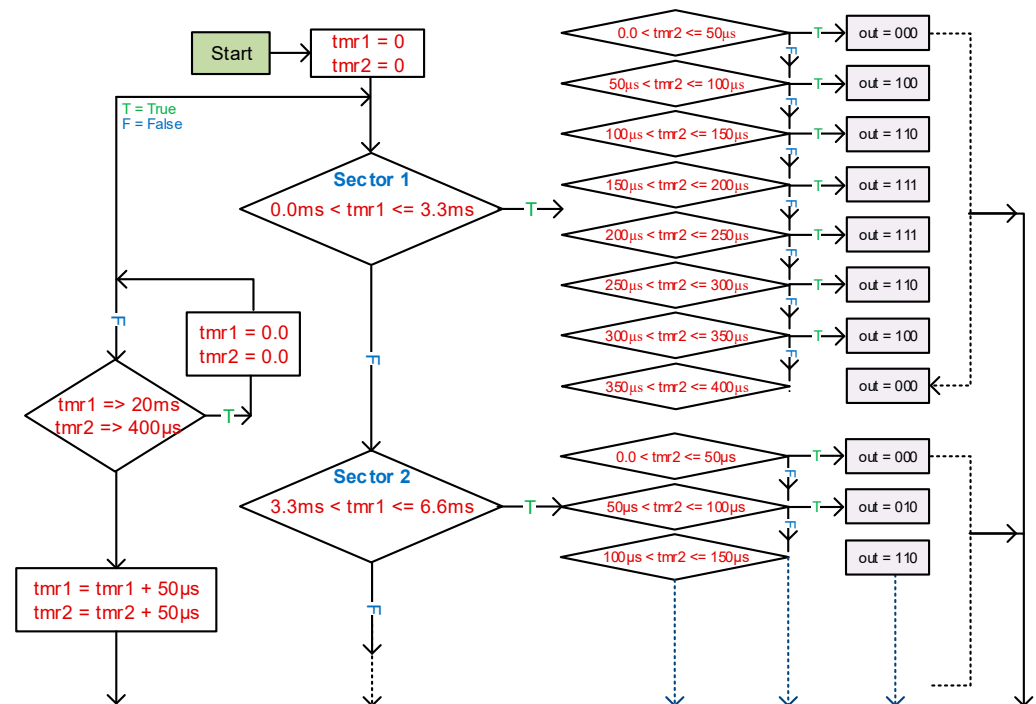


of the software, the simulation setup of the inverter can be explained in three steps. These steps are given in the following section.

#### 4.1.1. Step 1: Flow Chart of Basic Control Algorithm for SVM Technique

SVM should be performed by calculating the correct switching times and executing suitable switching sequences for all sectors in each  $t_s$ . The control algorithm's flow chart should be designed before application. Creating this algorithm, based on Equations (1)–(6), is not a simple task for programmers and may be challenging for undergraduate students to understand. Furthermore, complex algorithms are difficult to implement in microcontrollers due to the need for fast computing in floating-point calculations. Therefore, it is recommended to improve a simple control algorithm that can be easily understood by undergraduate students during laboratory sessions to learn the SVM technique effectively.

The switching times of the SVM algorithm are dependent on  $\phi$ , and their values instantaneously change in each sector. The switching sequences are applied according to these switching times. For example, the switching sequences of Sector 1 should be repeated in each  $t_s$  while  $U_{ref}$  is in Sector 1. The difference between the current and previous values of  $t_s$  is the change in  $t_1$  and  $t_2$  on-times. If we take the average of  $t_1$  and  $t_2$  on-times in any sector, considering Equations (4) and (5), the result will be the same. Therefore, we can assume that the on-times of  $t_1$  and  $t_2$  are equal for creating a simpler algorithm that can be more easily understood by undergraduate students. Furthermore, we can assume that the on-times of  $t_0$  and  $t_7$  are equal to those of  $t_1$  and  $t_2$ , considering Equation (6). The flow chart of the improved algorithm is illustrated in Figure 5 for the first two sectors, using the above assumptions.



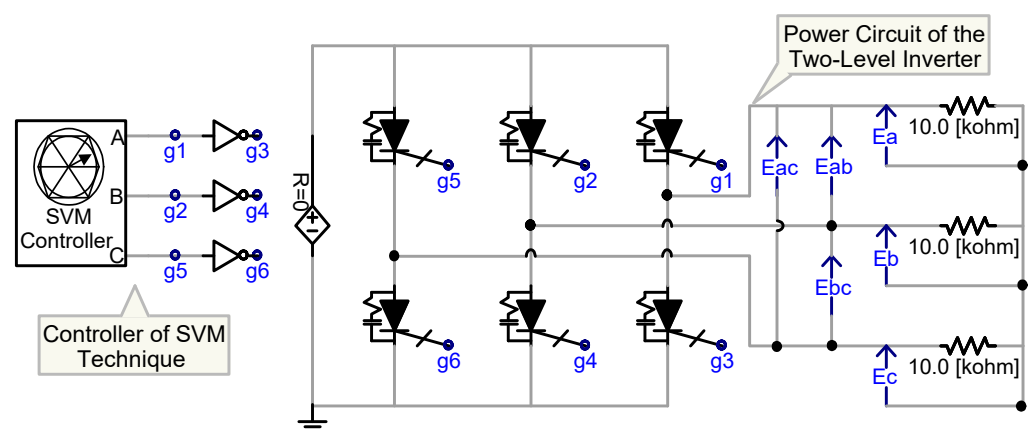
**Figure 5.** Flow chart of the improved algorithm for the first two sectors.

The duration of one period in a 50 Hz three-phase power system is 20 milliseconds, while the value of  $\phi$  changes from 0 to  $2\pi$ . The active vectors of any sector in which  $U_{ref}$  lies can be determined using either  $\phi$  or period duration. However, for the calculations of the improved algorithm, it is preferred to use period duration. In this case, the space vector diagram completes one cycle in 20 ms. The general SVM technique for a two-level inverter consists of six sectors, and the transition of  $U_{ref}$  from one sector to another takes 3.33 ms. The improved algorithm assumes that the switching times are equal and have

50  $\mu$ s duration. There are eight switching sequences for each sector, and for the first sector, the switching sequences given in Table 1 are realized in each  $t_s$  (400  $\mu$ s) between 0.0 to 3.33 ms using the improved algorithm. The rest of the algorithm can be completed by repeating the first sector and applying the correct switching sequences. Moreover, to make the improved algorithm more understandable for undergraduate students, one can create a component with Fortran codes in the simulation program. The sample codes for this component are provided in Step 3 of the following subsection.

#### 4.1.2. Step 2: Construction of the Test System

The PSCAD program was preferred to simulate the test system due to the fast computing and the easy integration of the programming codes to the simulation process. By using PSCAD, a three-phase, two-level inverter was created as a test system to make the SVM technique simple and understandable for the undergraduate students. In the test system, the rating of the source was selected as 10 V to minimize the risk of electric shock. The resistive load was used instead of inductive load to reduce the costs of the experimental design. Hence, the value of the load was selected as 10 k $\Omega$ . The Insulated-Gate Bipolar Transistors (IGBTs) were preferred as a switch in the inverter circuit due to the high-frequency switching ability. The power circuit and SVM controller of the test system are illustrated in Figure 6.



**Figure 6.** Power circuit of two-level DC/AC inverter and its SVM controller.

#### 4.1.3. Step 3: SVM Controller

There are many studies in the literature to show the readers how to create a component in PSCAD and how to integrate any Fortran codes into this component [35–37]. In this study, the creation of a component for the SVM controller and the integration of Fortran codes into the controller were actualized by considering these studies. The proposed algorithm for the SVM technique consists of four parts. The first part includes the definition of the local variables, and the clearing of the timer variables ( $t_{mr1}$  and  $t_{mr2}$ ) during the simulation time (time) is less than “0.0”. The Fortran codes of the first part are given in Figure 7.

```

!**** Definition of the local variables
#local real tmr1
#local real tmr2
#local real out

!**** Clearing of the timer variables before the program starting
if ((time).le.0.0) tmr1 = 0
if ((time).le.0.0) tmr2 = 0

```

**Figure 7.** Fortran codes of the first part.

The second part includes the calculation of each sector using  $\text{tmr}_1$  and the assignment of the switching sequences using  $\text{tmr}_2$ . In this part, the similar codes are repeated for each sector by changing the border of  $\text{tmr}_1$  and the switching sequences according to the sector in which  $U_{ref}$  lies. For this reason, the Fortran codes of the second part are illustrated for the first two sectors in Figure 8.

```

!**** Assignment of the switching sequences according to the timer variables
  if ((tmr1.ge.((0.02*0.0)/6)).and.(tmr1.lt.((0.02*1.0)/6))) then
    if ((tmr2.ge.(0*(delt))).and.(tmr2.lt.(1*(delt)))) out = 000
    if ((tmr2.ge.(1*(delt))).and.(tmr2.lt.(2*(delt)))) out = 100
    if ((tmr2.ge.(2*(delt))).and.(tmr2.lt.(3*(delt)))) out = 110
    if ((tmr2.ge.(3*(delt))).and.(tmr2.lt.(4*(delt)))) out = 111
    if ((tmr2.ge.(4*(delt))).and.(tmr2.lt.(5*(delt)))) out = 111
    if ((tmr2.ge.(5*(delt))).and.(tmr2.lt.(6*(delt)))) out = 110
    if ((tmr2.ge.(6*(delt))).and.(tmr2.lt.(7*(delt)))) out = 100
    if ((tmr2.ge.(7*(delt))).and.(tmr2.lt.(8*(delt)))) out = 000
  end if
  if ((tmr1.ge.((0.02*1.0)/6)).and.(tmr1.lt.((0.02*2.0)/6))) then
    if ((tmr2.ge.(0*(delt))).and.(tmr2.lt.(1*(delt)))) out = 000
    if ((tmr2.ge.(1*(delt))).and.(tmr2.lt.(2*(delt)))) out = 010
    if ((tmr2.ge.(2*(delt))).and.(tmr2.lt.(3*(delt)))) out = 110
    .....
    .....
    .....

```

Figure 8. Fortran codes of the second part.

The third part includes increasing the timer variables dynamically using the solution time step ( $\text{delt}$ ) of the simulation program and the resetting of the  $\text{tmr}_1$  and  $\text{tmr}_2$  values according to 20 ms and 400  $\mu\text{s}$ . In the real applications of the microcontrollers and DSPs, the value of  $\text{delt}$  is generally taken as 25 or 50  $\mu\text{s}$  by considering the length of the program codes. The processing of the proposed algorithm by the microcontrollers is completed at a time that is more than 25  $\mu\text{s}$ . Therefore,  $\text{delt}$  of the simulation is assigned as 50  $\mu\text{s}$ . The Fortran codes of the third part are given in Figure 9.

```

!**** Determination of the timer variables
  tmr1 = tmr1 + (delt)
  tmr2 = tmr2 + (delt)

!**** Resetting of the timer variables
  if (tmr1.ge.0.02) tmr1 = 0
  if (tmr2.ge.(8*(delt))) tmr2 = 0

```

Figure 9. Fortran codes of the third part.

The fourth part includes the application of the switching sequences to IGBTs according to the sectors. The similar codes are repeated for each sector by changing switching sequences. In the simulation, these switching sequences should be given to drive the switching devices using port connections. Therefore, the Graphical User Interface (GUI) of the SVM controller includes three port connections ( $\text{out}_1$ ,  $\text{out}_2$ , and  $\text{out}_3$ ) for the gates ( $g_1$ ,  $g_2$ , and  $g_3$ ) of IGBTs. It is important to note that the port connections of any component are not visible during the simulation workspace in PSCAD. The Fortran codes of the switching sequences and the GUI of the SVM controller with its hidden port connections are illustrated in Figure 10.

The similar codes are repeated for each sector in the second and fourth part of the simulation procedure. Therefore, the Fortran codes of the first sector were given as an

example to complete the other sectors easily by the undergraduate students. Moreover, this procedure can be given as homework before the simulation lectures of the PEC. If the Fortran codes of other sectors are completed and integrated into the GUI by adding one after another, respectively, the simulation example of the SVM algorithm will work properly. The voltage outputs of the simulation system given in Figure 6 are illustrated in the case studies section.

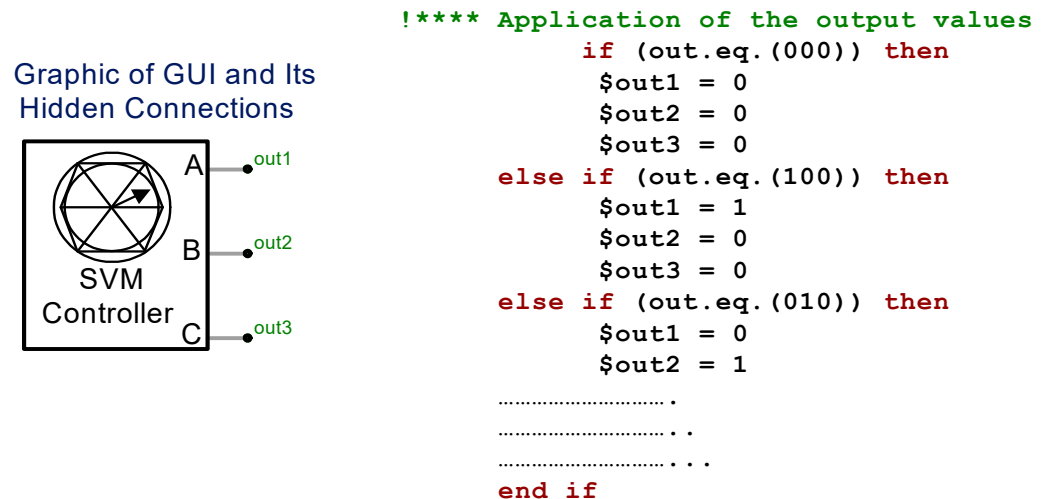


Figure 10. Fortran codes of the fourth part.

#### 4.2. Experimental Setup

In the previous section, a simple algorithm was explained with a graphical diagram for understanding the SVM technique by undergraduate students and easy implementation in the simulation environment for the PEC. In this section, the proposed SVM algorithm is performed in the laboratory environment for the PEL, and the experimental procedure is presented with details in three steps. These steps are given in the following.

##### 4.2.1. Step 1: Selecting of the Microcontroller Type

Microprocessor-based DSPs from Texas Instruments, ATMEGA microcontrollers from Atmel Corporation, and Peripheral Interface Circuit (PIC) microcontrollers from Microchip Technology were used for educational purposes in experimental studies. Generally, PICs and ATMEGAs are widely used in simple power electronics applications by undergraduate and graduate students. In Turkey, PICs are the most popular among their equivalents due to their wide range of products and easy availability.

Numerous studies support the use of microcontrollers in power electronics education through practical experiments, including a simulation model and experimental setup for a PIC-controlled two-level DC/AC inverter [37]. Various software programs are used for simulation studies and experimental implementation of the hybrid two-level DC/AC inverter. A brief overview of the development and technology of Fuel Cells (FCs) with their integration into undergraduate electrical engineering education was presented in [15,38]. These papers demonstrated how a small class project in power electronics could be used to analyze the behavior of a Proton Exchange Membrane (PEM) FC connected to a DC/DC boost converter designed by an undergraduate student. Other studies analyzed the best way to apply active and cooperative learning methodologies in a microprocessors course [39] and present a general-purpose experimental setup and four consecutive experiments on voltage source inverter-aided SCADA systems [40]. These papers gave detailed descriptions of some laboratory experiments in virtual-instrument-aided Supervisory Control and Data Acquisition (SCADA) systems and evaluated the advantages and challenges of these experiments.

Additionally, a paper introduced the newborn dsPIC30F microcontroller from Microchip and a new Harvard architecture. The paper gave examples of some simple applications written using assembly language for final year projects of undergraduate students [41]. In another study, the design and implementation of a real-time MATLAB interface to control the speed of an induction motor using a dsPIC30F4011 microcontroller for undergraduate students was presented [42]. In this application, a fuzzy logic controller was used to control the speed of the induction motor.

Several types of PIC microcontrollers were used in the experimental applications of PELs [37–42]. The program codes required for the tasks should be uploaded to the microcontrollers before the applications. It can be accomplished by utilizing the special programmers. These programmers are connected to the computers using RS232 or USB ports for data transferring or receiving to the microcontrollers. A new family named PIC18F was introduced by Microchip to eliminate these programmers. The microcontrollers of this family can be easily programmed with a USB port and interactively transfer data from the experimental board to the computer or vice versa. The PIC18F family was used in the experimental application of the proposed SVM algorithm by considering these features and PIC18F2550 was preferred for use for the special specifications given in the following:

- USBV2.0 compliant;
- 1 Kbyte dual access RAM for USB;
- Four crystal modes including high-precision PLL for USB;
- Two external clock modes, up to 48 MHz;
- C compiler-optimized architecture with optional extended instruction set;
- Dual oscillator options allow the microcontroller and USB module to run at different clock speeds.

#### 4.2.2. Step 2: Design and Construction Processes of the Experimental Board

In the education of Electrical and Electronics Engineering, the simulation programs are used to predict the behavior of a real circuit before the application. In other words, they are an alternative way to actualize the real experiments. They allow the user to observe an operation through simulation without actually performing. Hence, the costs caused by faults are minimized. Due to these positive aspects of the simulation programs, the experimental board of the proposed SVM algorithm was performed in the simulation environment firstly and then it was actualized with the real components. These processes are explained in the following subsections.

##### (a) Design Processes of the Experimental Board

Several interactive simulation programs that are compatible with the controlling of computer peripherals, updateable libraries, interactively designed toolbars for the GUI, and external code integration property and debugging are available on the market to use as instrumentation in professional engineering and engineering education. Proteus is one of the most preferred pieces of software to simulate the microcontroller-based electronic circuits [43]. Many analog and digital electronics circuits can be modeled, and the voltage and current waveforms of the circuits can be viewed graphically in this simulation program [37].

Proteus, as shown in Figure 11, can be also used to check the binary codes that the microcontroller can understand and execute before the real-time applications, and to design the graphical layout of the experimental board using its layout editor. Based on the specifications mentioned above, Proteus was the preferred software to simulate the power circuit of the two-level inverter shown in Figure 6. Further, the simulation of the experimental board is illustrated in Figure 12 by selecting the appropriate electronic components and the microcontroller from the Proteus library.



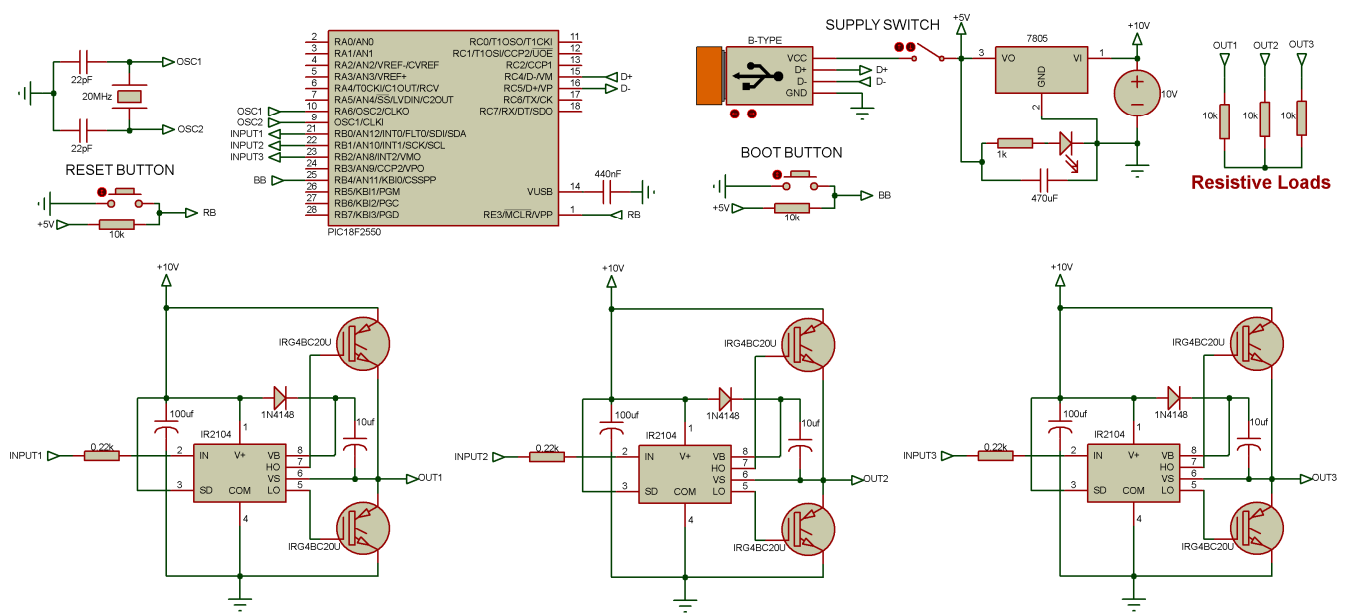


Figure 11. Proteus model of PIC-controlled two-level DC/AC three-phase inverter.

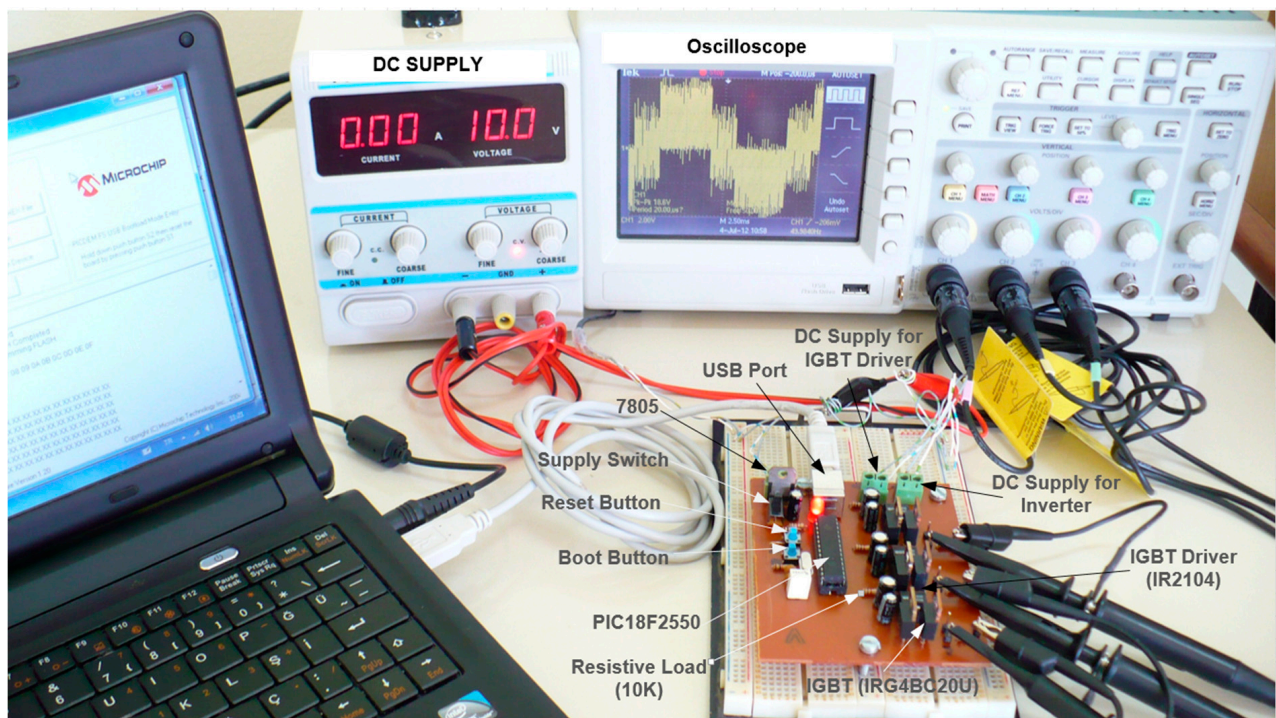


Figure 12. Experimental setup of the two-level inverter for undergraduate laboratory.

The experimental board can be divided into two parts. The first part can be named as the power circuit of the inverter. For this part, IRG4BC20U switches and IR2104 IGBT-Drivers were selected from the products of International Rectifier (IR). In inverters, anti-parallel diodes are generally used to protect IGBTs from reversed currents caused by inductive loads. However, in this circuit, 10 k $\Omega$  resistances were used to supply the resistive load and eliminate the need for anti-parallel diodes. The gate drive supply of IR2104 ranges from 10 to 20 V. Therefore, the voltage rating of the DC supply for the inverter was selected as 10 V to use the same supply as IR2104. The second part can be named



the power circuit of the microcontroller, which can be observed under four main topics given below.

#### 1. Supply selection

All types of PIC microcontrollers require a stable DC supply ranging between 5.0 and 5.5 V. Typically, a voltage regulator called LM7805 is utilized to ensure a stable 5 V supply for development boards of the microcontrollers. Alternatively, if the boards have a USB connection, they can also be powered with 5 V from the USB connection of the computer. However, incorrect connections of electronic components can lead to short circuits in electronic circuits, causing high currents to be drawn from the supply or main-board of the computer. These situations are not conducive to the robustness of electronic devices and can be avoided by selecting the proper supply and using a simple switch. In the power circuit illustrated in Figure 11, a switch called “supply switch” was utilized for this purpose. The switch was typically positioned on the right side of the external supply to prevent the short circuits at the USB connection of the computer.

#### 2. Preparation of the microcontroller for USB before the circuit connection

The PIC18F series has a USB port in its peripheral units. These microcontrollers can be easily programmed and activated using their USB connections. Before utilizing this unit, special codes called “boot loader” should be uploaded to the microcontroller using a PIC programmer. These codes introduce the microcontroller as a USB device to the computer. It is important to note that any USB device needs a driver to run smoothly with the computer. The PIC can then be programmed via USB using Microchip’s special software. The boot loader named “MCHPUSB Bootloader PICDEM FS USB Demo Board”, the software named “PICDEM FS”, and the driver can be downloaded from the official Microchip website under the name of MCHPFSUSB files [44].

#### 3. Online and offline modes selection

The microcontroller can operate in two modes: “Online” and “Offline”. In online mode, the microcontroller executes the boot-loader codes from 0x0000 to 0x07FF of its memory addresses and connects to the computer as a USB device. In this mode, it can only be programmed via USB port using the PICDEM FS program. In offline mode, the microcontroller executes the program codes from 0x0800 to 0x7FFF of its memory addresses and can perform specific functions according to the codes uploaded to it by the programmer. For example, it can send data taken from the experiments to the computer or receive instructions for controlling the experimental devices.

The fourth pin of port-B in the experimental board shown in Figure 10 was used to switch between online and offline modes of the microcontroller. The power circuit of the board included a push button named “Boot-Button” for this purpose. The microcontroller can be easily switched to offline mode by resetting it. To switch to online mode, the boot-button must be held down during the resetting process. Once the microcontroller is in online mode, any program codes can be downloaded or uploaded to it. If the port and pin number need to be changed, this can be accomplished using the MPLAB software of Microchip [44].

#### 4. Reset

The microcontroller may experience runtime errors due to voltage fluctuations in the supply or short circuits between its input and output terminals, which can prevent it from operating any program codes. In this situation, the microcontroller must be manually reset by the user. In the experimental board shown in Figure 10, a push button named “Reset Button” was used for this purpose. Generally, the microcontroller starts to operate the program codes from memory addresses 0x0000 to 0x7FFF after resetting. However, if it includes the boot loader, it begins operating the program codes from memory addresses 0x0800 to 0x7FFF.

#### (b) Construction Processes of the Experimental Board

A number of programs for designing electronic circuits have emerged in the market, which can be used in professional engineering and engineering education. Examples of such programs include ISIS, Eagle, and PCB Wizard [43,45,46]. These programs can even help reduce the time spent building circuits, especially when multiple units are produced. PCB Wizard is the easiest and most popular program among these programs, and it was used in this study to design the electronic circuit of the two-level inverter shown in Figure 10. After this process, the copper board for this circuit needs to be constructed for the experimental setup. There are several ways for undergraduate students to produce the copper board without using complex devices. The drying method is the easiest way to learn how to make a circuit board, which is used for making a prototype inverter suitable for the education of undergraduate students in PELs.

In the experimental procedure, the Microchip boot loader for PICDEM FS USB Demo Board was uploaded using a simple PIC programmer before installing PIC18F2550 onto the experimental board. After this, it was taken into online mode by holding down the boot-button while resetting it. This allows the program codes of the proposed SVM algorithm to be uploaded to its memory via USB connection using the PIC DEM FS program. Once the codes are uploaded, PIC18F2550 is activated to control the inverter with the program, as illustrated in Figure 12. If an error occurs during activation or operation of the codes, it should be restarted using the reset button. The experimental board can supply 4.6 V and 0.025 mA from each I/O pin of PIC18F2550 without using an external source [37]. However, for some applications, the power taken from the USB of the computer may not be enough, and an external source can be connected to the board using the supply switch.

There are many educational kits available from the companies such as National Instruments and other national firms. However, these educational kits can be expensive and require a significant amount of additional equipment. In addition, the assistance of a technician or an expert is required for demonstrating their operation to the students. In contrast, the experimental kit provided as an example in the article was designed to be much simpler and low cost. For these reasons, it was proposed in the paper that PEL education could be supported with the suggested experimental kit. The total cost of this kit is given in Table 2.

**Table 2.** Total cost of prototype SVM inverter board.

Part	Piece	Price (USD)	Total (USD)
PIC18F2550	1	14.75	14.75
LM7805 Voltage Regulator	1	0.50	0.50
IRG4BC20U IGBT	6	2.80	16.80
IR2104 IGBT Driver	3	2.18	6.54
20Mhz Oscillator	1	0.40	0.40
Other Passive Components	1	5.90	5.90
<b>Cost (USD)</b>			<b>44.89</b>

#### 4.2.3. Step 3: Program Codes of SVM Controller for PIC18f2550

Several software programs are available on the market to provide a complete integrated tool suite for developing and debugging embedded applications running on microcontrollers. They are designed to be smart and efficient, and students can rely on them to do the hard work. MPLAB, CCS-C, MikroC, and PICBasic are among the most popular programs used for this purpose in PELs [1,10,47,48]. Generally, these programs use the same commands as C language to actualize the flow chart describing the program algorithm and additional commands to prepare the microcontroller peripherals. Then, a *hex* file including the codes in binary form is created to upload the microcontroller using a PIC programmer. In the experimental procedure, CCS-C is used to create the hex file of the proposed SVM algorithm given in Figure 5. Then, the created hex file is easily uploaded via

USB connection to PIC18F2550. A simple algorithm was improved using CCS-C codes that could be learned quickly and applied easily by undergraduate students in the laboratory environment. It provides practical knowledge that can be applied to other laboratory studies. CCS-C codes of the algorithm consisted of four parts.

The first part includes the codes for the definition of the microcontroller type, preparation of the peripheral units, and adjustment of the memory allocation for the boot loader. These codes should be given in a specific order. Firstly, the type of the microcontroller, PIC18F2550, was introduced to the compiler by adding its library to the codes. Then, the value of the crystal oscillator was defined as 20 MHz and the pins of the port-B used for the input and output connections (io) were defined. Microcontrollers are designed for process control and are required to interface with the real system. Many of the peripheral units are integrated into the hardware of the microcontroller for those specific purposes. Secondly, these units should be prepared at the beginning of the code. The term “fuses” is used in the compiler to control the peripheral units. The required peripherals are adjusted for this application by writing their code after the fuses term.

The microcontroller may encounter runtime errors, rendering it incapable of operating any program codes. In such cases, the user must manually reset the microcontroller. Generally, vector addresses are used to indicate the start address of reset or interrupt subroutines. The addresses of the reset and interrupt vectors should be defined at the beginning of the code. CCS-C compiler assigns these vector addresses as 0x0000 and 0x0008, respectively. However, if the boot loader is used to program the microcontroller via USB or RS232 connection without using the programmer, the location of its codes from 0x0000 to 0x07FF addresses should be protected from writing any program code to. In such cases, the reset and interrupt vector addresses are changed to 0x0800 and 0x0808, respectively. The CCS-C codes of the first part are given in Figure 13.

```
#include <18F2550.h>
#use delay(clock=20000000)
#use fast_io(B)

#fuses NOWDT           //No Watch Dog Timer
#fuses XTPLL           //Crystal/Resonator with PLL enabled
#fuses NOPROTECT       //Code not protected from reading
#fuses NOCPD           //No EE protection
#fuses NODEBUG         //No Debug mode for ICD
#fuses NOWRT           //Program memory not write protected
#fuses NOWRTD          //Data EEPROM not write protected
#fuses NOWRTB          //Boot block not write protected
#fuses NOCPB           //No Boot Block code protection
#fuses MCLR            //Master Clear pin enabled
#fuses PLL5            //Divide By 5 (20MHz oscillator input)
#fuses CPUDIV2         //System Clock by 2
#fuses USBDIV          //USB clock source comes from PLL divide by 2
#fuses VREGEN          //USB voltage regulator enabled

#build (reset = 0x0800, interrupt = 0x0808)
#org 0x0000,0x07FF void bootloader18F22550(void) {}
```

Figure 13. CCS-C codes of the first part.

The second part includes the definition of the local variables, namely second and wait. The second variable is increased using the timer\_2 interrupt to determine the sectors of the space vector. Its value was divided into 3 units for each sector and is reset when it exceeds 18 units. The wait variable was used to assign the value of delay time used for an interval between the applications of switching positions. The CCS-C codes for the second part were given in Figure 14.

```

long int second = 0;
int wait = 0;

#int_timer2
void timer2(void)
{
    second++;
    if (second >= 18) { second = 0; }
}

```

Figure 14. CCS-C codes of the second part.

The third part includes the adjustment of interrupt specifications and pins of the microcontroller. There are three different timer interrupts on the microcontroller hardware, with timer\_2 being one of them and widely preferred for time applications. Therefore, it was used to increase the value of the second variable in this application. According to the settings given in Figure 15, it forces the microcontroller to suspend the execution of the main program and immediately execute the interrupt function given in Figure 14 for every 1.111 ms. The value of the second variable increases in each execution of this process, and it is automatically reset if it reaches 18. Then, a time duration occurs between the two resetting processes, equal to one period of the reference voltage, which can be increased or decreased by changing the settings of setup\_timer\_2.

```

void main(void)
{
    setup_timer_1 (t1_disabled);
    enable_interrupts (int_timer2);
    setup_timer_2 (t2_div_by_16,138,6);
    setup_timer_3 (t3_disabled);
    setup_CCP1 (ccp_off);
    setup_CCP2 (ccp_off);
    setup_adc_ports (no_analogs);
    set_tris_b (0x00);
    enable_interrupts (global);

    output_b (0x00);
    wait = 50;
}

```

Figure 15. CCS-C codes of the third part.

The wait variable was assigned a value of 50  $\mu$ s to apply the switching positions. The modulation index of the inverter can be increased or decreased by changing its value, and the full modulation can be achieved at the inverter output with this algorithm. All pins of port-B were set as out to use them for the output connections, and they were set as logic-zero only once before the infinite loop. The CCS-C codes for the third part are given in Figure 15.

The fourth part includes the application of the improved SVM algorithm. The first step of the algorithm was the determination of the sector, which was accomplished by tracking the value of the second variable. For example, if the second variable takes on integer values between 0 and 2, it indicates that the reference voltage vector lies in the first sector. Then, the switching positions of the first sector are applied in 50  $\mu$ s durations within this period. If this process is carried out for the other sectors, the SVM technique will be implemented for the two-level inverter with this algorithm. Implementing this part with CCS-C codes is very simple to understand for undergraduate students. The similar codes were repeated for each sector by changing the border values of the second variable and the switching positions according to the sector in which the  $U_{ref}$  vector lies. For this reason, the Fortran codes of the fourth part are illustrated for the first two sectors in Figure 16.

```

while(1)
{
    if ((second>=0) && (second<3))
    {
        output_b(0b00000000); delay_us(wait);
        output_b(0b00000001); delay_us(wait);
        output_b(0b00000011); delay_us(wait);
        output_b(0b00000111); delay_us(wait);
        output_b(0b00000111); delay_us(wait);
        output_b(0b00000011); delay_us(wait);
        output_b(0b00000001); delay_us(wait);
        output_b(0b00000000); delay_us(wait);
    }
    if ((second>=3) && (second<6))
    {
        output_b(0b00000000); delay_us(wait);
        output_b(0b00000010); delay_us(wait);
        output_b(0b00000011); delay_us(wait);
        .....
        .....
        .....
    } // While
} // Main

```

Figure 16. CCS-C codes of the fourth part.

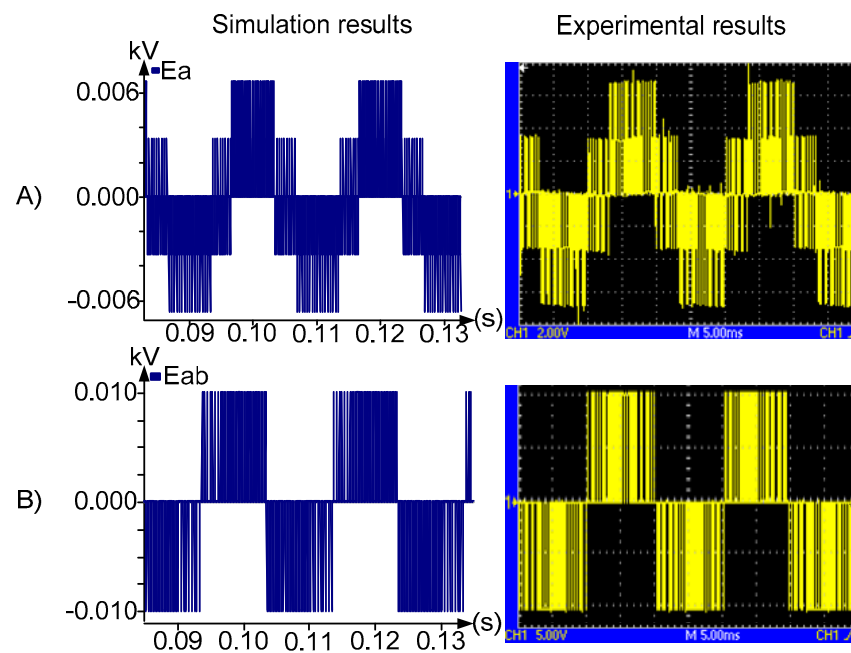
The peripheral devices and interrupts of the microcontroller were adjusted in the first three parts, followed by the application of the proposed SVM algorithm in the last part of this subsection. Similar codes were repeated for each sector in the fourth part. Therefore, CCS-C codes of the first sector were provided as an example to enable undergraduate students to easily complete the other sectors. This procedure can also be given as homework before the simulation lectures of PELs. Once the CCS-C codes of the other sectors are completed and added one after another, respectively, they can be compiled using CCS-C and uploaded to the microcontroller via USB port. The experimental prototype board will work properly. The voltage outputs of the experimental test system given in this study are presented graphically in the section on experimental case studies.

## 5. Experimental Case Studies

The two experimental case studies are presented in this section to demonstrate the easy application of the proposed SVM algorithm and its effectiveness for undergraduate students to learn. The experimental results are also illustrated alongside the simulation results to demonstrate the validation of PSCAD.

### 5.1. Case 1: Three-Phase Balanced-Load Condition

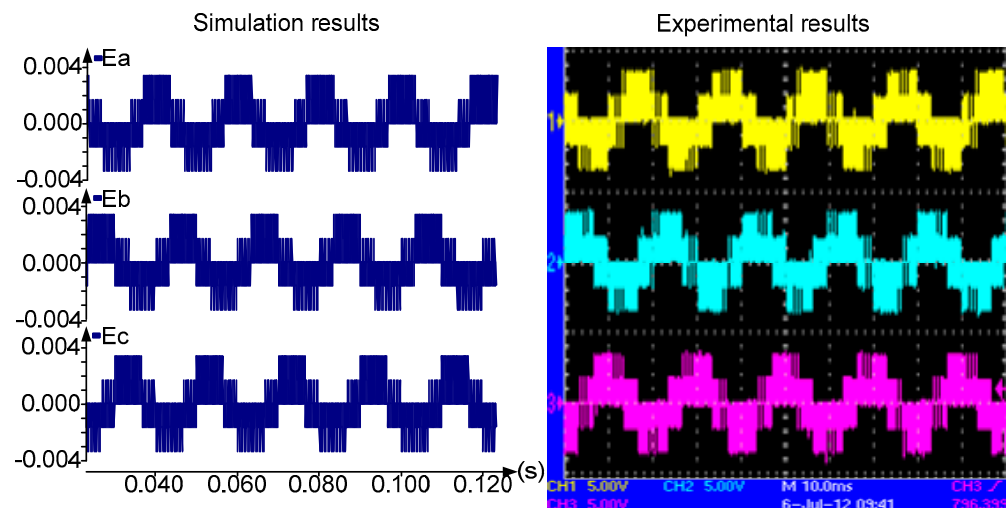
Both simulation (shown in Figure 6) and experimental (given in Figure 11) studies utilized a three-phase balanced-load system. The inverter's voltage rating and the three-phase resistive load in both systems were assigned as 10 V and 10 k $\Omega$ , respectively. The line-phase ( $E_a$ ) and line-line ( $E_{ab}$ ) voltages for phase A are compared in Figure 17 to illustrate the simulation results and experimental results.



**Figure 17.** (A) Line-ground and (B) line-line voltages for phase A.

### 5.2. Case 2: Three-Phase Unbalanced Load Condition

The second case study involved the creation of an unbalanced load system for both simulation and experimental studies to observe the behavior of the SVM technique during unbalanced system conditions. The inverter's voltage rating was assigned as 5 V. The three-phase resistive loads were selected as 10 k $\Omega$  for phase A, 5 k $\Omega$  for phase B, and 10 k $\Omega$  for phase C to create the unbalanced load conditions. The line-phase voltages are compared in Figure 18 to illustrate the simulation and experimental results.



**Figure 18.** Line-phase voltages of unbalanced load conditions.

The behavior of the SVM technique was observed under both balanced and unbalanced load conditions in the case studies. The line-phase and line-line voltages were compared in the simulation and experimental results. The results of the simulation and experimental case studies gave similar outputs that were closely matched, demonstrating the validation of PSCAD. In the first case study, the standard outputs of the SVM technique were obtained during balanced load conditions. However, in the second case study, distorted outputs were obtained from the SVM technique under unbalanced load conditions. Specifically, the



waveforms and magnitudes of phase A and phase C voltages were increased, while those of phase B were decreased. These results demonstrated the negative effects of unbalanced system conditions on the SVM technique.

## 6. Learning Objectives and Measurable Outcomes

A comprehensive understanding and proficient application of SVM is essential for undergraduate students and researchers working in the field of power electronics. The learning material on the SVM technique presented in this study aims to educate undergraduate students, both theoretically and in practical applications. The following learning objectives were targeted to be achieved through an academic process:

- To enable the undergraduate students to learn the theory of the SVM technique with mathematical derivation;
- To enhance familiarity and usage of various simulation programs among undergraduate students, enabling them to gain practical experience in applying theoretical concepts and conducting virtual experiments;
- To familiarize undergraduate students with microprocessors and microcontrollers and enable them to use these devices in their experimental studies;
- To teach undergraduate students algorithm creation using programming languages used in this study and encourage them to apply their algorithms to SVM techniques;
- To develop undergraduate students' circuit-making skills using the experimental setup in this study;
- To demonstrate to undergraduate students that simulation programs provide results similar to experimental studies, allowing them to verify the accuracy of their experiments before proceeding to the experimental stage.

In order to evaluate the achievement of the aforementioned learning objectives, a survey with six questions was conducted at the end of the autumn semester in the 2021–2022 academic year. Twenty-five students enrolled in the Power Electronics Course participated and rated the quality and clarity of the materials provided in this study. The questions and results are summarized in Table 3. According to the survey results, the lowest rating was 4.44, indicating that the majority of students were satisfied with the simulation and experimental application of SVM techniques, and the learning outcomes were well-received. After the survey, positive feedback was received during a meeting with the students, which was encouraging for future experiments using other PWM techniques. They also emphasized that their understanding of theoretical subjects was enhanced when the education was supported by interactive simulations and hands-on experiments.

**Table 3.** Student assessment of the SVM experiment.

Questions	E (5)	G (4)	F (3)	P (2)	VP (1)	Mean
Evaluate the theoretical explanation of SVM technique	13	10	2	0	0	4.44
Evaluate the simulation studies of SVM experiment	15	8	2	0	0	4.52
Evaluate the SVM experiment	20	5	0	0	0	4.8
Evaluate the laboratory manual	19	4	2	0	0	4.68
Evaluate the technical facilities of laboratory	18	6	1	0	0	4.68
Evaluate the overall laboratory experiment	12	12	1	0	0	4.44

## 7. Educational Benefits

Theoretical knowledge of the PEC and practical experience of the PEL can be learned in classroom and laboratory environments by undergraduate students, respectively. After these educational processes and related procedures, the students' achievements should be evaluated with quizzes and experimental projects. Modulation techniques are important topics covered in the PEC, with SVM being the most commonly used modulation technique

in inverter topologies. The theoretical background of this method is based on complex mathematical equations, and its implementation with a microcontroller requires good programming ability and a basic knowledge of computer vision. The objective of this study was to bridge the gap between the theoretical background of the SVM technique and its practical application for undergraduate students through experimental case studies.

The other educational benefits and the experimental gains of this study are summarized below:

- The theory of SVM is reinforced with the improved algorithm in PSCAD, which can be easily modified by undergraduate students to adapt to higher-level inverters.
- The improved algorithms in PSCAD and CCS-C programs can assist undergraduate students in developing new algorithms for the SVM technique.
- The values of resistive loads in the second case study can be changed to create different unbalanced load conditions. This allows for convenient observation of the effects of these conditions on the SVM technique in both simulation and experimental case studies.
- The frequency of the inverter output voltages can be increased or decreased by changing the values of the *timer2* variable in PSCAD and the *timer\_2* variable in CCS-C. This allows for the creation of new case studies for PELs by adjusting the frequency of the output voltage and the power circuit of the inverter.
- The effective use of PSCAD, CCS-C, Proteus, MPLAB, PCB-Wizard, and PICDEM programs was learned through this study, and they are actively used in PEC and PEL lessons of Adana Alparslan Türkeş Science and Technology University.
- In the experimental board, the Metal-Oxide-Semiconductor Field-Effect Transistor (MOSFET) can be used instead of IGBT due to the properties of the driver (IR2104). This allows for the creation of a new case study using MOSFET instead of IGBT in the power circuit of the inverter, and a performance comparison can be made between MOSFET and IGBT by considering the experimental results.
- The effective use of PSCAD in the education of PEC and PEL was demonstrated in this study.

Given the educational benefits and experimental gains mentioned above, undergraduate students have more opportunities to relate theoretical concepts to practical problems presented in design sessions [14].

## 8. Conclusions

The PEC provides an introduction to the analysis, design, and modeling of power electronic systems and its content is designed to meet the requirements of industrial applications along with their solutions. The main topics of this study are based on theoretical knowledge and experimental applications, which should be taught to undergraduate students through visual learning and hands-on experimental projects. This can be achieved using simulation programs and carrying out experiments in PELs. The PEL is an application of experimental prototype solutions for power electronics systems that should be convenient, inexpensive, available, and powerful for general-purpose industrial applications. Generally, microcontrollers are used to control the power electronic-based devices in these solutions. By training in PELs, students can develop the ability to use microcontrollers and relate theoretical knowledge to practical applications through microcontroller-based projects.

This paper presented the simulation model of SVM for PECs and the experimental procedure of the two-level, three-phase inverter for PELs to simplify understanding of the operating principles and controllers of the inverters. The simulation model and experimental setup of the inverter were created step-by-step for undergraduate students in both PECs and PELs. During the development procedure, students will be able to learn the theory behind the explanation of the SVM technique, simulation of the inverter with CCS-C, design of the experimental card with PCB, controlling of the experimental process using microcontrollers, and verification of results using PSCAD. The Fortran codes of the inverter controller in PSCAD and the microcontroller in CCS-C were also shared to enable

fast learning of the SVM technique for undergraduate students. After assembling these codes, students will fully understand how the SVM technique operates and how changes in different parameters affect its performance.

The board for the experimental prototype can be used to facilitate inverter applications in PELs. This allows for detailed learning of inverter technologies and quick proof of the SVM technique, all at a low cost. Additionally, the course trainer and students require fewer hours to learn the fundamentals of power electronics. In this study, various programs given in Table 4 were used for simulation studies and experimental implementations. In the case studies, the behaviors of the SVM technique were observed under balanced and unbalanced load conditions, and simulation and experimental results were given in the same scale graphically. They give similar outputs that were closely matched, and the negative effects of unbalanced system conditions on the SVM technique were also observed in the case studies. With the theoretical and experimental information obtained from this study, students and engineers can gain practical experience in using simulation programs, controlling microcontroller-based power electronic projects, and have the freedom to design additional features in their future works.

**Table 4.** Software and programs used in this study.

Software	Intended Use for This Study
PSCAD V 5	Power system and power electronic circuit simulation program [49]
CCS-C V 5.093	Programming code generator program for PIC microcontroller [50]
PROTEUS V 8.16	Electronic circuit simulation program [51]
MPLAB V 8.92	Another programming code generator program for PIC microcontroller [44]
PCB-Wizard 3 V4.0	Circuit layout drawing program [46]
Microchip PICDEM 2 Plus	Programming code downloader program to PIC microcontroller using the USB port and its protocols [52]

**Author Contributions:** Conceptualization, L.S.; methodology, L.S.; software, L.S.; validation, L.S.; formal analysis, L.S. and A.A.; investigation, L.S.; resources, L.S.; data curation, L.S.; writing—original draft preparation, L.S. and A.A.; writing—review and editing, L.S. and A.A.; visualization, L.S. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Data Availability Statement:** Not applicable.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

- Choi, S.; Saeedifard, M. An Educational Laboratory for Digital Control and Rapid Prototyping of Power Electronic Circuits. *IEEE Trans. Educ.* **2011**, *55*, 263–270. [\[CrossRef\]](#)
- Anand, S.; Farswan, R.S.; Fernandes, B.G. Unique Power Electronics and Drives Experimental Bench (PEDEB) to Facilitate Learning and Research. *IEEE Trans. Educ.* **2012**, *55*, 573–579. [\[CrossRef\]](#)
- Abramovitz, A. Teaching Behavioral Modeling and Simulation Techniques for Power Electronics Courses. *IEEE Trans. Educ.* **2010**, *54*, 523–530. [\[CrossRef\]](#)
- Arief, Y.Z.; Masdi, H.; Roslan, N.I.; Saad, M.H.I.; Eteruddin, H.; Al Hakim, R.R. Investigation of Various Faults of 500 KV Transmission Line Design in Sarawak, Malaysia Using Power Systems Computer Aided Design. In Proceedings of the 2022 Fifth International Conference on Vocational Education and Electrical Engineering (ICVEE), Surabaya, Indonesia, 10–11 September 2022; pp. 146–150.
- Horton, B. MATLAB Based Teaching Resources for the Modern Classroom. In Proceedings of the 33rd Australasian Association for Engineering Education Conference (AAEE 2022): Future of Engineering Education: Future of Engineering Education; Australasian Association for Engineering Education (AAEE), a Technical Society of Engineers Australia, Sydney, Australia, 4–7 December 2022; p. 494.
- Liao, W.-H.; Wang, S.-C.; Liu, Y.-H. Generalized Simulation Model for a Switched-Mode Power Supply Design Course Using MATLAB/SIMULINK. *IEEE Trans. Educ.* **2011**, *55*, 36–47. [\[CrossRef\]](#)

7. Kayisli, K.; Tuncer, S.; Poyraz, M. An Educational Tool for Fundamental DC–DC Converter Circuits and Active Power Factor Correction Applications. *Comput. Appl. Eng. Educ.* **2013**, *21*, 113–134. [[CrossRef](#)]
8. Gunasekaran, M.; Potluri, R. Low-Cost Undergraduate Control Systems Experiments Using Microcontroller-Based Control of a DC Motor. *IEEE Trans. Educ.* **2012**, *55*, 508–516. [[CrossRef](#)]
9. Chandra, A.P.J.; Venugopal, C.R. Novel Design Solutions for Remote Access, Acquire and Control of Laboratory Experiments on DC Machines. *IEEE Trans. Instrum. Meas.* **2011**, *61*, 349–357. [[CrossRef](#)]
10. Chu, R.H.; Lu, D.D.-C.; Sathiakumar, S. Project-Based Lab Teaching for Power Electronics and Drives. *IEEE Trans. Educ.* **2008**, *51*, 108–113. [[CrossRef](#)]
11. Doan, N.S.; Tsvetkov, A.N.; Nguyen, T.H. Study and Implementation of Space Vector Pulse Width Modulation Inverter on an Arduino. In *E3S Web of Conferences*; EDP Sciences: Les Ulis, France, 2021; Volume 288, p. 01059.
12. Aktaş, M.; Çavuş, B. A Computer-aided Educational Tool for Vector Control of AC Motors in Graduate Courses. *Comput. Appl. Eng. Educ.* **2020**, *28*, 705–723. [[CrossRef](#)]
13. Qaraei, P.Y.; Hashemi, A.; Salehi, M.H.; Karimi, M.; Moradi, A. DC Machines Remote Lab Case Study in Technical and Vocational University. *Comput. Appl. Eng. Educ.* **2022**, *30*, 1440–1453. [[CrossRef](#)]
14. Miaja, P.F.; Lamar, D.G.; de Azpeitia, M.A.P.; Rodríguez, A.; Rodríguez, M.; Hernando, M.M. A Switching-Mode Power Supply Design Tool to Improve Learning in a Power Electronics Course. *IEEE Trans. Educ.* **2010**, *54*, 104–113. [[CrossRef](#)]
15. Ortiz-Rivera, E.; Salazar-Llinas, A.; Velez-Delgado, J. An Enriched Undergraduate Research Experience Based on the Simulation, Experiments, and Theory of Fuel Cells. In Proceedings of the 2009 39th IEEE Frontiers in Education Conference, San Antonio, TX, USA, 18–21 October 2009; pp. 1–6.
16. Luo, F.L.; Ye, H. *Advanced DC/AC Inverters: Applications in Renewable Energy*; CRC Press: Singapore, 2017; ISBN 1351832352.
17. Yorat, E.; Özbek, N.S.; Saribulut, L. Performance Evaluation of Brushless Direct Current Motor Control Methods through Low-Cost Microcontroller-Based Real-Time Experiments. *Gazi Univ. J. Sci. Part C Des. Technol.* **2023**, *10*, 76–84. [[CrossRef](#)]
18. Parikh, R. Operation Overview of Three Phase Inverter with 120 150 and 180 Conduction Mode. In Proceedings of the 3rd International Conference on Emerging Technologies in Engineering, Biomedical, Medical and Science (ETEBMS-2017), New York, NY, USA, 17 July 2017; pp. 148–151.
19. Krismadinata, K.; Husnaini, I.; Asnil, A.; Lapisa, R.; Omar, R. Comparison of Various Conduction Modes for Three-Phase Inverter. *J. Appl. Inf. Technol.* **2022**, *100*, 2343–2357.
20. Saribulut, L.; Tümay, M. Robust Space Vector Modulation Technique for Unbalance Voltage Disturbances. *Electr. Power Syst. Res.* **2010**, *80*, 1364–1374. [[CrossRef](#)]
21. Bellini, A.; Bifaretti, S. Comparison between Sinusoidal PWM and Space Vector Modulation Techniques for NPC Inverters. In Proceedings of the 2005 IEEE Russia Power Tech, St. Petersburg, Russia, 27–30 June 2005; pp. 1–7.
22. Grandi, G.; Rossi, C.; Lega, A.; Casadei, D. Multilevel Operation of a Dual Two-Level Inverter with Power Balancing Capability. In Proceedings of the Conference Record of the 2006 IEEE Industry Applications Conference Forty-First IAS Annual Meeting, Tampa, FL, USA, 8–12 October 2006; Volume 2, pp. 603–610.
23. Srinivas, S.; Somasekhar, V.T. Switching Algorithms for the Dual Inverter Fed Open-End Winding Induction Motor Drive for 3-Level Voltage Space Phasor Generation. *Asian Power Electron. J.* **2007**, *1*, 96–110.
24. Rao, K.A.S.M.; Nadampalli, S.; Somasekhar, V.T. An Efficient Switching Algorithm for the Implementation of Synchronous Space Vector Modulation for an Induction Motor Drive with (V/f) Control. *Asian Power Electron. J.* **2008**, *2*, 83–90.
25. Muralidhara, B.; Ramachandran, A.; Srinivasan, A.; Channa Reddy, M. Space Vector PWM Signal Generation for a Three Phase Inverter and Hardware Implementation Using  $\mu$ -Controller. *Int. J. Eng. Sci. Technol.* **2010**, *2*, 5074–5979.
26. Barbi, I.; Batista, F.A.B. Space Vector Modulation for Two-Level Unidirectional PWM Rectifiers. *IEEE Trans. Power Electron.* **2009**, *25*, 178–187. [[CrossRef](#)]
27. Nisha, G.K.; Ushakumari, S.; Lakaparampil, Z.V. Harmonic Elimination of Space Vector Modulated Three Phase Inverter. In Proceedings of the International Multi Conference of Engineer and Computer Scientists, Hong Kong, China, 14–16 March 2012; Volume 2, pp. 14–16.
28. Dokić, B.L.; Blanuša, B. *Power Electronics*, 3rd ed.; Springer Cham: London, UK, 2015.
29. Akagi, H.; Watanabe, E.H.; Aredes, M. *Instantaneous Power Theory and Applications to Power Conditioning*; John Wiley & Sons: Hoboken, NJ, USA, 2017; ISBN 1118362101.
30. Babu, N.; Agarwal, P. Nearest and Non-Nearest Three Vector Modulations of NPCI Using Two-Level Space Vector Diagram—A Novel Approach. *IEEE Trans. Ind. Appl.* **2017**, *54*, 2400–2415. [[CrossRef](#)]
31. Mohiuddin, M.; Maricar, N.; Kadir, K.; Roslan, N.F.; Islam, M.; Khan, S.; Aboadla, E. Mitigation of Harmonics Using Novel Sector-Based Switching Pattern Space Vector Pulse Width Modulation. *IEEE Access* **2023**, *11*, 51465–51479. [[CrossRef](#)]
32. Zhou, K.; Wang, D. Relationship between Space-Vector Modulation and Three-Phase Carrier-Based PWM: A Comprehensive Analysis [Three-Phase Inverters]. *IEEE Trans. Ind. Electron.* **2002**, *49*, 186–196. [[CrossRef](#)]
33. Gupta, A.K.; Khambadkone, A.M. A Space Vector PWM Scheme for Multilevel Inverters Based on Two-Level Space Vector PWM. *IEEE Trans. Ind. Electron.* **2006**, *53*, 1631–1639. [[CrossRef](#)]
34. Saribulut, L.; Teke, A.; Tümay, M. Vector-Based Reference Location Estimating for Space Vector Modulation Technique. *Electr. Power Syst. Res.* **2012**, *86*, 51–60. [[CrossRef](#)]

35. Gianni, S.; Perez, A. Modeling Relays for Power System Protection Studies. Ph.D. Thesis, University of Saskatchewan, Saskatoon, SK, Canada, 2006.
36. Teke, A.; Meral, M.E.; Saribulut, L.; Tümay, M. Implementation of Fuzzy Logic Controller Using FORTRAN Language in PSCAD/EMTDC. *Int. J. Electr. Eng. Educ.* **2011**, *48*, 372–390. [[CrossRef](#)]
37. Saribulut, L.; Teke, A.; Meral, M.E.; Tümay, M. Computer Simulation and Experimental Implementation of a PIC-controlled Hybrid Multilevel Inverter for Undergraduate Laboratories. *Comput. Appl. Eng. Educ.* **2012**, *20*, 445–453. [[CrossRef](#)]
38. Dudrik, J.; Bauer, P. New Methods in Teaching of Power Electronics Converters and Devices. *Int. J. Eng. Educ.* **2008**, *24*, 1040–1048.
39. Carpeño, A.; Arriaga, J.; Corredor, J.; Hernández, J. The Key Factors of an Active Learning Method in a Microprocessors Course. *IEEE Trans. Educ.* **2010**, *54*, 229–235. [[CrossRef](#)]
40. Sahin, S.; Olmez, M.; Isler, Y. Microcontroller-Based Experimental Setup and Experiments for SCADA Education. *IEEE Trans. Educ.* **2009**, *53*, 437–444. [[CrossRef](#)]
41. Mohanna, Y.; Hamad, M.; Jabr, R.; Alaeddine, A.; Bazzi, O. Teaching Microprocessors, Microcontrollers, and Digital Signal Processing Courses Using Only One Target Processor: The Newborn DsPIC30F™ from Microchip™. *Comput. Appl. Eng. Educ.* **2007**, *15*, 185–191. [[CrossRef](#)]
42. Arulmozhiyal, R.; Baskaran, K.; Devarajan, N.; Kanagaraj, J. Real Time MATLAB Interface for Speed Control of Induction Motor Drive Using DsPIC 30F4011. *Int. J. Comput. Appl.* **2010**, *1*, 85–90. [[CrossRef](#)]
43. Waluyo, B.D.; Bintang, S.; Januariyansah, S. The Effect of Using Proteus Software as a Virtual Laboratory on Student Learning Outcomes. *Paedagogia J. Kaji. Penelit. Dan Pengemb. Kependidikan* **2021**, *12*, 140–145.
44. Microchip Technology Inc. MPLAB X Integrated Development Environment (IDE). Available online: <https://www.microchip.com/en-us/tools-resources/develop/mplab-x-ide> (accessed on 9 December 2022).
45. Monk, S.; Amos, D. *Make Your Own PCBs with EAGLE: From Schematic Designs to Finished Boards*; McGraw-Hill Education: New York, NY, USA, 2017; ISBN 1260019195.
46. New Wave Concepts PCB Wizard 3 [Computer Software]. Available online: <https://www.new-wave-concepts.com/ed/wizard.html> (accessed on 10 May 2023).
47. Mohamed, M.A.A. VFST: Virtual and Fully Software Based Toolchain for PC Interfacing Education and Research. *Comput. Appl. Eng. Educ.* **2023**, *31*, 389–407. [[CrossRef](#)]
48. Birbir, Y.; Yurtbasi, K.; Kanburoglu, V. Design of a Single-Phase SPWM Inverter Application with PIC Micro Controller. *Eng. Sci. Technol. Int. J.* **2019**, *22*, 592–599. [[CrossRef](#)]
49. PSCAD. PSCAD Software. Available online: <https://www.pscad.com/software/pscads/overview> (accessed on 12 June 2023).
50. Custom Computer Services (CCS). C Compiler Solution Suite. Available online: <https://www.ccsinfo.com/compilers.php> (accessed on 12 June 2023).
51. Proteus. Proteus Software. Available online: <https://www.labcenter.com/> (accessed on 12 June 2023).
52. Microchip. PICDEM 2 PLUS. Available online: <https://www.microchip.com/en-us/development-tool/DM163022-1> (accessed on 12 June 2023).

**Disclaimer/Publisher's Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.